

Conception de systèmes d'information à référence spatiale (SIRS) communicants pour les agrosystèmes

Vers une approche méthodologique adaptée aux spécificités et aux nouveaux besoins du secteur agricole

François Pinet ^a, Ahmed Lbath ^b, Frédéric Vigier ^a et Michel Schneider ^c

Les attentes de la société en matière de « développement durable » touchent plus particulièrement l'activité agricole. La réponse à ces attentes passe par le pilotage de l'activité agricole à tout niveau, du niveau de l'exploitation au niveau national voire européen (Martin et Pagès, 2002). Les décisions doivent se prendre à partir d'indicateurs fiables ; cela nécessite le suivi des pratiques agricoles et par conséquent, l'enregistrement et l'échange des informations liées à ces pratiques. Pour atteindre ce but, les applications informatiques en agriculture vont donc de plus en plus faire appel aux nouvelles technologies notamment en matière d'informatique et de télécommunication. Il devient en effet indispensable de fiabiliser la collecte des données relatives aux pratiques et aux produits ; cette nécessité passe par une automatisation notamment à travers des systèmes informatiques embarqués ou portables, issus entre autres du développement de l'agriculture de précision et couplés à divers capteurs. De plus, les particularités de l'activité agricole impliquent que les informations manipulées soient souvent géoréférencées, c'est-à-dire localisées géographiquement. Composés de programmes informatiques installés sur différents sites et communicant entre eux, les systèmes distribués d'information à référence spatiale (SIRS distribués) sont utilisés afin d'assurer le traitement de grandes masses de données géographiques, ainsi que les échanges rapides entre les différents acteurs du domaine agricole.

La nécessité d'une méthode de conception pour les SIRS agricoles

Cette évolution vers une utilisation en agriculture, des SIRS distribués et du matériel mobile embarqué, soulève de nombreux problèmes techniques et organisationnels qui font l'objet des travaux engagés au Cemagref dans le cadre d'un thème de recherche intitulé COPAIN (communication pour l'agriculture informatisée), dont les différents projets sont indiqués dans l'encadré 1 (page suivante).

La conception des applications informatiques ciblées par ces travaux demande un cadre méthodologique formel et structuré. En effet, une méthode de conception est indispensable à la réalisation d'applications agricoles communicantes. Il est communément admis que les méthodes informelles simplement basées sur l'intuition ont « conduit à l'informatisation des problèmes plutôt qu'à l'informatisation des solutions ! » (Bédard et Larrivée, 1992 ; Lbath, 1997). Une méthode de conception structurée doit être capable d'intégrer :

- un formalisme, c'est-à-dire un langage permettant de décrire les différentes parties des applications, ainsi que les différents points de vue sur les modèles. Cette description est généralement faite sous forme de diagrammes ayant une structure normalisée ;
- une démarche méthodologique indiquant comment aborder les différentes étapes des projets et dans quels contextes utiliser les diagrammes du formalisme.

Les contacts

a. Cemagref, UR Technologies, systèmes d'information et procédés pour l'agriculture et l'agroalimentaire, BP 50085, 24, avenue des Landais, 63172 Aubière Cedex
b. LIRIS Lyon, Campus de la Doua, 69627 Villeurbanne et CIRIL SA, 20, rue Guérin, 69100 Villeurbanne
c. LIMOS Clermont-Ferrand, Campus des Cèzeaux, 63177 Aubière

Encadré 1

Les différents projets développés dans le thème de recherche COPAIN :

- la standardisation et la normalisation des données agricoles pour l'échange entre systèmes informatiques,
- la collecte des données sur le terrain,
- la recherche de sources de données pertinentes dans un environnement informatique distribué et hétérogène,
- la conception d'architectures pour le réseau Internet avec comme objectif l'échange d'information et l'accès à distance à des fonctionnalités de traitement de données spatio-temporelles,
- la mise en œuvre de réseaux sans fil hétérogènes composés d'équipements mobiles (par exemple embarqués dans des véhicules) pour assurer la continuité du réseau de communication.

Ces différents projets proposent des contributions à la fois spécifiques et complémentaires développées dans différentes publications (Thirion et Soullignac, 2000 ; Martin, 2001 ; Chanet *et al.*, 2003 ; Jaudoin *et al.*, 2003 ; Soullignac et Gibold, 2003 ; Jaudoin *et al.*, 2004) et dans *Ingénieries-EAT*, n° spécial Technologies pour les agrosystèmes durables, 2003.

Notre objectif était donc de faire le point sur les méthodes de conception existantes pour envisager leur application au développement d'applications agricoles communicantes. La contribution du présent article est d'établir un état de l'art des méthodes de conception pouvant être appliquées dans les projets informatiques du domaine agricole et de montrer en quoi ces méthodes peuvent être utiles pour développer la modélisation. Nous avons aussi identifié les limites des méthodes existantes et les besoins de formalisation complémentaires. Les applications ciblées sont en fait très proches des applications qualifiées de « télégéomatique » (Laurini, 2000), car elles comprennent des programmes informatiques communicants s'adressant à une classe de problèmes de gestion de données spatio-temporelles. À notre connaissance, il n'existe à l'heure actuelle qu'une seule proposition de méthodes de conception pour la télégéomatique (Pinet, 2002).

Nous présenterons tout d'abord des rappels, sur la modélisation orientée-objets sur laquelle se base les méthodes présentées. Ensuite, nous décrirons les méthodes de conception existantes pour la géomatique. Ces méthodes se donnent pour objectif principal de guider la mise en place d'applications manipulant des données spatio-temporelles. Puis, nous donnerons quelques exemples de méthodes pour le développement de systèmes communicants. Nous évoquerons aussi l'existence d'une démarche méthodologique pour

la télégéomatique et nous conclurons en mettant en évidence les limites d'adaptation des méthodes existantes au domaine agricole.

Modélisation orientée-objet

Il y a quelques années, la méthode MERISE était très utilisée pour concevoir des systèmes d'information ; cette méthode est basée sur l'analyse et la modélisation séparées des données et des traitements. MERISE a déjà largement prouvé son efficacité dans les projets liés aux agrosystèmes (Martin, 2001) notamment avec l'utilisation d'ateliers de génie logiciel (AGL) comme PowerAMC.

