Conception de Bases de données Localisées et Temporelles (COBALT)

Modélisation conceptuelle : un module de translation pour MapInfo

Stéphane Joost* - Olivier Ertz**

* MicroGIS Geomatics & Swiss Federal Institute of Technology, CH-1015 Lausanne, stephane.joost@epfl.ch
** COMEM Department – Vaud School of Engineering, CH-1000 Lausanne, olivier.ertz@eivd.ch

RESUME : Dans le domaine de la modélisation conceptuelle, les derniers développements de l'éditeur de schéma MADS effectués dans le cadre du projet de recherche COBALT, permettent maintenant à l'utilisateur de désigner le logiciel avec lequel il a l'intention d'exploiter la structure de données élaborée. Le modèle est exporté et les informations, compatibles avec le logiciel SIG ou SGBD cible, sont stockées dans un fichier XML. Parmi les logiciels SIG désignés en début d'étude figure MapInfo pour lequel il a justement été nécesssaire de développer un translateur permettant de lire le modèle élaboré dans MADS et de créer la structure de données correspondante. Ce texte décrit le fonctionnement de cette application, mais attire également l'attention du lecteur sur la pertinence d'un tel développement, avant de proposer quelques pistes concernant les compléments qu'il serait judicieux d' apporter à ce projet de recherche.

Introduction

Dans le cadre du projet de recherche Interreg II COBALT dont l'objectif était de développer et de valider des outils susceptibles de faciliter la mise en œuvre et l'exploitation d'applications à données géographiques, l'une des tâches à réaliser était d'assurer la transmission d'informations issues d'une phase initiale de modélisation des données depuis un logiciel spécialisé vers des applications de Systèmes d'Information Géographique (SIG) standards du marché. En partenariat avec le Laboratoire de Bases de Données (LBD) de l'Ecole Polytechnique Fédérale de Lausanne (EPFL), la société MicroGIS, le laboratoire de Systèmes d'Information à Référence Spatiale (SIRS, EPFL) et le Département COMEM de l'Ecole d'Ingénieur du Canton de Vaud ont réalisé les travaux permettant de faire transiter l'information de structure des données issues du logiciel MADS vers l'application MapInfo.

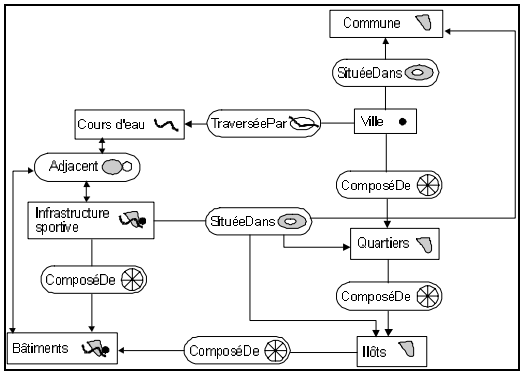

***Figure 1** : ébauche de schéma conceptuel de donnée selon les concepts MADS, de Sède 1998.*

Pourquoi modéliser ?

Comme C. Parent et al. [2] le rappellent justement, les concepteurs de bases de données passent usuellement par une étape de modélisation conceptuelle, centrée sur les caractéristiques des données considérées dans le cadre d'un projet, avant d'aborder la modélisation logique qui conduit à la mise en œuvre logicielle. Le premier avantage de ces modèles est d'être à la portée des utilisateurs. De cette manière, ces derniers ont la possibilité d'exprimer précisement leur connaissance des données applicatives et de participer activement à l'élaboration du schéma conceptuel. De plus, le résultat obtenu reste valable en cas d'évolution technologique. Au rythme actuel d'évolution, notamment pour les outils SIG, ceci est un facteur clé de flexibilité et de réduction des coûts. Enfin, la modélisation conceptuelle, par sa lisibilité, facilite les échanges d'informations entre partenaires

