

Projet « Intégration – MMM »

Développer un système ouvert et modulaire de modélisation et simulation multi-agents à porté interdisciplinaire, par intégration de solutions pré-existantes

Note à l'intention de Bernard Victorri

auteurs :

Jacques Ferber (LIRMM, Université de Montpellier)

Jean-Pierre Muller (CIRAD & LIRMM, Université de Montpellier)

Denis Phan (ENST de Bretagne , ICI - UBO & Leibniz-imag, Grenoble)

Montpellier, le 28 avril 2004

Référence : note du 29/03/2004 «Fédérer utilisateurs et concepteurs de SMA pour concevoir un système ouvert et modulaire à porté interdisciplinaire ».

Résumé

L'objectif du projet « **intégration-MMM** » est de proposer à l'horizon de six mois une version stable d'un logiciel modulaire et générique fondé sur le paradigme « agent » capable de supporter des plate-formes flexibles de simulation, visualisation et expérimentation dédiés à la modélisation des systèmes complexes. Le moyen choisi pour atteindre cet objectif consiste à utiliser le cadriciel Moduleco pour éprouver selon deux points de vues complémentaires les fonctionnalités offertes par le processus d'intégration de l'environnement multi-agents Madkit et de la méta-plateforme Mimosa.

1 – Positionnement

Les plates-formes « multi-agents » pour la modélisation de la dynamique des systèmes complexes permettent de disposer d'un cadre stable et convivial pour procéder à des expérimentations contrôlées et reproductibles. Elles permettent également de comparer les modèles entre eux pour mieux les évaluer. Il s'agit aussi de fédérer les travaux de recherche et de capitaliser les expériences, en particulier par la mise à disposition de bibliothèques de modèles, utiles pour l'apprentissage des chercheurs jeunes et moins jeunes dans ce domaine. Les plate-formes permettent encore de capitaliser les moyens informatiques disponibles, par exemple : fournir des composants informatiques réutilisables, des modèles de conceptions, des abstractions de comportements utilisés dans tels ou tels disciplines, de faciliter la programmation, par la mise à disposition d'un ensemble de facilités variés selon les compétences du modélisateurs, de guider les concepteurs dans des démarches spécifiques d'approche des systèmes complexes

Il existe déjà en France des avancées significatives dans ce domaine et des outils existent. Il y a même des expériences de coopération avec des utilisateurs dans des niches spécifiques de marché (biologie, écosystèmes et environnements, certains domaines de l'économie etc...) Une fédération de moyens est en cours à travers le groupe de travail « Mimosa » du GDR-I3,

animé par Jean Pierre Müller (*Cirad*). Ce groupe est parvenu au prototypage d'une *méta plate-forme générique de représentation et de simulation de systèmes complexes*, autour de laquelle pourraient venir se greffer des modules spécialisés, à des propositions de standardisation pour l'interopérabilité logicielle et sémantique des modèles et à une amorce de réflexion sur les démarches de modélisation

Le projet « Mimoso », animé par Jean Pierre Müller, se trouve déjà à la confluence d'un certain nombre de projets ou de réalisations existantes compatibles et ou complémentaires qui visent élargir la palette des outils de simulation « individus centrés ». Il s'agit par exemple de la plate-forme « Mobidyc », associant biologistes, informaticiens et biométriciens, de la plate forme « MadKit » de Jacques Ferber, du projet VLE développé par le Laboratoire d'Informatique du Littoral (LIL) du projet SimExplorer développé au LISC-cemagref en association avec l'équipe de « Mobidyc » (Vincent Ginot) et le LIL, ou encore du prototype de plate-forme « Moduleco », co-développée par informaticiens et économistes. Des travaux complémentaires sont également poursuivis autour du thème de la « simulation participative », en particulier par Alexis Drogoul (Université Paris VI – LIP 6) et Nils Ferrand (cemagref). L'ensemble du processus de coopération mis en oeuvre dans Mimoso et les outils correspondants sont présentés en détail dans la note du 29/03/2004 référencée en tête. Nous nous limiterons ici à une brève description des trois entités du projet « integration MMM ».

Une nouvelle initiative fédératrice, accompagnée de financements, apparaît maintenant nécessaire. En effet, comme le note Jean Pierre Müller dans une contribution (annexée à la note du 29/03/2004), les projets actuels devraient conduire à un ensemble d'outils originaux qui auraient des chances d'être reconnus au niveau international si on arrive à les rendre *opérationnels dans un délai court*. En effet, *La concurrence outre-Atlantique* est sinon réellement « nombreuse », du moins en *période de démarrage accélérée* chez les utilisateurs (il s'agit principalement du logiciel **RePast**, développé par l'Université du Michigan, qui se pose en successeur de SWARM). Comme cela est connu en économie industrielle, lorsqu'il y a ainsi compétition entre « clubs » pour des biens à « rendements croissants d'adoption », ce n'est pas la qualité intrinsèque du bien considéré qui est la variable déterminante, c'est le nombre des utilisateurs. Si la qualité intrinsèque de MadKit ou de Cormas, « parents » de Mimoso, n'a rien à envier à celle de Repast, la bibliothèque de modèle de ce dernier logiciel est notablement plus remplie que la totalité des modèles disponibles sur l'ensemble des plates-formes françaises citées plus haut. Un de nos objectifs devrait être de compenser ce handicap, pour éviter de voir marginaliser le produit de plusieurs années de travaux (et de financements).

2 - L'existant

MadKit fournit un *environnement entièrement orienté agent* basé sur le modèle « AGR », c'est-à-dire des *Agents* jouant des *Rôles* dans des *Groupes*. La notion d'agent met en place le cycle de vie d'un agent et lui permet de communiquer avec d'autres agents par envoi de messages. Les rôles structurent les interactions possibles des agents à l'intérieur de groupes. En particulier, les *ordonnanceurs* et les *sondes* sont des agents permettant dans l'état actuel de procéder à des simulations à pas de temps constant, de les visualiser et de traiter les résultats.

<http://www.madkit.org>

Mimoso est une *méta plate-forme de représentation et de simulation de systèmes complexes* basée sur les notions de *composant* (les éléments), de *composé* (les ensembles d'éléments), et de *relation* (entre composants ou composés) pour la partie structurelle, d'*état* et d'*événements* pour la partie dynamique. En particulier, elle permet de décrire le modèle AGR en

introduisant d'une part, des populations (composés) d'agents (composants) s'envoyant des messages (événements) et d'autre part, les groupes (composés) comme ensembles de rôles (composants). Les *relations* entre les populations et les groupes spécifient quels agents jouent quels rôles dans quels groupes. Un ordonnanceur à événement discret implémente la sémantique opérationnelle des descriptions des entités et de leurs dynamique décrites à l'aide de Mimosa. Elle constitue donc *une fusion des modèles de représentation des connaissances et des modèles dynamiques*.

