

# Projet Intégration Moduleco-Madkit

## Premiers Résultats

### *The Moduleco-MadKit integration project* *First results*

#### **Fabien Michel, Jacques Ferber**

LIRMM

161, Rue Ada 34092 Montpellier Cedex 5 - France

Tel. 33 4 67 41 85 81 et 85 07

Fax 33 4 67 54 85 00

E.mail : [fmichel@lirmm.fr](mailto:fmichel@lirmm.fr), [ferber@lirmm.fr](mailto:ferber@lirmm.fr)

#### **Gilles Daniel**

AI Group, Computer Science Department

University of Manchester, UK

E.mail : [gilles.daniel@cs.man.ac.uk](mailto:gilles.daniel@cs.man.ac.uk)

#### **Denis Phan**

CREM UMR 6211

Université de Rennes 1, , Rennes, France

E.mail : [denis.phan@univ-rennes1.fr](mailto:denis.phan@univ-rennes1.fr)

#### **Résumé**

L'objectif du projet *intégration-MMM* est d'obtenir une version stable d'un logiciel modulaire et générique fondé sur le paradigme agent et permettant de réaliser des plates-formes de simulation hétérogènes dédiées à la modélisation des systèmes complexes. Le moyen choisi pour atteindre cet objectif consiste à utiliser le cadriciel Moduleco (Phan D. et Beugnard A., 2001) pour éprouver selon deux points de vues complémentaires les fonctionnalités offertes par le processus d'intégration de l'environnement multi-agents Madkit (Gutknecht O. *et al.*, 2001) et de la méta-plateforme Mimosa (Müller J.P., 2004). La présente communication est centrée sur le premier axe d'intégration : Moduleco-Madkit.

#### **Abstract**

*The goal of the MMM-integration project is to obtain a stable release of an agent-based modular and generic software designed for the simulation of complex systems. To reach this objective, the Moduleco software is used as a case-study to trial the integration capabilities of the MadKit and of the Mimosa meta-model. This extended abstract is focused on the first integration process: Moduleco-MadKit*

**Mots-clés** : Simulation multi-agents, Moduleco, Madkit

**Keywords**: *agent-based simulation, Moduleco, Madkit*

## **1. Les plates-formes Moduleco et MadKit**

Moduleco<sup>1</sup> est un cadriciel multi-agents qui permet la réalisation de modèles qui se situent à l'intersection de plusieurs disciplines et de plusieurs points de vue. Ce prototype a été conçu pour simuler les marchés, les organisations, les phénomènes sociaux et la dynamique des populations. Il s'agit d'étudier les conditions de prises de décisions d'agents cognitifs en interaction au sein de réseaux. Cette approche est appelée de *l'économie computationnelle* (Phan D., 2004). Dans ce cadre, Moduleco permet notamment

---

<sup>1</sup> <http://www.cs.man.ac.uk/ai/public/moduleco> ; <http://perso.univ-rennes1.fr/denis.phan/moduleco/>

de définir et de gérer différents types de voisinage (une forme d'organisation; un marché par exemple), grâce à la réification de la notion de *médium*. La définition d'un *Medium* particulier permet ainsi de spécifier les liens et les règles d'interaction du réseau social considéré.

Madkit<sup>2</sup> est une plate-forme générique de développement et d'exécution de systèmes multi-agents (SMAs) basée sur le modèle agent/groupe/rôle (AGR) (Gutknecht O. *et al.*, 2001). Le modèle AGR se fonde sur un principe de conception de SMAs basé sur l'utilisation de concepts organisationnels (approche OCMAS : Organization Centered Multi-Agent Systems, Ferber J. *et al.*, 2003). L'idée fondamentale de cette approche est de considérer qu'une structure organisationnelle permet de faciliter l'analyse, la conception et l'exécution de systèmes multi-agents hétérogènes. Le modèle AGR est ainsi basé sur trois concepts clés : *l'agent*, le *groupe* et le *rôle*. L'agent est décrit comme une entité autonome communicante qui joue des rôles au sein de différents groupes. Un rôle est une représentation abstraite d'une fonction, d'une position sociale occupée par un agent dans un groupe. Dans Madkit, le modèle AGR est utilisé à la fois pour fournir une structure organisationnelle aux systèmes multi-agents qui sont exécutés et pour son propre fonctionnement interne. L'implémentation de Madkit intègre par ailleurs trois principes de conception supplémentaires : une *architecture à micro noyau*, *l'agentification systématique des services* et l'utilisation d'un *modèle graphique componentiel*.