Les méthodes orientées-objet (méthodes OO) sont plus récentes que MERISE et se basent sur une manière plus naturelle d'aborder les représentations. En effet, contrairement à MERISE, les méthodes OO reposent sur la traduction du monde réel en **objets** encapsulant les données et les traitements. Le concepteur modélise ainsi les entités du monde réel en considérant les données à stocker (appelées **attributs**) et les traitements à effectuer (appelées **opérations**) par chaque objet. Dans la terminologie OO, les objets ayant les mêmes attributs et les mêmes opérations sont issus de la même **classe d'objets**. Par exemple, chaque parcelle est appelée « objet de la classe Parcelle ». Les applications informatiques sont donc décrites par un ensemble de classes d'objets définies par le concepteur. Des relations entre

les classes peuvent être spécifiées pour décrire des liens entre les objets. Il est ainsi possible de définir que les objets de la classe Véhicule (c'est-à-dire des véhicules) « cultivent » des objets de la classe Champ (c'est-à-dire des champs) ou bien encore que les objets de la classe Produit Phytosanitaire « sont appliqués » sur des objets de la classe Champ.

Depuis près de dix ans, une unification des formalismes de modélisation OO a débutée ; le langage graphique UML (*Unified Modelling Language*) en est le résultat actuel (Booch *et al.*, 1998). UML propose plusieurs diagrammes, qui utilisés en conjonction, peuvent décrire une application informatique ou un système d'information :

- le diagramme de cas d'utilisation décrivant l'environnement d'un système et ses fonctionnalités (les services qu'il va rendre) ;
- le diagramme de classes décrivant les différentes classes d'objets de l'application ou de la base de données ;
- le diagramme d'objets présentant une illustration des objets des classes ;
- les diagrammes de collaborations et de séquence pour tracer les échanges de messages et d'informations entre objets ;
- le diagramme d'états-transitions permettant de définir les différents états par lesquels passe un objet (par exemple une parcelle peut passer de l'état « non épandue » à l'état « épandue ») ;
- le diagramme d'activités modélisant le déroulement des procédés et des algorithmes le plus souvent sous-jacent aux cas d'utilisation ;
- le diagramme de composants correspondant aux choix liés à l'implémentation d'une application ;
- le diagramme de déploiement indiquant l'implantation des applications dans l'environnement matériel choisi.

La modélisation OO possède de nombreux atouts :

- elle offre un cadre pour la réutilisabilité et l'extension des applications grâce à des mécanismes spécifiques (notion d'héritage, de polymorphisme) (Booch *et al.*, 98) ;
- les formalismes OO ont rencontré un franc succès aussi bien dans le milieu de la recherche que dans le milieu industriel ; un nombre grandissant d'outils informatiques les supportant sont commercialisés (par exemple, Rational Rose) ;

- de nombreuses propositions d'approches OO ont été faites dans le cadre de méthodes de conception spécifiques à des domaines tels que la géomatique ou les systèmes communications ;

- les méthodes OO proposent de modéliser toutes les facettes des applications informatiques (communication entre les composants, déploiement du système, description des algorithmes...) et offre une cohérence entre chaque partie des descriptions ;

- l'aspect « graphique » des formalismes OO facilite la compréhension de la description des systèmes notamment dans les phases de présentation des spécifications aux différents acteurs des projets.

Dans cet article, nous nous focaliserons sur les méthodes OO.

Méthodes et formalismes de conception pour la géomatique

Plusieurs travaux soulignent l'intérêt d'une méthode de conception OO spécifique à la géomatique ; c'est ainsi que des propositions de méthodes ont vu le jour. Leur objectif est de faciliter la description des données géographiques en proposant un formalisme et parfois une démarche de conception spécifiques. Le but est de faire ressortir les informations géographiques dans les diagrammes tout en diminuant la complexité et facilitant la lisibilité de ces derniers. Le domaine des méthodes OO pour la géomatique semble être arrivé à maturité avec plusieurs propositions majeures : MECOSIG (Pantazis et Donnay, 1996), GeoOOA (Kosters *et al.*, 1997), OMEGA (Lbath, 1997), T-OMEGA (Pinet, 2002), MADS (Parent *et al.*, 1999), OMT-G (Borges *et al.*, 1999), Geo-Frame (Jugurta et Cirano, 1999), Geo-UML (Brodeur *et al.*, 2000). Nous citerons aussi la proposition du Cemagref (travaux de thèse d'André Miralles en cours) qui se concrétise par le développement d'un « profil SIG » robuste, au sein de l'AGL Objecteering.

La quasi-totalité des travaux incorpore une extension des diagrammes de classes OO. Par une étude de ces propositions, nous avons pu faire émerger plus précisément les besoins principaux auxquels les formalismes répondent, à savoir :

- la modélisation des types géographiques manipulés (point, lignes, polygone...),

– la modélisation de relations topologiques entre les objets géographiques décrits,

– la modélisation de la dimension temporelle pour traduire l'évolution des entités concernées.

Nous allons exprimer chacun de ces besoins et décrire comment les formalismes y répondent. Le but de la présentation est de souligner ce que permettent de modéliser les méthodes pour la géomatique.

La modélisation des types géographiques

Dans les formalismes étudiés, un type géographique correspond à un type spatial (les types polygone, ligne ou point sont des types géographiques). Dans ces formalismes, chaque type d'entités géographiques du monde réel est modélisé par une classe d'objets. Lors de l'élaboration des spécifications, le concepteur doit alors pouvoir définir simplement au sein des diagrammes de classes, les types géographiques qu'il souhaite affecter aux classes d'objets. Par exemple, sur la figure 1 présentant un diagramme de classes du formalisme de GeoOOA, un pictogramme est associé à chaque classe afin d'indiquer le type géographique de chaque type d'entités. Les pictogrammes indiquent que les véhicules seront stockés sous forme de point, et les parcelles et les champs sous forme de polygones.

Certaines méthodes permettent d'affecter aux classes, des types que nous qualifierons de combinés (Lbath, 1997 ; Brodeur *et al.*, 2000 ; Pinet, 2002). Par exemple, il est possible de spécifier qu'une classe a pour type géographique soit le

type ligne, soit le type polygone. La géométrie affectée aux objets de cette classe sera donc soit une ligne soit un polygone.

Enfin, des propositions introduisent des notations dans les diagrammes pour distinguer d'une part, la géométrie allant servir pour le stockage de l'information dans les bases de données et d'autre part, la géométrie telle qu'elle apparaît sur un écran d'ordinateur (Pinet, 2002 ; Bédard, 2002). En effet, par exemple, une ville peut être stockée sous la forme d'un polygone mais représentée sous la forme d'un point à l'écran à un certain niveau de zoom.