<http://lil.univ-littoral.fr/Mimosa/>

Moduleco est un cadriciel* (framework) c'est à dire une collection d'éléments de conceptions (e.g. patterns - ou patrons de conception) et d'implémentations (e.g. composants logiciels) en coopération et réutilisables qui permettent de créer des applications ou des parties d'applications dans un domaine spécifique (Campos 2000). Il s'agit de fournir à un expert du domaine, éventuellement assisté d'un informaticien, les moyens de concevoir et de mettre en oeuvre un logiciel sans connaître tous les concepts du génie logiciel utilisés pour la conception du cadriciel. Moduleco a été conçu initialement pour étudier les conditions de *prises de décisions d'agents cognitifs en interactions au sein de réseaux*. Cette plate-forme se place à la fois dans une perspective d'économie « computationnelle » dite « Agent Based Computational Economics » et dans une perspective d'économie cognitive (Phan, 2002). Sa principale utilisation actuelle est d'ordre pédagogique (Ecoles thématique CNRS de Porquerolles en 2001, Agay en 2004, cours ENST, ENS). Du fait d'une insuffisance de documentation, son utilisation scientifique reste limitée. Le développement de Moduleco, supervisé par Denis Phan, est animé par Gilles Daniel de l'Université de Manchester :

<http://www2.cs.man.ac.uk/~danielg/moduleco/index.php>

<http://www-eco.enst-bretagne.fr/~phan/moduleco/>

3 - Objectif poursuivi

L'objectif immédiat du projet « intégration-MMM » est de proposer à l'horizon de six mois une version stable d'un logiciel modulaire et générique fondé sur le paradigme « agent » capable de supporter des plate-formes flexibles de simulation, visualisation et expérimentation dédiés à la modélisation des systèmes complexes. Le moyen choisi pour atteindre cet objectif consiste à utiliser le cadriciel Moduleco pour éprouver selon deux points de vue complémentaires les fonctionnalités offertes par le processus d'intégration de l'environnement multi-agents Madkit et de la méta-plateforme Mimosa.

Plus spécifiquement, ce travail d'intégration se traduira parallèlement (1) par une adaptation des composants et des modèles de conceptions utilisés dans Moduleco à la fois aux composants Madkit et aux modèles de conceptions de Mimosa et (2) par une finalisation complémentaire du processus d'intégration de Madkit et de Mimosa. A l'issue du projet « intégration MMM » on s'attend à ce que les modèles déjà développés dans Moduleco soient exécutables avec des agents Madkit et descriptibles par des ontologies exprimées dans le méta-langage de Mimosa.

Du point de vue de la mutualisation des forces, ce travail représente un aboutissement partiel du projet Mimosa, animé par Jean Pierre Müller. La finalisation autour de la classe de modèles proposés par Moduleco d'un logiciel intégrable et distribuable de manière modulaire représente une première étape d'intégration des « couches basses » qui sera complétée par la suite au niveau des « couches hautes » par des développements complémentaires concernant la normalisation XML des modèles, la finalisation du projet SimExplorer pour traiter les

données, ou l'interopérabilité entre l'environnement Mimosa et le système de conception de modèles de biologie des populations Mobidyc. Au delà de l'intégration des couches basses, le projet « intégration MMM », grâce à l'état d'avancement des composantes à intégrer et de l'ébauche d'un processus bilatéral d'intégration (Mimosa MadKit d'une part, MadKit Moduleco d'autre part), permettra en six mois de fournir à l'ensemble intégré Mimosa-MadKit un ensemble de composants utilisateurs et une bibliothèque de modèles susceptibles de servir de base pour tester les résultats d'une partie des projets complémentaires qui viennent d'être cités.

Du point de vue des compétences acquises, l'utilisation de Moduleco comme moyen pour éprouver la capacité de l'intégration Madkit-Mimosa à fournir un environnement générique représente une solution intermédiaire entre l'intégration d'Aquabot de Nils Ferrand (déjà complètement intégré dans l'environnement Madkit) et la récupération de modèles supportés par Repast, un environnement multi-agent proposé aux USA. Elle devrait permettre, dans une étape ultérieure, de s'attaquer à l'intégration de classes de modèles proposés sur Repast.

4 - Mise en oeuvre

La mise en oeuvre du projet « intégration MMM » repose sur une organisation tripolaire qui vient boucler deux processus d'intégration bilatéraux actuellement en cours.

A - Les processus d'intégration en cours

1°) L'intégration de MadKit et de Mimosa se fait sur deux plans :

- *Descriptive* en utilisant Mimosa pour décrire MadKit permettant ainsi potentiellement de *définir de manière déclarative la structure et les fonctionnalités* de MadKit.
- *Opérationnelle* en utilisant le paradigme agent de MadKit pour implémenter la sémantique opérationnelle de Mimosa, ce qui permet sa gestion en termes d'agents pour la distribution, la mobilité et la gestion de l'ensemble de la plate-forme.

2°) L'intégration de MadKit et de Moduleco se déroule actuellement sur un mode adaptatif souple, grâce à la conception modulaire de Madkit 4. Elle permet d'éprouver la modularité de Madkit, en particulier au niveau des ensembles interdépendants et des patrons de conception. Cette intégration permet aussi de mettre en évidence la nécessité de mise en place de certaines procédures d'installation pour gérer la modularité.

- Les principaux composants de Moduleco sont transformés en agents Madkit, étendant ainsi leurs fonctionnalités potentielles, sans cependant transformer de manière drastique les patrons de conception et les fonctionnalités déjà existantes.
- Les composants de Madkit nécessaires au fonctionnement de Moduleco ont été réduits au strict minimum, permettant en particulier à Moduleco de continuer à fonctionner indépendamment de l'environnement Madkit, que Moduleco vient enrichir en tant que « plug-in ».

B - La démarche de mise en oeuvre du projet

La démarche générale se décompose en trois lots d'activité :

1. La réécriture des notions de base de Moduleco (agents, médiums, environnements, etc.) dans MadKit (lui-même en cours d'intégration avec Mimosa) permettra d'obtenir plus rapidement une intégration opérationnelle et donc une possibilité de généralisation des outils de description de la structure des modèles décrits dans Moduleco. Cette réécriture ne sera cependant efficace pour l'intégration globale que si elle est conforme aux principes du méta-langage de Mimosa (2)

2. L'analyse, la conception et l'implémentation directe selon le méta-langage de Mimosa d'un certain nombre de méthodes de description du comportement des agents permettra d'utiliser et de mettre à disposition dans Mimosa ce qui est déjà proposé dans MadKit et de mettre à disposition du modélisateur des concepts plus proches de ses formulations théoriques ;
3. La réification dans Mimosa des objets pour la visualisation (graphique), le support de la démarche expérimentale (procédure de description et d'initialisation et d'exécution des simulations), du traitement des données issues de la simulation, permettra à l'utilisateur-modélisateur non seulement de décrire son modèle mais aussi de décrire comment il veut le construire, comment il veut le visualiser et comment il veut l'explorer par la simulation de la même façon que le modèle lui-même.