d'organisations différentes. A l'époque où l'accès via Internet à des sources d'informations extérieures devient courant, la capacité à comprendre la sémantique des informations obtenues est déterminante pour leur utilisation correcte. Le modèle MADS a été conçu par une équipe réunissant spécialistes de la modélisation conceptuelle et utilisateurs de bases de données spatialisées. Grâce à l'éditeur visuel de schémas MADS, un utilisateur peut très rapidement être en mesure de définir les structures de ses données. Il obtient en retour soit un fichier texte, qui permet d'exporter la définition du schéma vers un autre logiciel, ou la copie papier du diagramme qu'il a dessiné, représentatif des entités et des relations caractéristiques de son application (figure 1). Dans tous les cas, le schéma, même incomplet, est conservé par l'éditeur et réutilisable. Le concepteur pourra montrer son schéma aux utilisateurs et discuter avec eux de sa validation ou de sa reformulation. Le graphisme utilisé permet une compréhension intuitive de la signification du schéma.

Le modèle de données de MapInfo dans les grandes lignes

Une table MapInfo se compose de données attributaires et de données spatiales dans le cas où celle-ci est cartographiable ou "mapable" selon la terminologie MapInfo (présence de coordonnées géographiques). Un projet (workspace) MapInfo fait référence à une ou plusieurs table MapInfo. MapInfo est capable d'implémenter la plupart des types de données spatiales (point, ligne, polygone, etc.), mais ne permet pas d'implémenter les contraintes d'intégrité exprimées par un schéma relationnel : il n'est pas possible de définir de clé, ni de contraintes de références, ni de contraintes de cadinalité.

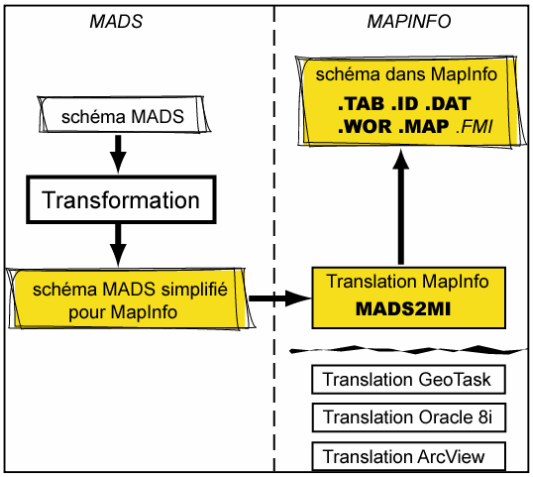

***Figure 2** : l'application MADS2MI au sein du projet Cobalt*

MADSTOMI : le module de translation pour MapInfo

Le module de translation pour MapInfo repose sur une application externe à MADS et à MapInfo. Il s'agit d'un logiciel développé en Visual Basic qui utilise le schéma XML généré par MADS pour MapInfo, et qui le traduit ensuite en une structure de fichiers au format MapInfo. Initialement, il avait même été envisagé de développer le translateur avec MapBasic directement afin de lier le plus possible les fonctionnalités désirées à MapInfo. Mais l'absence de fonctions importantes nous a finalement décidé à concevoir ce programme à l'aide de VB. Cela nous évitait notamment de réécrire un parseur XML et de le piloter à partir de MapBasic.

En déchiffrant la structure XML du fichier MADS simplifié pour MapInfo, MADS2MI exploite les caractéristiques du modèle décrit et génère les éléments suivants : les tables MapInfo (nom de table, noms des champs de la table, types des champs); un fichier MAP qui contient les coordonnées si la table comporte des données spatiales; deux tables contenant des meta-données sur le modèle : la première contient les commentaires et informations sur les différentes tables importées (METABLE_TABLE) et la seconde contient les commentaires et informations sur toutes les variables présentes dans les tables décrites décrites dans méta-table "METABLE_TABLE" (METABLE_COLUMNS); un projet MapInfo (workspace) pour stocker la structure des tables créées : "nom_du_schéma.wor"; et finalement un script MapBasic pour, si besoin est, recréer la structure sans le translateur MADS2MI ("nom_du_schéma.fmi").