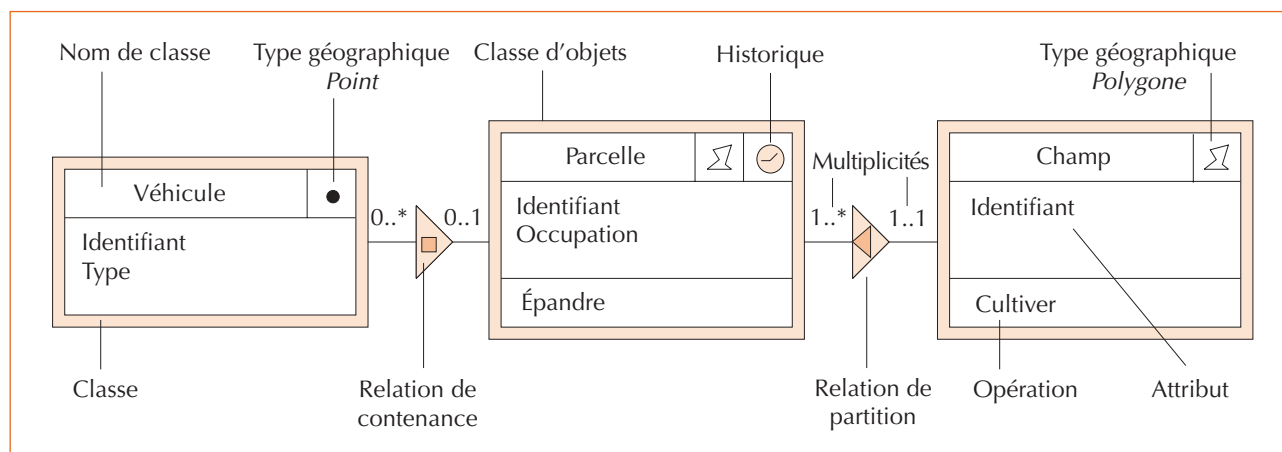
La modélisation des relations spatiales

Comme précisé dans Jugurta et Cirano (1999), les relations spatiales établissent des associations entre la géométrie des phénomènes. Dans la majorité des méthodes de conception pour la géomatique (Pantazis et Donnay, 1996 ; Kusters *et al.*, 1997 ; Lbath, 1997 ; Parent *et al.*, 1999 ; Borges *et al.*, 1999), les relations spatiales entre les classes sont généralement utilisées pour spécifier :

– des contraintes topologiques et métriques entre la géométrie associée aux objets des classes ; par exemple, les géométries de deux objets de la classe Parcelle sont toujours disjointes ou adjacentes ;

– des opérations d'agrégation dites spatiales entre des objets de classes ; par exemple, la géométrie d'un objet de la classe Pays est composée par l'agrégation des géométries des objets de la classe Région.

▼ Figure 1 – Exemple de modélisation d'un diagramme de classes avec le formalisme de GeoOOA. Quelques éléments d'une application agricole sont décrits. Trois classes sont définies : Véhicule, Parcelle et Champ. Les attributs et les opérations sont énumérés pour chacune des classes.



Dans les travaux existants, la spécification des relations spatiales se fait dans les diagrammes, par la modélisation de relations prédéfinies liant les classes, ce qui permet d'indiquer :

- des contraintes d'intégrité dans les bases de données allant contenir les informations géographiques ;
- des contraintes devant être prises en compte par les programmes informatiques développés ; par exemple, interdire le fait qu'un bâtiment puisse être dessiné sur une rivière avec un module de saisie interactive d'objets géographiques.

Par exemple, dans la figure 1, une relation relie chaque classe afin d'indiquer les liens topologiques existants entre les entités modélisées. Selon les multiplicités établies sur la figure 1, une parcelle contient de 0 à N véhicules (noté 0..*) et un véhicule peut être contenu par 0 ou 1 parcelles (noté 0..1). Il s'agit de l'expression d'une contrainte topologique. Aussi, un champ est composé de 0 à N parcelles (écrit 0..*) et une parcelle ne peut appartenir qu'à 1 champ. Ce dernier exemple exprime une opération d'agrégation spatiale ; dans l'exemple, l'emprise spatiale du champ est en effet l'agrégation des emprises spatiales des parcelles.

La contribution de Pinet (2002), Pinet et Lbath (2003) et Kang *et al.* (2004) introduit l'utilisation d'un langage textuel formel pour décrire les contraintes topologiques. Ce langage s'ajoute à la représentation graphique du diagramme de classes et a pour avantage de permettre d'exprimer de manière encore plus précise les contraintes topologiques. Néanmoins, en comparaison avec la modélisation par relations entre classe, le principal inconvénient des langages formels est que l'expression des contraintes est réalisée de manière moins intuitive.

L'expression de la temporalité

Les informations des nouvelles applications agricoles peuvent être de nature spatio-temporelle. Un besoin de ce type d'applications est l'expression du temps pour modéliser les évolutions des informations spatiales ; par exemple, pour pouvoir spécifier que l'historique des évolutions d'une donnée particulière est stocké afin de fournir une traçabilité. Selon Kosters *et al.* (1997), une application répondant à un tel besoin est qualifiée d'application spatio-temporelle. Les formalismes OO modélisant les informations spatio-temporelles proposent le plus souvent

la possibilité de définir un historique rattaché à un objet. Cet historique permettra de conserver l'évolution de l'objet c'est-à-dire toutes les modifications appliquées sur l'objet. La figure 1 illustre le rattachement d'une temporalité à une classe d'objets. Dans cet exemple, un pictogramme en forme d'horloge indique, dans les spécifications, que les différentes évolutions de la géométrie d'une parcelle seront conservées dans la base de données de l'application décrite.

Le bilan

Nous pensons que les méthodes de conception pour la géomatique fournissent aux concepteurs des moyens réellement efficaces pour décrire les applications agricoles. À l'heure actuelle, de nombreuses méthodes se basent directement sur le standard UML. Grâce à des outils appelés atelier de génie logiciel (AGL), les diagrammes peuvent être saisis dans un format électronique et une partie des programmes peut être générée automatiquement à partir des spécifications. Le but de l'outil AGL est de diminuer les coûts de développement et d'assurer une qualité de modélisation et de documentation. La majorité des méthodes de conception pour la géomatique sont supportées par un outil de type AGL. Le tableau 1 (page 148) compare les différentes méthodes en soulignant leurs spécialités.