Au total, l'ensemble de ces trois lots a l'ambition de réaliser et valider une plateforme capable de couvrir toute la chaîne de conception d'un modèle dans son aspect structurel et dynamique ainsi qu'une partie de son exploitation par la simulation.

Références

4.0 - Documents contributifs : (Annexés)

contribution de J.P. Müller (Mimosa)

contribution de D. Phan (Moduleco)

contribution de J. Ferber (MadKit)

proposition complémentaire (Aquabot) de Nils Ferrand :

<http://www.montpellier.cemagref.fr/ppp/nils-ferrand/projets/mimosa/presentmimosa.pdf>

4.1. Projets informatiques

<http://lil.univ-littoral.fr/Mimosa/>

http://www.lisc.clermont.cemagref.fr/Labo/activite_recherche/projets/Projets_en_cours/SimExplorer/SimExplorer.htm

4.2. projets 2003 AC systèmes complexes pour SHS en relation directe avec le projet

Les deux projets suivants sont directement concernés, car des membres de Mimosa participent activement au projet :

<http://www.lps.ens.fr/%7Eweisbuch/scshs/projets2003/Berestycki.pdf>

<http://www.lps.ens.fr/~weisbuch/scshs/projets2003/ferrand.pdf>

4.3 – Autres Plate-formes françaises concernées

Cormas (JP Müller, Cirad)

<http://cormas.cirad.fr/>

Mobidyc (V. Ginot, INRA département de biométrie, unité de statistiques spatiales d'Avignon.)

http://www.avignon.inra.fr/internet/unites/biometrie/mobidyc_projet/version_index_html

<http://www.avignon.inra.fr/mobidyc>

un concurrent américain : <http://repast.sourceforge.net/>

L'initiative MIMOSA

Coordinateur : Jean-Pierre Müller
CIRAD-TERA-REV-GREEN, jean-pierre.muller@cirad.fr
(<http://lil.univ-littoral.fr/Mimosa/>)

Résumé

L'interdisciplinarité, notamment entre les sciences humaines et sociales et les sciences de la nature, se construit autour d'un objet commun relativement à une question. Nous proposons ici les éco-sociosystèmes relativement au problème de leur gestion. La complexité des éco-sociosystèmes fait ressortir un objet d'étude à caractère transdisciplinaire, à savoir les systèmes complexes et leur compréhension. Dès lors les sciences humaines et sociales doivent être convoquées sur deux plans, en tant que contributeurs de connaissances sur ces systèmes complexes eux-mêmes et en tant que contributeurs de connaissances sur les processus de co-construction d'une compréhension de ces systèmes. L'initiative Mimosa s'inscrit dans ce cadre pour *proposer un objet médiateur : une plateforme générique de modélisation et de simulation participative afin de faire dialoguer les disciplines et les acteurs*. Le résultat attendu est *une plateforme originale par sa structure et ses concepts et qui a ses chances d'être reconnue internationalement si on arrive à la rendre opérationnelle dans un délai court*. Les concurrents outre-atlantique sont nombreux.

Problématique

La gestion de système complexes que ce soit des systèmes irrigués, des bassins versants, des terroirs, des provinces, etc... c'est-à-dire des éco-sociosystèmes caractérisés par l'interactions de dynamiques biophysiques et sociales diversifiées et changeantes, et ceci dans un contexte de décentralisation de la décision, requiert des nouveaux outils de représentation et de simulation pour aider à une compréhension partagée, qu'elle soit rétrospective, actuelle ou prospective. La caractéristique principale de ces outils est le besoin de prendre en compte la diversité des regards que les disciplines scientifiques mais aussi les acteurs portent sur leur territoire à travers, à la fois la diversité de leur perspective (écologique, économique, sociale) de leurs intérêts (préservation, rentabilité, durabilité, bien-être individuels et sociaux) et la diversité des niveaux d'intervention et de décision (parcelle, communauté, village, district, bassin, etc.). La diversité des points de vue est accompagnée par la diversité des moyens de représentations que les disciplines (écologues, géographes, économistes,...) et les acteurs se donnent pour comprendre, visualiser et s'aider dans les processus de décision. Aussi un outil au service de l'interdisciplinarité et de la communication avec des acteurs multiples avec ou non une culture scientifique se doit de pouvoir permettre l'expression de cette diversité et d'offrir la possibilité de leurs articulations.

Objectif

L'objectif du projet MIMOSA est :

- de faire l'inventaire des moyens de représentation utilisés par les modélisateurs dans un premier temps et par les acteurs dans un second afin de bâtir de façon incrémentale une ontologie des formalismes ;
- de définir les dénominateurs communs à ces formalismes afin de leur permettre de communiquer les uns avec les autres dans des modèles intégrés, c'est-à-dire d'assurer une interopérabilité logicielle ;

- de définir les ontologies de base permettant de décrire les modèles indépendamment de leur moyens d'implémentation, c'est-à-dire d'assurer une interopérabilité des plateformes;
- d'utiliser ces ontologies de base pour définir des modèles communicables, réutilisables et articulables les uns aux autres, c'est-à-dire d'assurer une interopérabilité sémantique ;
- d'ajouter aux moyens de représentation formels des moyens de représentation graphiques permettant de prendre en compte non seulement la diversité des formalismes et des points de vue mais aussi la diversité des façons de se donner à voir ce que l'on décrit ainsi que les simulations qui en résultent.

Enfin il s'agit d'explorer et de valider le vaste champs ainsi ouvert par la réalisation d'une plateforme générique implémentant cet ensemble de propositions et l'utilisation de cette plateforme dans des contextes variés de gestion de l'environnement, du territoire, des filières ou des ressources naturelles.

Bref état de l'art

On peut ranger les plateformes de simulation dans deux catégories :

- Les plateformes génériques ne faisant pas d'hypothèse sur les types de modèles en fournissant l'accès au niveau des langages de programmation ;
- Les plateformes dédiées à un formalisme ou ensemble de concepts à disposition.

Dans la première catégorie, on peut citer Swarm développé au Santa-Fe Institute, écrit en Objective-C et maintenant en Java avec une très grande librairie d'utilitaires due à une communauté forte d'utilisateurs mais difficilement utilisable par des thématiciens. Repast, à la suite de Ascape, est un environnement « swarm-like » écrit en Java qui hérite donc de cette communauté. Dans la même direction, on peut citer Mason, ou Moduleco. Oris fournit un langage proche du C++ et est dédié à la réalité virtuelle. Elle présente l'intérêt de pouvoir « immerger » le modélisateur dans son modèle.