## 2. Intégration Moduleco/Madkit

Les difficultés liées au processus d'intégration d'une architecture logicielle multi-agents préexistante dans une autre plate-forme plus générique se situent principalement à deux niveaux. Le premier concerne l'intégration des aspects conceptuels qui sont liés à l'assimilation du modèle de conception inhérent au système. La plate-forme hôte doit en effet impérativement fournir un moyen de conserver les modèles d'agents et de systèmes multi-agents qui sont explicitement ou implicitement présents dans la plate-forme originelle. Le deuxième niveau se situe purement sur un plan génie logiciel. Il est en effet impératif que la plate-forme hôte soit suffisamment souple et générique dans son implémentation pour que le processus d'intégration ne déroute pas les utilisateurs habituels de la plate-forme cible.

En ce qui concerne les aspects de modélisation, la capacité d'intégration de la plate-forme Madkit repose essentiellement sur une utilisation intensive des principes liés aux approches de type OCMAS, et donc sur l'utilisation du modèle AGR. L'idée fondamentale est de ne poser aucune contrainte sur l'architecture des agents et de se concentrer sur les mécanismes liés à l'utilisation de la structure organisationnelle. Il est donc possible d'implémenter toutes sortes d'architecture interne dans la plate-forme Madkit. De la même manière, aucune contrainte n'est posée sur les modèles de simulation qui peuvent être implémentés grâce aux outils de simulation offerts par la plate-forme (Michel F. *et al.*, 2001). Par exemple, ces outils n'imposent aucune politique d'ordonnancement a priori. Par ailleurs, en dehors de la structure organisationnelle elle-même, Madkit ne repose pas sur un modèle d'environnement particulier et il est donc possible de définir des modèles d'environnement situé quelconques.

En ce qui concerne les aspects implémentatoires, la plate-forme Madkit n'impose aucune contrainte sur la nature des agents qui peuvent être exécutés. Le principe de récupération d'un agent préexistant consiste ainsi à faire en sorte que la nouvelle entité créée ait accès aux fonctionnalités du noyau. Pour cela, il suffit de spécialiser ou d'utiliser directement la classe `Agent` (agent asynchrone) ou `AbstractAgent` fournies dans Madkit. Ces deux classes contiennent l'ensemble des moyens qui permettent d'utiliser la structure organisationnelle et de communiquer à travers elle par envoi de messages. Sur le même principe, toujours grâce à la structure organisationnelle, les outils de simulation offerts par la plate-forme Madkit ne nécessitent pas d'adopter une architecture logicielle prédéfinie. Il ne représente en fait que les moyens de construire un modèle de simulation correspondant aux besoins de l'étude considérée.

Dans ce cadre, l'intégration de Moduleco dans Madkit a nécessité principalement deux étapes :

- la mise en place d'une structure organisationnelle correspondant à la logique de la plate-forme Moduleco : agentification des services et définition de l'organisation sous-tendue par l'architecture de Moduleco.
- la définition des mécanismes de simulation correspondant aux modèles proposés dans Moduleco.