En ce qui concerne la géomatique, nous pensons que l'extension de l'AGL d'Objecteering réalisée par le Cemagref est tout à fait adaptée aux besoins internes. Cette extension permet de générer des programmes informatiques ainsi que des modèles d'implémentation, tout en donnant la possibilité de décrire le type de géométries associé aux classes. L'AGL Objecteering supporte la génération de codes informatiques dans différents langages de programmation (C++, Java, Visual Basic, SQL) ; ce qui couvre un grand nombre d'applications développées au sein du Cemagref.

Méthodes et formalismes de conception pour les systèmes communicants

Au-delà de la description de l'information géographique, il faut pouvoir spécifier la distribution et les communications entre les applications télégéomatiques agricoles ; par exemple le partage de l'information géographique par les différentes applications informatiques et les messages envoyés pour communiquer. Plusieurs

	Année approximative de création	Version précédente	Démarche méthodologique	Spécialités	AGL
MECOSIG	1994	-	Oui	Regroupement des classes par thèmes	Non
OMEGA	1995	-	Oui	Adapté pour les contacts avec les utilisateurs finaux	AIGLE
GeoOOA	1995	-	Partielle	Description des réseaux (canalisation, réseau hydraulique...)	REGIS
MADS	1997	-	Partielle	Représentation multi-résolution	COBALT
OMT-G	1999	Geo-OMT	Oui	Description des réseaux, données spatiales continues	Non
GeoFrame	1999	GMOD	Partielle	Description des données spatiales continues	Non
Geo-UML	1999	MODUL-R	Adaptation possible de la démarche de MODUL-R	Modélisation précise des historiques de données	Perceptory
T-OMEGA	2002	OMEGA	Oui	Méthode intégrée pour la spécification des applications géomatiques mais aussi télé-géomatiques	T-AIGLE
Profil SIG	En cours	-	Partielle	Extension d'UML pour les SIRS – basée sur des patrons de conception et développée par le Cemagref	Extension de Objecteering

▲ Tableau 1 – Méthodes de conception et formalismes pour la géomatique.

formalismes ou méthodes OO ont été spécialement conçus pour la modélisation de systèmes communicants. Ces propositions sont faites pour répondre aux besoins de spécification des composants de communication. En effet, comme l'indique Liu (1998), ces méthodes cherchent à répondre aux besoins du concepteur de systèmes communicants :

- spécification de composants de communications réutilisables sous forme de « boîtes noires »,
- définition de protocoles des communications utilisés,
- définition de moyens de synchronisation,

- modélisation de la création dynamique de processus,
- modélisation de la reconfiguration des structures de communication,
- spécification de contraintes pour exprimer le temps réel,
- modélisation de la concurrence inter et intra objets,
- définition de l'affectation des tâches aux ressources systèmes ou processeurs,
- distribution et répartition des composants,
- ...

Un avantage majeur des méthodes est d'offrir un cadre formel pour valider de manière automatique les protocoles de communication c'est-à-dire vérifier s'ils n'entraînent pas un comportement non souhaité par le concepteur (boucles infinies, inter-blocage...).

Nous présentons ici quelques exemples de méthodes (ou formalismes) OO pour les systèmes communicants. Ces méthodes ont été choisies parmi les plus récentes et sont relativement représentatives des méthodes OO proposées pour la conception des systèmes communicants.

Le standard UML

Quelques notations et concepts utilisables pour la modélisation des systèmes communicants ont été identifiés dans le standard UML (Booch *et al.*, 1998). Nous citerons par exemple :

- les classes actives pour définir des objets supportant la concurrence (l'exécution simultanée de certaines opérations informatiques). Techniquement parlant, un objet actif est rattaché à un *thread* de contrôle spécifique, ce qui lui permet d'exécuter des traitements concurrents des autres objets actifs ;
- les signaux (c'est-à-dire les messages) que des objets actifs peuvent s'échanger ;
- les diagramme d'états-transitions décrivant le comportement de l'objet lorsqu'il reçoit un signal ;
- le type de communication des messages entre les objets (synchrone/asynchrone) ;
- la distribution des applications par un concept d'UML appelé valeur taguée qui se présente sous la forme d'une annotation {localisation =...} associée à chaque classe. Par exemple, une classe pourrait avoir comme valeur taguée {localisation = Serveur Aristote Clermont Fd} pour indiquer que les objets de la classe sont physiquement stockés sur le machine serveur Aristote de Clermont-Ferrand.

Les possibilités du standard UML pour réaliser des spécifications de système d'information communicants restent somme toute limitées. D'autres propositions existent afin de définir plus précisément les modules de communications des applications.

L'extension UML-RT

Une extension très aboutie d'UML pour les systèmes communicants est sans nul doute UML-RT

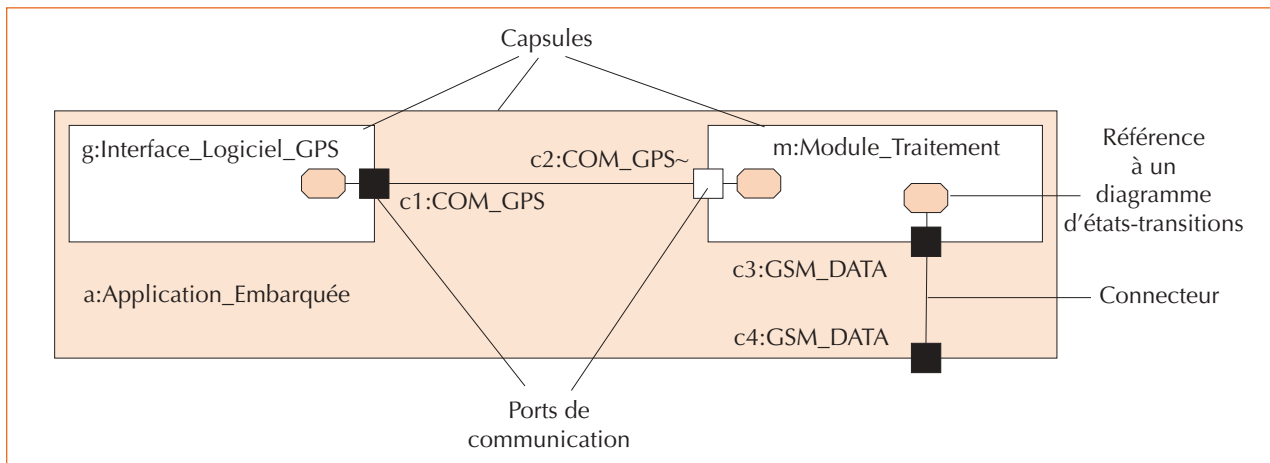
(*UML for Real Time*) (ObjectTime, 1998). Cette méthode est supportée par l'AGL nommé *Rational Rose Real Time* et se base sur l'intégration dans UML du célèbre formalisme ROOM (*Real Time Object Oriented Modeling*) (Selic *et al.*, 1994). Ainsi, l'extension d'UML développée consiste en l'intégration d'un diagramme supplémentaire à UML : le diagramme de collaboration de capsules. Ce diagramme pourra être renseigné durant la phase de conception du système dès que les modules de communication des applications seront identifiés. Ce nouveau diagramme permet de concevoir des modules de communication sous forme de composants imbriqués et/ou interconnectés.