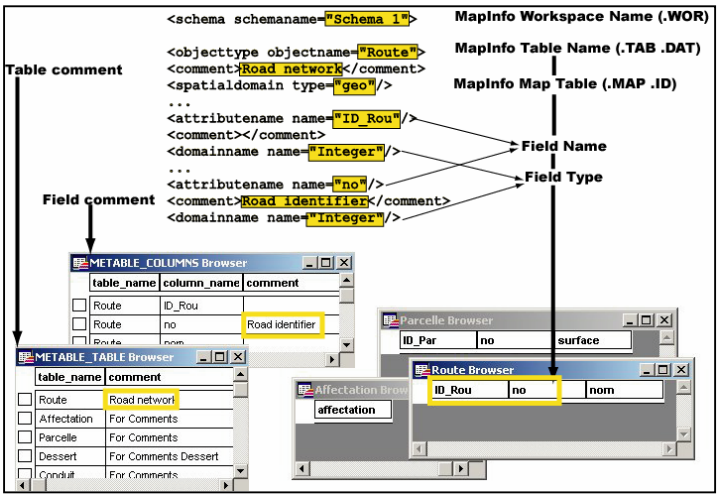

Figure 1 *: du schéma MADS à MapInfo.*

```
Create table METABLE_TABLE (table_name char(80),comment char(254)) FILE
"C:\tmp\METABLE_TABLE"
Create table METABLE_COLUMNS (table_name char(80),column_name
char(80),comment char(254)) FILE "C:\tmp\METABLE_COLUMNS"
Insert Into METABLE_TABLE values ("Route","")
Insert Into METABLE_COLUMNS values ("Route","ID_Rou","")
Insert Into METABLE_COLUMNS values ("Route","no","")
Insert Into METABLE_COLUMNS values ("Route","nom","")
Create Table Route(ID_Rou Integer , no Integer , nom Integer  ) FILE
"C:\tmp\Route"
Create Map For "Route"
```

Figure 2 *: exemple de script MapBasic créé par MADS2MI (extension . fmi).*

Utilisation de l'application MADS2MI

L'application et constituée d'une petite interface qui a l'apparence suivante :

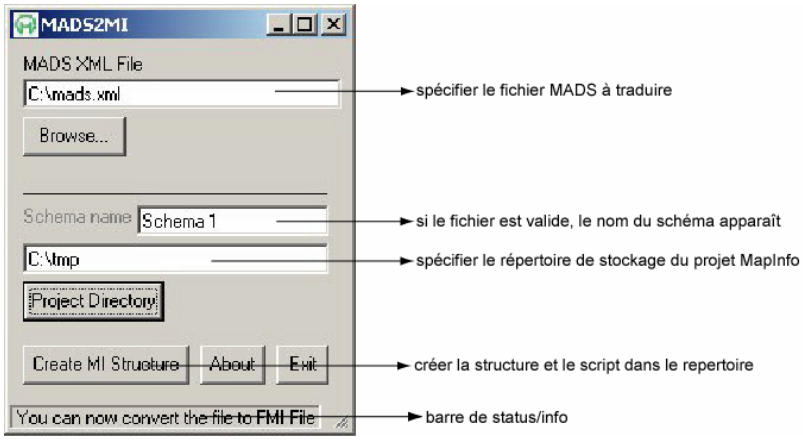

Figure 4 *: Interface de MADSTOMI et ses différentes composantes.*

Le premier bouton "Browse" permet de localiser le fichier XML issu de MADS et de le charger. Une fois le fichier localisé, si son format est correct, son nom s'affiche dans le champ "Schema name". Il faut ensuite désigner le répertoire de projet dans lequel les divers fichiers générés par MADSTOMI vont être enregistrés. Le