Dans la seconde catégorie, on a StarLogo qui possède un langage dédié à la simulation à pas de temps discret et a quelques successeurs comme NetLogo et Breve pour la simulation 3D. Cormas, développé par le CIRAD, définit les briques de base que sont les automates cellulaires, les agents, les interactions et les agrégats spatiaux ou sociaux mais dont les comportements sont écrits en Smalltalk. D'autres systèmes de simulation existent, basés sur les modèles à compartiments comme dans Stella et SME ou DEVS pour la multi-modélisation.

MadKit est une plateforme générique écrit en Java utilisée à la fois pour la recherche en SMA en général. Il dispose d'un environnement de simulation à pas constant similaire à StarLogo. Il dispose aussi d'autres modules de simulation (Warbot, etc.. voir annexe). Il se situe donc à cheval sur les deux catégories.

La multi-modélisation est reconnue et traitée dans des systèmes comme SME, mais avec un seul type de modèles ou comme DEVS à travers une encapsulation standardisée (le DEVS-bus). Les autres plateformes n'offrent pas de telles possibilités. La plupart des plateformes offrant des moyens d'expression au niveau d'un thématicien ou d'un modélisateur sont construites sur un formalisme spécifique et unique. Finalement aucune plateforme à ce jour ne couple les représentations des dynamiques avec les représentations des connaissances ce qui est indispensable dès lors qu'une partie des informations sont données par des modèles conceptuels (ontologies, etc.), des bases de données ou des SIG.

La plate-forme Mimosa

La plate-forme Mimosa se compose d'un certain nombre de couches :

- Couche 0 : c'est la couche fonctionnelle définissant les moyens de communication et de simulation causalement cohérente d'un ensemble de modèles dynamiques que ce soit sur une machine ou en distribué. Tout modèle dynamique doit avoir un état, des paramètres, des événements d'entrée et de sortie, des observables et une façon de mesurer le temps ;
- Couche 1 : c'est la couche structurelle encapsulant les modèles dynamiques dans des objets élémentaires à la base de n'importe quel formalisme : tout formalisme de modélisation repose sur la description d'élément, d'ensemble d'éléments et de relations entre éléments ou entre ensembles. Tout élément, ensemble ou relation est un modèle dynamique dont l'état peut changer au cours du temps ;
- Couche 2 : elle est constituée de « plugins » décrivant chacun un formalisme particulier par la description des éléments, ensembles et relations spécifiques (états, automates et transitions pour les automates, lieux, espaces et relations spatiales pour les descriptions spatiales, etc.). Un formalisme peut décrire soit des structures (spatiales, temporelles, conceptuelles, sociales), soit des dynamiques (automates, state-charts, réseaux de Petri, modèles à compartiments,...) ;
- Couche 3 : elle est constituée de tous les outils de visualisation et d'édition graphique des modèles exprimés grâce aux formalismes mis à disposition par la couche 2. On y inclut aussi les moyens de description des plans d'expérience et de post-traitement des simulations, notamment les outils statistiques.

L'ensemble de ces couches permet ensuite aux modélisateurs et thématiciens de décrire leurs modèles et de les visualiser comme ils le souhaitent en décrivant les ontologies spécifiques à leur domaine (ensemble des concepts qu'ils manipulent et la façon dont ils les visualisent) et en instanciant ces ontologies dans des modèles concrets qu'ils peuvent ensuite simuler. Il n'est pas nécessaire à ce niveau d'accéder à la couche de programmation.

La modélisation participative

La modélisation participative peut se voir à trois niveaux :

- Au premier niveau, il s'agit d'animer la co-construction de la plate-forme Mimosa par l'ensemble des développeurs intéressés. Il est envisager pour cela d'animer une communauté de développeur dans la philosophie des logiciels libres (Open source) par l'utilisation de SourceForge. *Cela nécessite cependant au moins un animateur à plein temps pour intégrer les apports des partenaires.* L'avantage attendu est le partage de l'effort de réalisation des outils par les personnes intéressées à implémenter des formalismes spécifiques et les outils de visualisation, de pré- et de post-traitement ;
- Au deuxième niveau, il s'agit de réfléchir à *l'utilisation de cette plateforme comme collectif de conception de modèles et de leurs articulations dans des simulations intégrées.* Le but est de pouvoir échanger des modèles, de les faire communiquer permettant ainsi d'exprimer la multiplicité des points de vue puis leurs articulations, éventuellement la co-construction d'un modèle partagé ;
- Au troisième niveau, il s'agit de *permettre à des acteurs d'interagir avec et à travers des modèles existants.* Cette immersion peut servir à s'approprier un modèle en explorant comme il réagit aux sollicitations. Elle permet aussi d'observer les dynamiques sociales en interactions avec un modèle biophysique ou un autre modèle

social et d'aider ainsi à l'explicitation des connaissances sur les stratégies des acteurs (Cf. AS-CPSOA animé par A. Drogoul).

Si la première possibilité est relativement bien connue dans la communauté du génie logiciel, *les deux dernières possibilités sont entièrement nouvelles*, et davantage encore si on envisage l'utilisation de ce genre d'approche avec des acteurs n'ayant pas forcément une culture scientifique. Or ce dernier point est essentiel si on veut assurer une utilité sociale de ce genre d'outils dans les processus de décision, eux-mêmes participatifs, voire citoyens où tout un chacun peut via Internet ou son téléphone portable s'approprier le problème de décision (et donc le système sur lequel il porte) et y ajouter ses propres connaissances et suggestions.

Mimosa et SHS

L'interaction entre l'initiative Mimosa et les sciences humaines et sociales se fait à deux niveaux :

- D'une part, l'ambition de Mimosa est de faire dialoguer les disciplines, les sciences de la nature, certes, mais aussi les sciences humaines et notamment l'anthropologie, la sociologie, la géographie et l'économie. Si cette dernière est déjà bien rompue à la modélisation (Cf. travaux de D. Phan) ainsi que la géographie à travers les SIGs, les deux premières commencent seulement à s'y essayer en tout cas dans le monde francophone (Cf. GT-ARP animé par N. Ferrand) ;
- D'autre part, Mimosa en voulant offrir un support aux processus d'explicitation des connaissances, d'auto-apprentissage, de co-construction de représentations partagées et de décisions à besoin des sciences humaines pour analyser les interactions entre ces interactions et ce support et comprendre en quoi et comment un tel support peut améliorer la compréhension réciproque entre les acteurs ainsi que leurs décisions. Ainsi la réalisation technologique ne se passera pas d'analyses et de validation anthropologiques et sociologiques.