---

<sup>2</sup> [www.madkit.org](http://www.madkit.org)

Il faut noter que ce travail d'intégration a été facilité par certains aspects de l'implémentation originelle de Moduleco. Par exemple, l'implémentation des agents Moduleco est définie par une interface `CAgent`, ce qui a simplifié leur intégration. En outre, préalablement à l'intégration elle-même, un travail conséquent a été réalisé pour séparer le noyau d'exécution de la plate-forme de son interface graphique, suivant le principe Modèle/View/Contrôleur. Moduleco permet notamment la définition de scripts dédiés à l'exécution d'une série de simulations de manière automatique. Cette séparation des différentes fonctionnalités de la plate-forme a grandement facilité son intégration.

### 3. Conclusion

Les premiers résultats de cette intégration sont assez positifs. Premièrement, la plate-forme Moduleco n'a pas eu à souffrir de son intégration dans le sens où les différentes fonctionnalités qui étaient présentes dans la version préexistante ont toutes été récupérées. Deuxièmement, la plate-forme Moduleco a maintenant accès à toutes les fonctionnalités de la plate-forme Madkit. La structure organisationnelle a notamment permis de simplifier et de généraliser de nombreuses parties du code qui étaient associées à la gestion et à la définition de réseaux sociaux. Ceci a notamment permis de remplacer les mécanismes de simulation préexistants, qui étaient peu souples, par un principe de simulation basé sur le réseau social défini dans un modèle particulier. Il est par exemple maintenant possible de définir des simulations comportant des agents hétérogènes jouant différents rôles dans le réseau social. Les principales difficultés qui sont encore actuellement rencontrées sont en majorité liés au passage d'une architecture objet à une programmation purement orientée agent. En effet, alors que Madkit se fonde sur des principes de programmation où le passage par référence des entités existantes est à proscrire, Moduleco a été développée autour d'une architecture objet. Il est ainsi parfois difficile de propager les simplifications dues à l'utilisation de la structure organisationnelle dans le sens où la suppression d'un composant devenu obsolète n'est pas immédiate.

L'intégration étant maintenant presque achevée, l'une des étapes suivantes de ces travaux de recherche consistera à améliorer les aspects de Moduleco qui concernent son utilisation par les utilisateurs finaux. Dans ce cadre, l'utilisation de la structure organisationnelle mise en place sera un atout non négligeable. Elle permettra de clarifier la définition des réseaux sociaux lors de la création d'un modèle de type Moduleco. Il sera notamment très intéressant d'étudier dans quelle mesure la notion de *médium* peut être remplacée par la notion de *groupe* déjà présente dans Madkit.

### 4. Bibliographie

- Phan, D., Beugnard, A, 2001.: Moduleco, a multi-agent modular framework, for the simulation of network effects and population dynamics in social sciences, market & organisations. 8èmes Rencontres Internationales ACSEG (Approches Connexionnistes en Sciences Economiques et de Gestion), Université de Rennes (2001) 307–321
- Gutknecht, O., Ferber, J., Michel, F, 2001, Integrating tools and infrastructures for generic multi-agent systems. In: Proceedings of the fifth international conference on Autonomous agents, AA 2001, ACM Press (2001) 441–448
- Müller, J.P, 2004, The mimosa generic modeling and simulation platform: the case of multi-agent systems. In: proceedings of Agent Based Simulation 5 ABS'2004, Lisbonne, Portugal, SCS Publishing House (2004)
- Phan, D., 2004, From agent-based computational economics towards cognitive economics. In Bourguine, P., Nadal, J.P., eds.: *Cognitive Economics*. Springer-Verlag (2004) 371–398.
- Ferber, J., Gutknecht, O., Michel, F, 2003, From agents to organizations: an organizational view of multi-agent systems. In Paolo Giorgini, Jörg P. Müller, J.O., ed.: *Agent-Oriented Software Engineering IV: 4th International Workshop*, Aose 2003. Lecture notes in computer science LNCS, Melbourne, Australia, Springer Verlag (2003) 185–202
- Michel, F., Ferber, J., Gutknecht, O.: Generic simulation tools based on mas organization. Proceedings of the 10th European Workshop on Modelling Autonomous Agents in a Multi Agent World MAMA'2001, Annecy, France (2001)