La conception d'un diagramme de collaboration de capsules se fait par la définition :

- de capsules, c'est-à-dire des composants potentiellement concurrents. Chaque composant peut contenir d'autres composants (des sous-capsules) ;
- de ports ; les composants interagissent entre eux grâce à des ports. Les ports appartiennent à une capsule. Chaque port peut être associé à un protocole de communication au sens défini par UML-RT (c'est-à-dire l'ensemble des signaux qu'un port peut recevoir ou envoyer) ;
- de connecteurs ; les ports sont reliés entre eux par des connecteurs véhiculant les signaux.

La figure 2 (page 150) décrit un exemple de diagramme de collaboration de capsules. La capsule (c'est-à-dire le composant) notée *a* de type *Application_Embarquée* est composée de deux sous-capsules (*g* : de type *Interface_Logiciel_GPS* et *m* : de type *Module_Traitement*). Ces deux sous-capsules communiquent grâce aux port de communication *c1* et *c2* utilisant un protocole appelé *COM_GPS*. La capsule *m* communique avec l'extérieur de l'application embarquée grâce aux ports *c3* et *c4*.

Des diagrammes d'états-transitions peuvent être ajoutés afin de décrire le comportement des capsules lors de la réception de messages (par exemple, si un port reçoit le signal A alors envoyer le signal B sinon attendre). La figure 3 (page 150) présente la description, par un diagramme d'états-transitions, du comportement du port de communication *c1* (interface logiciel GPS). Ce port reçoit des demandes d'activation/désactivation de la fonctionnalité GPS, ainsi que des requêtes de localisation.



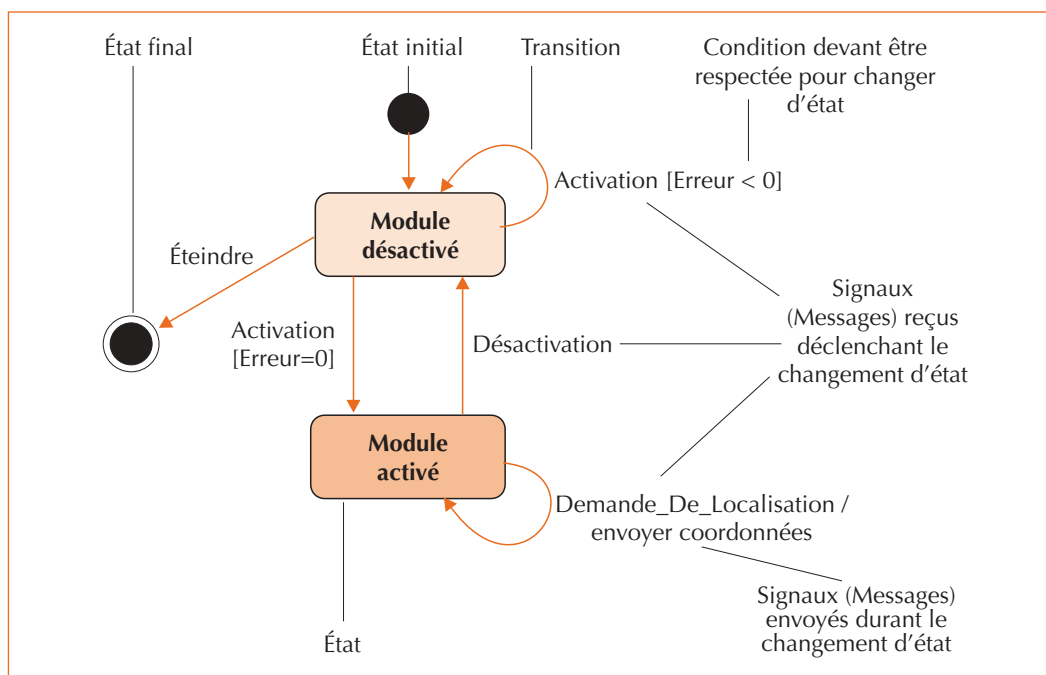
▲ Figure 2 – Exemple de modélisation d'un diagramme de collaboration de capsules avec le formalisme de UML-RT – Description des composants d'une application embarquée intégrant une interface GPS (*Global Positioning System*). Le port *c1* peut recevoir des messages inclus dans la liste de messages entrants définie par le protocole COM_GPS. De même, ce port peut envoyer des messages inclus dans la liste de messages sortants définie par le protocole COM_GPS. Le port *c2* apparaît en blanc, ce qui indique que ce port utilise l'inverse du protocole COM_GPS (tous les messages entrants du protocole deviennent des messages sortants et tous les messages sortants deviennent entrants). Le port *c4* est un relais du port *c3*.

Ainsi, le nouveau diagramme proposé par UML-RT rend possible la modélisation de composants (capsules) imbriqués. La cohérence des diagrammes d'états-transitions associés aux diagrammes de collaboration de capsules peut être vérifiée grâce à des outils formels.

Le formalisme SDL 2000

SDL (*Specification and Description Language*) est un formalisme développé pour la modélisation des systèmes communicants (Ellsberger et al., 1997). Le standard SDL est maintenu par l'ITU (*International Telecommunications Union*). De

► Figure 3 – Exemple de modélisation d'un diagramme d'états-transitions – Description du port de comportement du port de communication *c1* (figure 2).



grands groupes tels que Alcatel, Ericsson, Fujitsu, Hewlett-Packard, Motorola, Nokia ou Siemens ont adopté SDL pour leurs développements. Fin 1999, des modifications majeures et des concepts proches d'UML ont été intégrés au standard afin d'obtenir le formalisme SDL 2000 (ITU, 1999).