Les partenariats

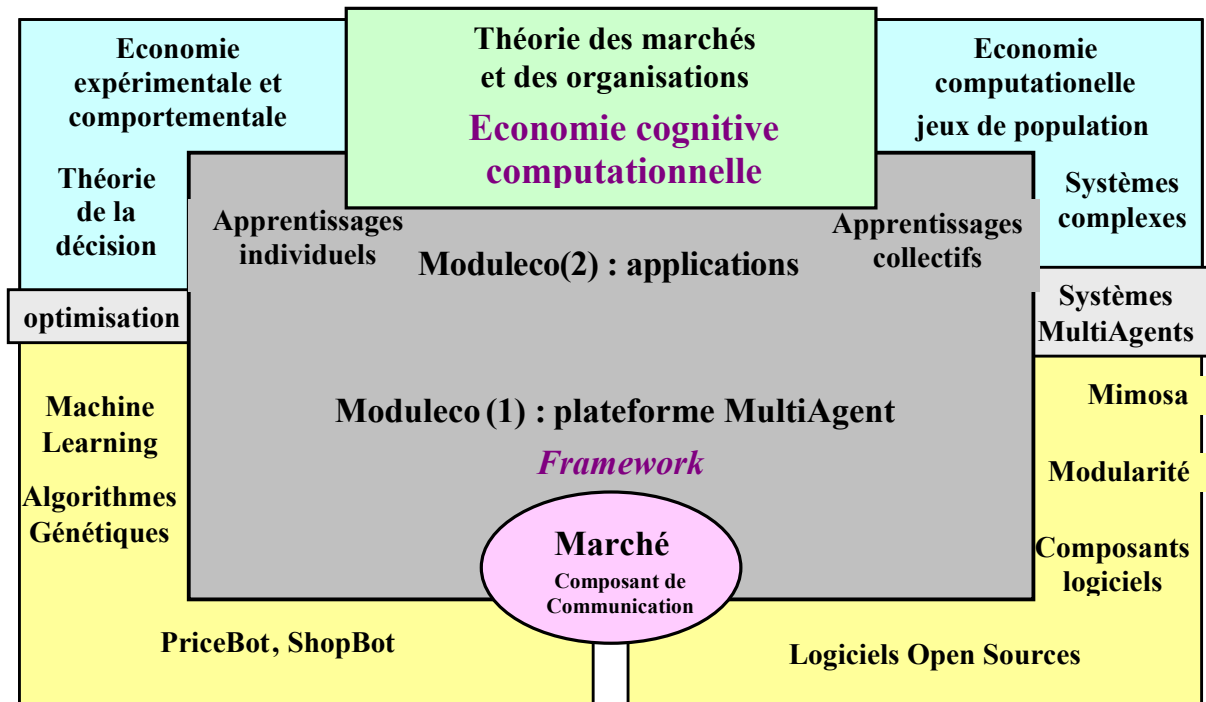
Les laboratoires suivants, impliqués pour la plupart dans la modélisation et simulation individu-centrées et multi-agents, participent à cette initiative :

- CEMAGREF-Irri Montpellier
- CEMAGREF-Lisc Clermont-ferrand
- CIRAD-Amis-Amap Montpellier
- CIRAD-Tera-Rev-Green Montpellier
- EMSE St Etienne
- ENIB Brest
- ENST-Bretagne Brest
- EPFL-Lasen Lausanne
- INRA-Avignon Avignon
- IRD Dakar
- IRD-Geodes Bondy
- IREMIA St Denis de la Réunion
- LIFL Lille
- LIL Calais
- LIMOS Clermont-Ferrand
- LINA Nantes
- LIP6 Paris
- LIRMM Montpellier

Moduleco

Moduleco est un prototype de plate-forme « multi-agents », dont les applications (les modèles) se situent à l'intersection de plusieurs disciplines et de plusieurs point de vues. Ce prototype a été conçu pour simuler les marchés et les organisations, les phénomènes sociaux et la dynamique des populations. Il s'agit donc d'étudier les conditions de *prises de décisions d'agents cognitifs en interactions au sein de réseaux* (projet « Choix individuels et interactions multi-agents dans les réseaux »* financé par le Groupe des Ecoles des Télécommunications, projet « Marché virtuel », financé par France Télécom R&D). Cette approche s'appelle de *l'économie cognitive computationnelle* (Phan, 2004a)

Figure 1 : Moduleco, une plate-forme trans-disciplinaire



1 - Le positionnement de Moduleco relativement aux approches existantes

Dans le haut de la figure 4, on mentionne les approches économiques et non économiques qui ont été prises en compte au niveau de la conception de la plate-forme. Une finalité de Moduleco est *d'intégrer la dimension individuelle et les dimensions collectives*.

Pour une première approche, on retiendra *l'économie expérimentale et comportementale* (Camerer, 2003) et la *théorie de la décision* comme deux démarches représentatives du programme « *cognitif* » de l'économie cognitive (Walliser, 2000). Bien entendu, ces deux démarches prennent en compte le collectif (par exemple la théorie de la décision interactive), mais elles sont orientées vers le comportement du décideur. Pour l'instant, seuls des travaux de Benoit Leloup (2002a, 2002b) ont utilisés Moduleco selon cette perspective. D'autres développement devraient cependant être présentés en 2004, à la suite de Phan (2004b)

Quand on aborde la dimension collective, l'économiste a à l'esprit le programme « *évolutionniste* » de l'économie cognitive au sens de Walliser (2000). C'est selon cette dimension, paradoxalement moins familière aux économistes, qu'ont été développés une majorité des modèles de Moduleco que ce soit au niveau de la recherche (Boitout, Delahaut, 2003, Phan 2004a, Phan, Pajot, Nadal 2003, Nadal, Phan, Godon, Vanimendus, 2003) ou au niveau de la démonstration, à travers des modèles bien connus (Phan, Beugnard, 2001 a & b, Phan 2004a)

Beaucoup plus « jeune » que Cormas et Mobidic (les premiers développements datent de 2000), elle souffre de quelques défauts de conception et de documentation (à la fois faute de temps et de moyens) qui devraient se résorber dans les mois à venir. Cette plate-forme se place à la fois dans une perspective d'économie « computationnelle » dite « Agent Based Computational Economics » et dans une perspective d'économie cognitive (Phan, 2002 « From Agent-Based Computational Economics towards Cognitive Economics » in Bourguin, Nadal eds. (2004) *Cognitive Economics : An Interdisciplinary Approach* ; Springer Verlag : version longue : <http://digemer.enst-bretagne.fr/~phan/papers/ACEPhan.pdf>) A ce titre, une première présentation en a été faite à l'Ecole thématique CNRS de Berder en Economie Cognitive (mai 2000), puis elle a servi à l'animation d'un atelier à celle de Porquerolles (septembre octobre 2001). Son développement a été limité à la fois par le manque de moyen financier et informatique et par le manque corrélatif d'utilisateurs, du à l'absence complète de documentation, en particulier. Son utilisation scientifique s'est donc limitée pour l'instant à son co-créateur, en association avec des physiciens (JP Nadal, M.B. Gordon pour Nadal et al. 2003, Phan et al. 2003) et à une thèse et des articles correspondants (B. Leloup, 2002), ainsi qu'à un contrat industriel avec FT R&D, sur le management des communautés virtuelles. Par contre, elle a fait l'objet d'une utilisation en cours d'introduction aux systèmes complexes, que ce soit à l'ENST de Bretagne, à l'ENS (Jean Pierre Nadal, DEA de sciences cognitive) ou encore dernièrement à l'Ecole thématique CNRS d'Agay : Dynamique des systèmes complexes et applications aux SHS, citée plus haut.