bouton "Project Directory" permet de sélectionner le répertoire adéquat[1]. Une fois que le répertoire est désigné, un message dans la barre de statut indique qu'il est possible de générer la structure importée pour le logiciel MapInfo. Il suffit alors de cliquer sur le bouton "Create MI Structure". Un message est affiché : l'application confirme le nom du répertoire dans lequel les fichiers ont été créés et indique qu'un script MapBasic a été généré. Ce script "schema1.fmi"[2] n'est pas compilé; c'est l'équivalent des fichiers standards ".mbx" générés par l'application MapBasic. Pour le compiler, il suffit de l'ouvrir avec MapBasic qui créera un fichier "schema1.fmi.mbx". Cette procédure est provisoire et il s'agit maintenant de déterminer si le traitement effectué par MADSTOMI devrait également intégrer la compilation de ce fichier. Actuellement ce n'est pas le cas car il est en effet possible que l'utilisateur désire faire des modifications dans le script de base pour l'adapter à ces besoins.

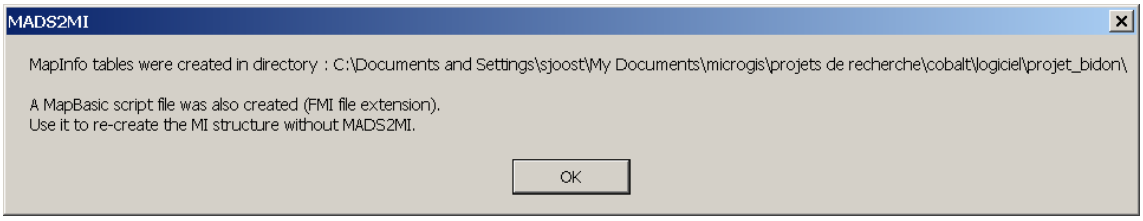

Figure 5 *: Message de confirmation du programme MADSTOMI : la structure de fichiers pour MapInfo a été correctement générée.*

Un clic sur le bouton "ok" qui suit le message de confirmation (figure 5) fait apparaître un nouveau message. Celui-ci permet à l'utilisateur de créer un fichier de projet (workspace, ".wor" pour MapInfo). Il est possible que le workspace ne soit pas désiré (accès séparé à chacune des tables) et c'est pourquoi le choix est laissé à l'utilisateur (figure 6).

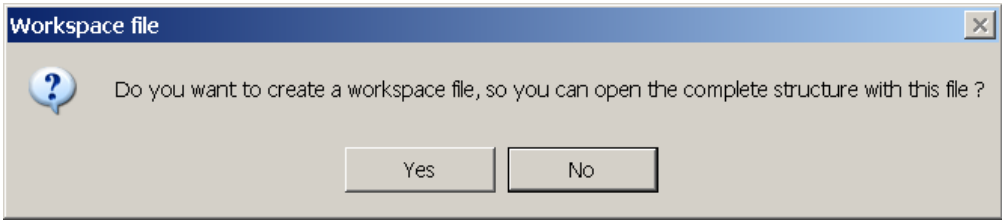

Figure 6 *: Création du workspace pour MapInfo.*

Finalement, l'application demande à l'utilisateur d'enregistrer la structure des tables créée par MADSTOMI dans MapInfo (figure 7). Il s'agit d'une boîte de dialogue standard de MapInfo.

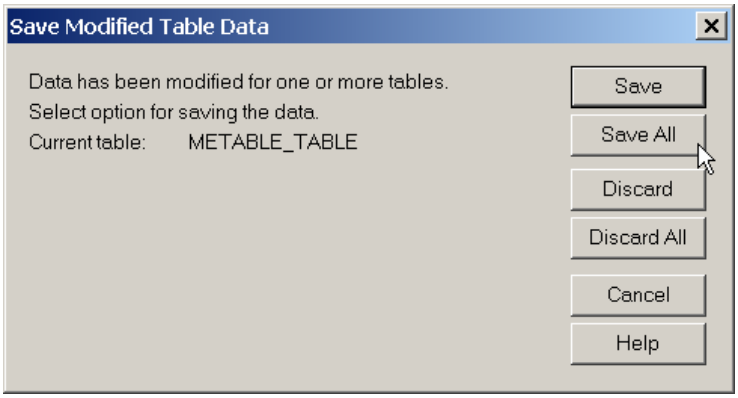

Figure 7 *: Enregistrement des différentes tables constituant le workspace dans MapInfo.*