SDL 2000 permet de modéliser la structure et le comportement des systèmes communicants. On retrouve la même notion de composant que dans UML-RT. SDL 2000 permet de définir des agents (des composants) en interactions. Un agent est :

- soit un processus. En SDL, le terme processus ne revêt pas forcément le même sens que dans les systèmes d'exploitation informatiques. En SDL, un processus est un agent pouvant être composé d'autres processus. L'exécution des éléments inclus dans un processus n'est pas concurrente ;
- soit un bloc. Un bloc est un agent pouvant contenir des processus et/ou des blocs. L'exécution des éléments inclus dans un bloc peut être concurrente.

La description des agents avec le formalisme de SDL revient à la modélisation en capsules de UML-RT. Les agents communiquent entre eux par des signaux, à travers des canaux (c'est-à-dire des ports). Un ou plusieurs diagrammes d'états-transitions peuvent composer un bloc ou un processus afin de décrire les comportements des agents lorsqu'ils reçoivent des signaux. Les diagrammes d'états-transitions de SDL permettent d'exprimer des concepts poussés et sont plus riches que ceux de UML-RT. En effet, des notations particulières ont été intégrées aux diagrammes d'états-transitions de SDL pour permettre de définir des notions évoluées comme l'utilisation de variables, la gestion de *timers*, la sauvegarde de signaux reçus pour un traitement ultérieur, la gestion des exceptions (c'est-à-dire des erreurs)...

En résumé, SDL 2000 offre des fonctionnalités très complètes, notamment en ce qui concerne les diagrammes d'états-transitions. De plus, de nombreux outils de type AGL supportent SDL : TAU SDL Suite (Telelogic), Cinderella SDL (Cinderella), OSD Tool (LG Soft Lab), Melba96 (*Center for Advanced Technology in Telecommunications*, université RMIT Australie), JADE (université fédérale de Minas Gerais, Brésil), QUEST (université d'Essen, Allemagne)... Certains de ces outils permettent l'utilisation conjointe de diagrammes de SDL avec des diagrammes UML.

Une démarche de conception pour la réalisation des applications télégéomatiques

En complément des diagrammes de modélisation, une méthode de conception complète pour la télégéomatique doit intégrer une démarche méthodologique adaptée aux projets ciblés. Actuellement, il n'existe qu'une seule proposition de démarche pour la gestion des projets télégéomatiques (Pinet, 2002). En effet, seule la méthode T-OMEGA offre une méthodologie couplant la mise en place des données spatio-temporelles avec la définition de la distribution et de la communication entre applications. T-OMEGA préconise donc d'utiliser des diagrammes pour la représentation de l'information géographique (tels que ceux présentés dans la partie sur les méthodes de conception pour la géomatique) et des diagrammes pour la définition des communications et de la distribution (comme ceux présentés dans la partie sur les méthodes de conception pour les systèmes communicants).

Les créateurs de T-OMEGA soulignent que la conception des applications télégéomatiques implique des acteurs ayant des profils différents. Selon eux, la première grande catégorie d'acteurs est composée des concepteurs spécialistes de la géomatique (les « géomaticiens ») : ils sont spécialistes de l'acquisition, de la mise en place, de la présentation et du traitement des informations géographiques. Nous prenons le terme géomaticien au sens large en y incluant les spécialistes des SIG, les géomètres, les géographes... Pour les utilisateurs finaux, les applications télégéomatiques sont souvent perçues comme des applications géomatiques, les interactions avec d'autres applications étant souvent complètement encapsulées dans le système. Par exemple, l'utilisateur final d'une application embarquée d'un véhicule peut demander un itinéraire. L'application se connecte à un site central pour demander l'état du trafic routier puis fait part des résultats à l'utilisateur. À aucun moment, l'utilisateur n'a conscience des communications avec un site central. De la même façon, un utilisateur peut se connecter à une application client et accéder, sans en avoir conscience, à des données stockées sur un serveur. Le rôle de collecter les besoins des utilisateurs finaux revient donc principalement aux géomaticiens. Ils doivent impérativement mener une collaboration étroite avec une deuxième catégorie d'acteurs que nous appellerons les concepteurs « spécialistes des systèmes communicants (ou bien des

télécommunications) ». Il s'agit d'informaticiens, de spécialistes des réseaux, d'ingénieurs en télécommunication... Les spécialistes de systèmes communicants sont capables de proposer des modules de communications adaptés aux besoins des applications télégéomatiques en choisissant les architectures logicielles et les supports matériels adéquats. En effet, les problèmes liés aux communications entre les applications télégéomatiques sont souvent très pointus (utilisation de caches, architectures embarquées, mobilité...) et nécessitent donc des descriptions détaillées réalisées par des spécialistes.

T-OMEGA étend les démarches incrémentales classiques des méthodes orientées-objets que l'on retrouve par exemple dans RUP (*Rational Unified Process*) (Booch, 1998). Les étapes dans le cycle de développement qui sont proposées par T-OMEGA sont le reflet de plusieurs rapports de consultants et de retour d'expérience en collectivités locales (Bernard et Essevaz-Roulet, 1995). Les différentes phases de la démarche de T-OMEGA pour la mise en place d'applications télégéomatiques sont :

- la première phase dite d'analyse préalable est consacrée à la mise en place du cadre du projet, c'est-à-dire à l'élaboration du programme de travail, des calendriers et des procédures de décision ;
- la seconde phase est dite de conception prenant à la fois en compte la constitution des diagrammes modélisant l'information spatio-temporelle et le développement des diagrammes décrivant la distribution et les communications. Les rôles respectifs du géomaticien et du spécialiste des systèmes communicants sont décrits ;
- la troisième phase dite de développement intègre l'implémentation et la mise en place des applications ;
- la quatrième phase appelée phase d'audit est dédiée à l'évaluation et à l'évolution des applications réalisées.

À l'issue de chacune des phases, un document technique doit être émis afin de valider le niveau atteint dans le cycle de développement. Il est néanmoins possible de revenir à une phase antérieure afin d'y introduire des modifications. Il est aussi possible de revenir à des étapes précédentes au sein des phases. Le détail de la démarche méthodologique peut être trouvé dans Pinet (2002).