2. - Perspectives nouvelles et développements futurs

Si son utilisation est restée jusqu'ici assez confidentielle, des *perspectives nouvelles* sont apparues récemment. Un doctorant en Informatique de l'Université de Manchester, Gilles Daniel l'a adoptée comme « coeur » de sa thèse sur *les applications de la simulation multi-agents aux marchés financiers*. Il maintient sur Internet un site de développement en ligne pour Moduleco. L'étude des modalités de l'intégration avec Madkit5 (Jacques Ferber), en particulier l'architecture « agent groupe rôle » est programmée pour fin avril, dans le cadre du projet ELICCIR de l'AC « systèmes complexes pour SHS ». Dans ce même cadre, des réflexions sont menées conjointement avec Jean Louis Dessalles (ENST) sur les aspects cognitifs. Finalement, pour mettre l'accent sur les applications et la dimension pédagogique qui reste le point fort de Moduleco selon Leigh Tesfatsion (spécialiste américaine du multi-agents, intéressée au développement de RePast).

Un groupe de travail composé d'économistes, de Gestionnaires et d'informaticiens devrait se réunir à Brest début juillet pour lancer *un ouvrage pédagogique sur les applications multi-agents à l'économie et à la gestion*. Dans le cadre de la « remise à plat » de l'architecture logicielle de Moduleco, des synergies sont également recherchées avec le projet « SimExplorer » du LISC (les membres du LICS participent au volet « multi-agent » du projet ELICCIR). A terme, nous cherchons à maintenir un logiciel « léger » accessible principalement aux chercheurs sachant déjà programmer (nombreux en économie), compatible avec Mimosa et pouvant également servir d'interface pédagogique pour présenter au non programmeurs certaines fonctionnalités de ce dernier système, dans le cadre de cours ou d'ateliers.

Références

Moduleco & ACE

Phan D., Beugnard A., (2001a) "Moduleco, a modular multi-agent platform, designed for to simulate markets and organizations, social phenomenons and population dynamics" ; CD Rom, Ecole CNRS d'Economie Cognitive, Porquerolles, 25 Septembre - 5 octobre.

<http://www-eco.enst-bretagne.fr/~phan/moduleco/english/moduleco00.htm>

Phan D., Beugnard A., (2001b) "Moduleco, a multi-agent modular framework, for the simulation of network effects and population dynamics in social sciences, market & organisations" ; 8° Rencontre Internationale ACSEG (Approches Connexionnistes en Sciences Economiques et de Gestion), Université de Rennes, IGR 22-23 novembre.

Phan D. (2004a) "From Agent-Based Computational Economics towards Cognitive Economics" in Bourguine P., Nadal J.P. eds. (2004) Cognitive Economics ; Springer Verlag, p. 371-398.

Phan D. (2004b) Hierarchy of cognitive interactive agents and statistical mechanics : how Object Oriented Programming highlights the connection, Agent Based Simulation 5, Lisbon, Portugal, 3-5 May.

Modèles :

Boitout N., Delahaut T (2003) Long range dependence in a real trading time agent-based model

Daniel G. (2004) Random time in Agent-Based Market Models Communication aux journées de Rochebrune

Leloup B. (2002b) Dynamic Pricing with Local Interactions – Logistic Priors and Agent Technology » Proceedings of the 2002 International Conference on Internet Computing & International Conference on artificial Intelligence, June, Las Vegas, USA.

Leloup B. (2002b) *L'incertitude de deuxième ordre en économie : le compromis « exploration exploitation*, thèse de doctorat d'économie inédite ENS Cachan

Phan D., (2003) "Small Worlds and Phase Transition in Agent Based Models with Binary Choices" in: Muller J.P., Seidel M.M. eds. 4° workshop on Agent-Based Simulation, Montpellier.

Nadal J.P., Phan D., Gordon M.B, Vannimenus J. (2003), "Monopoly Market with Externality: an Analysis with Statistical Physics and ACE", 8th Annual Workshop on Economics with Heterogeneous Interacting Agents, Kiel, May 29-31 (full paper) also available at the physicist's base : <http://arxiv.org/abs/cond-mat/0311096>

Phan D., Pajot S. Nadal J.P. (2003) "The Monopolist's Market with Discrete Choices and Network Externality Revisited: Small-Worlds, Phase Transition and Avalanches in an ACE Framework" ; Ninth annual meeting of the Society of Computational Economics University of Washington, Seattle, USA, July 11 - 13, 2003.

Moduleco (D. Phan, ENST de Bretagne, G. Daniel, Université de Manchester, DC)

informations générales et présentations : <http://www-eco.enst-bretagne.fr/~phan/moduleco/>

développement et download : <http://www2.cs.man.ac.uk/~danielg/moduleco/index.php>

Notoriété : référencement sur le site de Leigh Tesfatsion

« General Software and Toolkits Agent-Based Computational Economics (ACE) and Complex Adaptive Systems (CAS) » : <http://www.econ.iastate.edu/tesfatsi/acecode.htm>

MadKit

MadKit est une plate-forme d'exécution de systèmes multiagents basée sur le modèle organisationnel agent/groupe/rôle. MadKit fonctionne de manière distribuée selon un mode 'pair à pair'. MadKit introduit aussi la notion de communauté, c'est-à-dire d'application distribuée à laquelle un ensemble d'agents peuvent se réunir et partager des applications et des documents. Il est ainsi possible de réaliser facilement de nombreuses applications distribuées: chat, partage de documents, jeux distribués, etc..

MadKit est conçu pour supporter des architecture d'agents hétérogènes (agents cognitifs ou réactifs), des modèles de communications distincts (types de messages et protocoles hétérogènes), des environnements d'exécution variés (applications stand-alone, applet, module intégré à une application hôte, etc..). MadKit est utilisé actuellement dans des contextes applicatifs très variés qui vont des applications distribuées en mode peer to peer, à la simulation de robotique collective, en passant par la gestion de connaissance, l'administration de réseaux, les recherches en architectures hybrides pour le contrôle de robots, le développement de jeux distribués et des simulations multiagent de type « vie artificielle ».