Conclusion : limitations des méthodes existantes pour la modélisation des agrosystèmes communicants et pistes de recherche

Notre objectif était de faire un tour d'horizon des méthodes et formalismes existants permettant de spécifier les applications informatiques communicantes en agriculture. Ces applications sont en fait qualifiées de télégéomatique et il existe, d'une part, des moyens pour décrire sous forme de diagrammes les données géographiques manipulées et d'autre part, des langages graphiques pour spécifier les communications et la distribution. La démarche méthodologique de T-OMEGA orchestre la conception des diagrammes issues de ces deux classes de formalismes.

L'effet de ces méthodes est d'améliorer la production de systèmes informatiques et l'état de l'art que nous avons fait a permis d'évaluer les possibilités offertes par les propositions existantes.

Néanmoins, il reste de nouvelles extensions à définir afin d'adapter les méthodes plus spécifiquement au domaine des agrosystèmes communicants. Nous allons donc énumérer différentes pistes de recherche possibles.

Vers une spécialisation de la démarche méthodologique

Dans le cadre de conception d'agrosystèmes communicants, le nombre d'acteurs impliqués dans un projet (par exemple, les chambres d'agriculture, les ministères, les exploitants, les coopératives agricoles...) demande une prise en compte des besoins et des souhaits de chacun. D'une manière similaire, le caractère économique doit bien sûr aussi être considéré. Les paramètres liés à la protection de l'environnement dans le cadre d'une agriculture raisonnée ont un impact majeur sur les décisions. Enfin, des contraintes réglementaires doivent être respectées.

Ainsi, au cours des projets de conception d'agrosystèmes communicants, des paramètres sociaux, économiques, environnementaux et réglementaires entrent en ligne de compte dans les choix techniques. Il serait donc souhaitable d'inclure dans une démarche méthodologique de conception, des règles de choix techniques qui en fonction de contraintes sociales, économiques, environnementales et réglementaires fournissent un ensemble de solutions d'implantations possibles. En fait, il s'agit de procéder à une analyse

des besoins, des données et des systèmes de communication. Les solutions qui résulteraient de ce processus d'aide à la décision seraient par exemple :

- des types d'architecture du système (centralisé, distribué, hybride, ad hoc...)
- une proposition de distribution des données pour les différentes entités du système,
- la définition des outils logiciels à employer,
- le matériel et les moyens en infrastructure réseau à utiliser (réseau filaire, bluetooth, IEEE 802.11, satellite...)
- ...

Par conséquent, la démarche de gestion de projet guiderait les concepteurs dans le processus de choix en offrant un moyen d'aide à la décision.

Par ailleurs, il semble indispensable de proposer une spécialisation de la démarche en fonction de la thématique agricole. Devant l'inefficacité et la difficulté d'adaptation souvent constatées des propositions trop génériques, il est primordiale d'offrir une personnalisation en fonction du type de projet agricole. Une spécialisation de démarche pour différents projets est en cours.

Vers une extension du formalisme

Les diagrammes des spécifications (et notamment les diagrammes de classes) peuvent intégrer des concepts directement liés à agriculture. En effet, tout comme des notations ont été ajoutées aux formalismes pour pouvoir modéliser simplement les types d'objets géographiques ou bien encore les données de type canalisation, des notations peuvent être introduites pour définir la notion de terre cultivée ; une des propriétés d'une terre cultivée est de faire hériter des caractéristiques à la culture (dans le cas d'une pollution par exemple). La relation existant entre une terre cultivée

et sa culture est une relation de type « support ». La culture hérite par exemple de la pollution de son support (une parcelle).

En ce qui concerne l'expression des contraintes, les formalismes existants se limitent à la spécification d'invariants topologiques entre objets géographiques (un exemple de ce type de contraintes est « un objet géographique O1 doit toujours être inclus dans un second objet géographique O2 »).

Les systèmes communicants agricoles nécessitent la spécification de contraintes prenant à la fois en compte des critères spatiaux et des conditions de communications. Par exemple, il faudrait pouvoir modéliser formellement dans les diagrammes, la contrainte « une application serveur ne peut communiquer qu'avec les sites se trouvant dans une zone géographique donnée ».

De plus, les systèmes communicants supportant la traçabilité agricole nécessitent la spécification de contraintes considérant un critère non plus seulement géographique mais aussi spatio-temporel. Un exemple de contrainte spatio-temporelle est « si une céréale a été entreposée dans une exploitation E à un instant T alors la céréale aura une propriété P ».

Les travaux de Pinet (2002), Pinet et Lbath (2003) et Kang *et al.* (2004) proposent un début de réponse à la modélisation de telles contraintes. Mais le caractère textuel et trop formel du langage de contraintes proposé rend son utilisation difficile pour le concepteur non initié. Une piste de recherche tendant à rendre ce type de langage plus abordable peut donc être envisagée. La possibilité d'utiliser les langages visuels doit par exemple être étudiée ; ce type de langages informatiques permet en effet aux concepteurs de décrire une contrainte en « dessinant » les objets spatiaux et en indiquant visuellement leur positionnement relatif. □

Résumé

Les nouveaux besoins de suivi des pratiques agricoles induisent l'apparition d'un nouveau type d'applications informatiques. Ces applications se caractérisent par un géoréférencement des données et par de nombreuses interactions entre programmes distants. Le développement des nouveaux agrosystèmes informatisés nécessite de plus en plus une compétence mixte, dans le domaine de la géomatique et dans le domaine des systèmes communicants. La spécification de ce type de système est complexe et rend indispensable l'utilisation d'une méthode structurée de conception. Cet article présente ce qu'il est possible de réaliser avec les différentes méthodes pouvant être employées au cours de projets de mise en place d'applications informatiques communicantes pour l'agriculture.

Abstract

The new needs for agricultural practices following result in the development of a new computer-based application type. The main characteristics of these applications are georeferenced data and interactions between remote programs. The setting up of new computer-based agricultural systems needs combined skills in the geomatic field and in the communicating system field. The specification of this system type is complex and the use of a structured methodology is essential. This paper presents the capabilities offered by the different methods generally used in projects for agricultural communicating applications.