MadKit est utilisé internationalement dans de nombreux pays (Etats-Unis, Europe, pays du Maghreb, Chine, Japon, etc...). Il est particulièrement utilisé dans le domaine de l'enseignement des systèmes multiagents, grâce à ses interfaces utilisateurs et sa grande souplesse d'utilisation.

Il a été mis en ligne fin 1999 sur le site www.madkit.org, et à ce jour plus de 8000 téléchargements ont été effectués. Plusieurs applications industrielles (jeu distribué, gestion de performances de réseaux, logiciel collaboratif, logiciel d'interaction artistique, etc.) ont été réalisées à partir de MadKit.

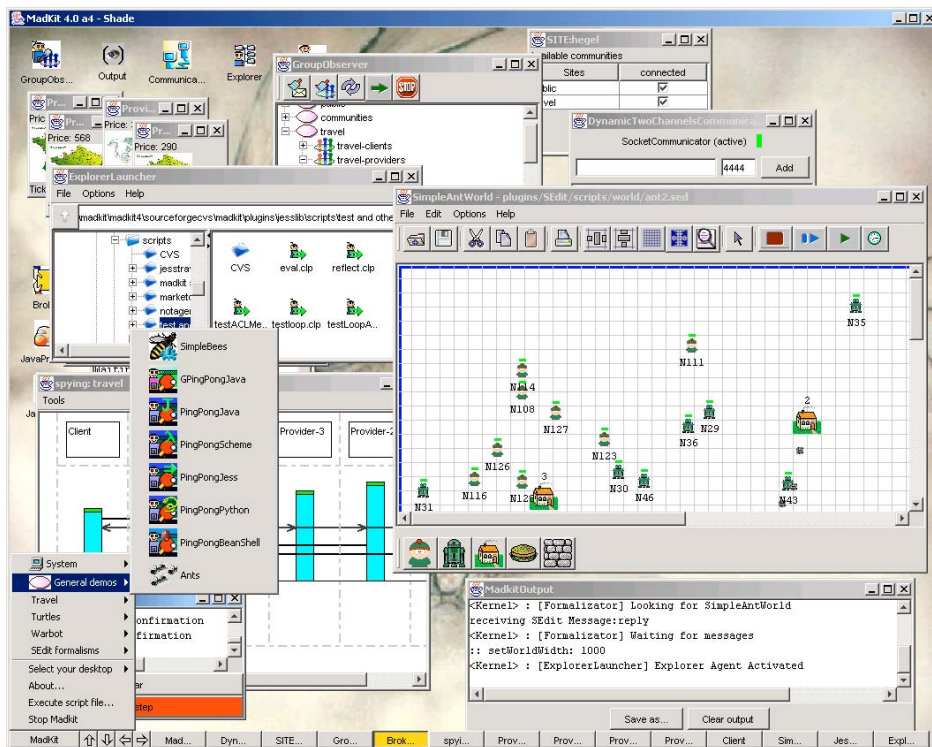


Figure 1: vue générale du "desktop" de MadKit

MadKit est distribué avec une licence LGPL pour toutes les parties fondamentales et GPL pour tout ce qui a trait aux interfaces graphiques et au développement.

Le modèle AGR

Le modèle AGR est fondé sur trois composantes: les agents, les groupes et les rôles, qui sont définis les uns par rapport aux autres. L'*agent* est simplement décrit comme une entité autonome communicante qui joue des *rôles* au sein de différents *groupes*. Un agent peut être membre de plusieurs groupes. Un rôle est une représentation abstraite d'une fonction, d'une position sociale occupée par un agent dans un groupe.

Le modèle AGR, qui est très général, s'est montré particulièrement puissant pour définir des systèmes multiagent organisationnels. Des extensions de ce modèle vers l'intégration de normes sociales est en cours de développement.

Architecture générale

L'architecture de MadKit est basée sur un noyau minimal qui fait moins de 100koctets et qui n'assure que les fonctions primitives (communications locales, gestion du cycle de vie d'un agent, gestion des groupes/rôles locaux). Les services de plus haut niveau (service de communication à distance, gestion des groupes distribués, migration des agents, visualisation et contrôle du fonctionnement de la plate-forme, etc.) sont fournis sous la forme d'agents pour offrir un maximum de flexibilité : il est possible d'ajouter et/ou de modifier ces services sans avoir à toucher à la plate-forme. Par exemple, il est possible de passer d'une communication « socket » à une communication « http » par un simple remplacement d'un agent par un autre.

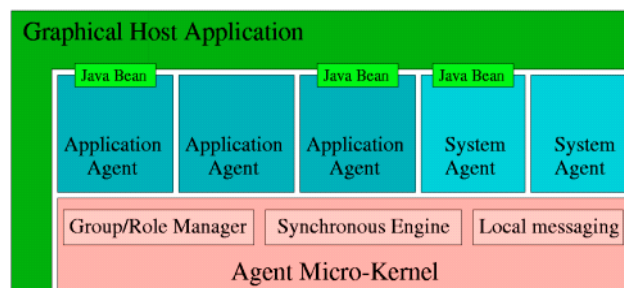


Figure 1. L'architecture générale de MadKit

MadKit n'impose aucune architecture spécifique aux agents : le modèle des agents est volontairement resté simple et générique pour faciliter l'intégration de différents modèles d'agents. Bien que MadKit soit développé en Java, il est possible de programmer des agents dans plusieurs langages: outre Java, on peut définir ses agents en Scheme, Jess (langage à base de règles semblable à CLIPS), BeanShell (Java interprété), ou Python. Tous ces agents communiquent ensemble sans aucun problème et partagent la méthodologie organisationnelle fondée sur les groupes et les rôles.

D'autre part, à partir de la version 4.0, MadKit se présente sous une forme totalement modulaire, ou plus exactement sous la forme d'un ensemble de plugins. Il est ainsi facile d'ajouter de nouvelles fonctionnalités à MadKit par l'intermédiaire de nouveaux plugins. Il est aussi possible de construire des versions allégées de MadKit en n'utilisant que les plugins désirés.

L'interface graphique, fondée sur un modèle componentiel (java beans) totalement découplé des agents, permet de définir son propre look & feel et ses propres interfaces pour chaque type d'usage. Il est ainsi possible d'intégrer facilement un SMA à l'intérieur d'une application hôte. Les interfaces étant indépendantes du noyau, il est ainsi possible de passer d'une phase de développement (en utilisant le desktop, l'environnement d'exécution qui est

montré figure 1) à une phase d'utilisation finale, sans avoir modifier le code des agents où leur représentation graphique.

Simulations sous MadKit

MadKit contient de nombreux outils permettant de développer facilement des simulations multiagents. Il comprend un moteur de simulation de système multiagent par pas constant auquel nous avons récemment ajouté un système de simulation par événements discret.

MadKit dispose notamment d'un environnement de simulation pour agents réactifs TurtleKit qui reprend les fonctionnalités générales de StarLogo et les étend avec de nombreuses fonctionnalités supplémentaires. Il est ainsi possible de réaliser facilement des applications de type "vie artificielle" dans lesquelles des agents réactifs simples coordonnent leur action par l'intermédiaire d'un milieu support des actions et des traces des agents.

Un environnement de tests pour évaluer des stratégies de coopération est fourni par Warbot. Warbot, qui utilise le module SEdit de MadKit (cf. ci-dessous) se présente comme deux équipes de robots simulés qui doivent détruire les bases du camp adverse. Pour cela, chaque joueur peut programmer les "têtes" des agents de son équipe. Warbot a déjà été testé plusieurs années dans le cadre du DEA d'informatique de Montpellier.

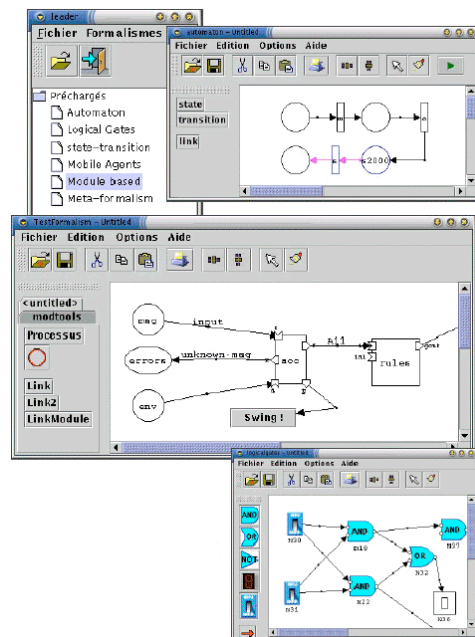
A partir de Warbot, un environnement de simulation orienté vers la gestion de l'eau dans les cultures, Aquabot, a été développé en collaboration avec le Cemagref.

Plusieurs environnements de simulation orientés vers la simulation de robots ont été développés sur MadKit.

SEdit

SEdit est un outil de manipulation et d'animation de diagrammes développé sous forme d'un SMA et fonctionnant comme un plugin MadKit. Il permet d'écrire et d'animer rapidement tout formalisme graphique représentable sous forme de graphe en XML. A l'aide de quelques classes optionnelles, il peut passer d'un simple visualiseur à un simulateur pour certains types de structures. Son intégration à la plate-forme permet de mener de front construction et exécution en place d'agents.

SEdit fournit ainsi un ensemble d'outils graphiques pour les nombreux modèles qui apparaissent lors de la conception et le développement de SMAs. Les formalismes, c'est-à-dire les types de diagrammes, sont définis en XML. Cette description est interprétée par SEdit pour construire automatiquement un éditeur de ce type de formalisme. Ces éditeurs peuvent être étendus par des classes Java pour définir des formalismes actifs (réseaux de Petri simulés, génération de code à partir de diagrammes de classes, etc.). Certains de ces formalismes peuvent être considérés comme des langages de description de comportement d'agents (langage BRIC) et être exécutés à l'intérieur de MadKit, donnant ainsi la possibilité à un développeur de construire des agents sous forme graphique. De plus, comme SEdit est lui-même une application MadKit, les éditeurs de diagrammes, qui sont simplement des agents spécialisés, sont totalement intégrés à l'environnement de développement et peuvent être intégrés à une application MadKit quelconque.



Références sur MadKit (et AGR)

Site: www.madkit.org

Articles:

Ferber J., Gutknecht O., Michel F., *From Agents to Organizations : an Organizational View of MultiAgent Systems*, in Agent-Oriented Software Engineering (AOSE) IV, P. Giorgini, Jörg Müller, James Odell, eds, Melbourne, July 2003, LNCS 2935, pp. 214-230, 2004.

F. Michel, G. Beurier, A. Gouaïch and J. Ferber, *The TurtleKit platform : application to multi-level emergence*, (Poster), Agent Based Simulation 4 ABS4, Montpellier, 28 - 30 Avril 2003.

Ferber J., Michel F., Gutknecht O., *Agent/Group/Roles : Simulating with organizations. Agent Based Simulation* Agent Based Simulation 4, Montpellier, 28 - 30 Avril 2003.

Ferber, J. and Gutknecht, O. *Operational Semantics of a Role-Based Agent Architecture* (best paper award). Agent Theories, Architectures and Languages (ATAL 99), N. Jennings and Y. Lespérance Eds, Orlando, Springer-Verlag, LNAI, 1757, pp. 205-217, 2000.

Ferber, J. et Gutknecht, O. *Pour une sémantique opérationnelle des systèmes multi-agents*. 8ème Journées Francophones sur l'Intelligence Artificielle Distribuée et les Systèmes Multi-Agents (JFIADSMA'2000), La Réunion, Hermès, pp. 39-56, 2000.

Ferber J., Gutknecht O. Aalaadin: *a meta-model for the analysis and design of organizations in multi-agent systems*, ICMAS 98 (International Conference on Multi-Agent Systems), Paris, Y. Demazeau (ed), IEEE Press, pp. 128-135, 1998.

Gutknecht, O. and Ferber, J. *Madkit: a Generic Multi-Agent Platform*. Autonomous Agents (AGENTS 2000), Barcelona, ACM Press, pp. 78-79, 2000.

Gutknecht, O., Ferber, J. and Michel, F. *Madkit: une expérience d'architecture de plate-forme multi-agent générique*. 8ème Journées Francophones sur l'Intelligence Artificielle Distribuée et les Systèmes Multi-Agents (JFIADSMA'2000), La Réunion, Hermès, pp. 223-236, 2000.

Gutknecht, O. and Ferber, J. *Vers une méthodologie organisationnelle de conception de systèmes multi-agents*. JFIADSMA'99, La Réunion, Hermès, pp. 93-104, 1999.

Gutknecht, O. and Ferber, J. *Un méta-modèle organisationnel pour l'analyse, la conception et l'exécution de systèmes multi-agents*. JFIADSMA'98, Nancy, Hermès, pp. 267-280, 1998.

Michel, F., Ferber, J. and Gutknecht, O. *Generic Simulation Tools based on MAS organization*. MAAMAW 2001, Annecy, Y. Demazeau and F. J. Garijo, eds., INPG Publications, Annecy, France, 2-4 Mai 2001.

Simonin O, Michel J., Chapelle J., and Ferber J., *Un simulateur de systèmes multi-robots dans MadKit*, in Xèmes journées Francophones pour l'Intelligence Artificielle Distribuée et les Systèmes Multi-Agents, JFIADSMA'02, (Lille France), 28-30 Octobre 2002

Simonin, O., Coulondre, S. and Ferber, J. *Un modèle de représentation et d'analyse des systèmes multi-agents*. JFIADSMA'99, La Réunion, Hermès, pp. 81-92, 1999.