Bibliographie

- BEDARD, Y. ; BERNIER, E., 2002, Supporting Multiple Representations with Spatial View Management and the Concept of "VUEL", *Joint Workshop on Multi-Scale Representations of Spatial Data*, Ottawa, Canada.
- BEDARD, Y. ; LARRIVEE, S., 1992, Développement des systèmes d'information à référence spatiale : vers l'utilisation d'ateliers de génie logiciel, *CISM Journal ACSGC*, vol. 46, n° 4, p. 423-433.
- BERNARD, J.-L. ; ESSEVAZ-ROULET, M., 1995, *Mise en œuvre d'un système d'information géographique*, Édition de La lettre du cadre territorial, 145 p.
- BOOCH, G. ; RUMBAUGH, J. ; JACOBSON, I., 1998, *The Unified Modelling Language User Guide*, Addison-Wesley, Baltimore, USA, 482 p.
- BORGES, K. ; LAENDER, A. ; DAVIS, C., 1999, Spatial Data Integrity Constraints in Object Oriented Geographic Data Modeling, *in the 7th ACM Symposium on Advances in Geographic Information Systems*, Kansas City, USA, p. 1-6.
- BRODEUR, J. ; BEDARD, Y. ; PROULX, M.-J., 2000, Modelling geospatial application databases using UML-based repositories aligned with international standards in geomatics, *in the 8th ACM Symposium on Advances in Geographic Information Systems*, Washington D.C., USA, p. 39-46.
- CHANET, J.-P. ; BOFFETY, D. ; LI, J. ; PINET, F. ; GERBE, P. ; SCHNEIDER, M. ; VIGIER, F. ; HOU, K.-M., 2003, Ad Hoc Network for Agriculture, *in the 4th Conference of the European Federation for Information Technology in Agriculture, Food and the Environment*, Budapest, Hongrie, p. 899.
- ELLSBERGER, J. ; HOGREFE, D. ; SARMA, A., 1997, *SDL: Formal Object-Oriented Language for Communicating Systems*, Prentice Hall, New Jersey, USA, 280 p.
- ITU, 1999, *Specification and Description Language (SDL)*, *International Telecommunication Union Recommendation Z.100 (11/99)*, 246 p.

- JAUDOIN, H. ; REY, C. ; SCHNEIDER, M. ; VIGIER, F., 2003, Interoperability of Agricultural Information Systems: a Common Ontological Approach for Various Exchange Types, *in the 4th Conference of the European Federation for Information Technology in Agriculture, Food and the Environment*, Budapest, Hongrie, p. 293-299.
- JAUDOIN, H. ; SCHNEIDER, M. ; VIGIER, F., 2004, Querying Data Sources in a SuperPeer Data Management System, *in the 6th International Conference on Flexible Query Answering Systems*, Lyon, France, à paraître dans *Lecture Notes in Computer Science*, Springer Verlag.
- JUGURTA, L.-F. ; CIRANO, I., 1999, Specifying Analysis Patterns for Geographic Database on the Basis of Conceptual Framework, *in the 7th ACM Symposium on Advances in Geographic Information Systems*, Kansas City, USA, p. 181-188.
- KOSTERS, G. ; PAGEL, B.-U. ; SIX, H., 1997, GIS-Application Development with GeoOOA, *International Journal of Geographical Information Science*, vol. 11, n° 4, p. 307-335.
- KANG, M.-A. ; PINET, F. ; SCHNEIDER, M. ; CHANET, J.-P. ; VIGIER, F., 2004, How to Design Geographic Databases ? Specific UML Profile and Spatial OCL Applied to Wireless Ad Hoc Networks, *in the 7th European Conference on Geographic Information Science*, Heraklion, Grèce, à paraître.
- LAURINI, R., 1998, La télégéomatique : problématiques et perspectives, *Revue internationale de géomatique*, vol. 8, n° 1-2, p. 27-44.
- LBATH, A., 1997, *AIGLE : Un environnement visuel pour la conception et la génération automatique d'applications géomatiques*, thèse de doctorat, spécialité informatique, INSA de Lyon, 282 p.
- LIU, A., 1998, *Dynamic Distributed Systems Design: An Architectural Design and Verification Approach*, thèse de doctorat, spécialité informatique, université du New South Wales, Sydney, Australie, 112 p.
- MARTIN, C., 2001, La mise en place de systèmes d'information dédiés au management de la qualité et de l'environnement de la production agricole: une opération pilote appliquée au secteur des grandes cultures / Méthodologie d'analyse et de modélisation d'un système d'information à référence spatiale partagé : application au projet SIREME, *Ingénierie- Eau, Agriculture, Territoires*, n° 26, p. 27-48.
- MARTIN, C. ; PAGES, C., 2002, Conditions et moyens d'une gestion informatisée des données pour la mise en oeuvre de l'agriculture raisonnée : exploration stratégique par la méthode d'audit patrimonial, *Ingénieries-Eau, Agriculture, Territoires*, n° 30, p. 43-58.
- OBJECTTIME, 1998, Designing for Concurrency, *OBJECTTime Technical Report*, Lexington, USA, 26 p.
- PANTAZIS, D. ; DONNAY, J.-P., 1996, *La conception des SIG*, Hermès, Paris, 343 p.
- PARENT, C. ; SPACCAPIETRA, S. ; ZIMANYI, E., 1999, Spatio-Temporal Conceptual Models: Data Structures + Space + Time, *in the 7th ACM Symposium on Advances in Geographic Information Systems*, Kansas City, USA, p. 26-33.
- PINET, F., 2002, *Vers une méthode de conception pour la télégéomatique : intégration et validation au sein d'un atelier de développement*, thèse de doctorat, spécialité informatique, INSA de Lyon, 243 p.
- PINET, F. ; LBATH, A., 2003, Object-Oriented Modelling of Communicating GIS-Based Applications, *in the 6th European Conference on Geographic Information Science*, Lyon, France, p. 743-749.
- SELIC, B. ; GULLEKSON, G. ; WARD, P.-T., 1994, *Real-time Object Oriented Modeling*. Wiley Professional Computing, John Wiley & Sons, New York, 1994.
- SOULIGNAC, V. ; GIBOLD, F., 2003, *Projet Sigemo, système informatisé de gestion des épandages de matières organiques – Propositions pour une architecture, version 2.0.*
- THIRION, F. ; SOULIGNAC, V., 2000, An integrated Management System of Sewage Sludge Spreading in Agriculture by using GPS and GIS, *in the 1st World Congress of International Water Association*, Paris, France, p. 258-265.