MADSTOMI affiche encore dans la barre de status "FMI file created". Il est ensuite nécessaire de lancer l'application MapInfo et d'ouvrir le workspace "Schema1.wor". Pour visualiser les tables, aller dans le menu "Window" et choisir "New Browser Window…". Ensuite sélectionner les tables et cliquer sur "Ok" (figure 8).

[1] Dans la version actuelle (1.0 beta), le répertoire cible doit exister avant d'être désigné par l'intermédiaire des fonctions du bouton "project Directory". La possibilité de créer ce répertoire au moment de sa localisation sur le disque dur sera implémentée dans une version future.
[2] ".fmi" pour "**F**or **M**ap**I**nfo"

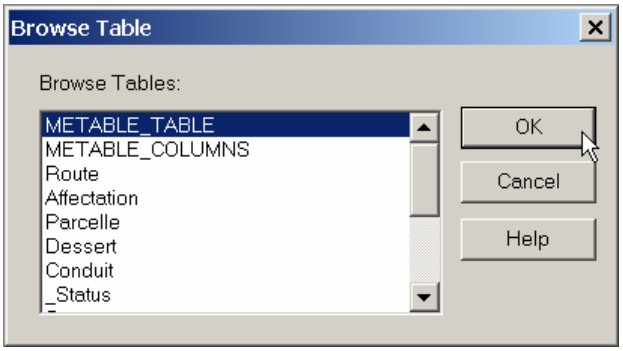

***Figure 8** : sélection des tables à visualiser dans MapInfo.*

Une fois la structure créée, il serait souhaitable que l'application MADSTOMI lance elle-même MapInfo d'une part, et qu'elle affiche immédiatement toutes les tables. Cela consituterait en effet l'aboutissement du processus depuis la création du schéma dans MADS. Cela permettrait en outre de "rassurer" définitivement l'utilisateur en lui mettant directement sous les yeux le résultat du travail du translateur. Cette adaptation devrait également être développée pour la prochaine version de MADSTOMI.

Pertinence de l'application

Le développement d'un translateur permettant d'importer une structure de données issues d'un modèle conceptuel élaboré grâce à MADS vers un logiciel SIG du type de MapInfo n'est au premier abord pas pertinent. En effet, poser le problème de la pertinence d'un tel développement, c'est se demander qui va utiliser la chaîne d'applications MADS – translateur MADSTOMI – MapInfo ? En somme, peut-on rationnellement envisager l'existence d'un utilisateur qui maîtriserait les aspects de modélisation conceptuelle (ou du moins qui aurait connaissance de cette matière, suffisamment pour utiliser l'éditeur de schémas MADS), dont le projet justement est de taille et de complexité suffisante pour justifier une phase de modélisation, et qui désirerait utiliser MapInfo (ou Arcview) mais sans recourir aux services spécialisés d'outils tels que Spatialware, SDE, Oracle 8i, ou d'un SGBD tel que Dbase, Informix, voire même Access ? Pourquoi vouloir se passer d'outils spécialisés, et d'autre part, après avoir développé un modèle conceptuel, pourquoi utiliser un logiciel qui ne pourra tenir compte ni de cardinalité, ni de contraintes d'intégrité ? Et surtout, finalement, est-il judicieux de se lancer dans la mise en place d'une solution qui ne permet pas les opérations de Reverse Engineering ? En effet, il se peut très bien qu'une première application du modèle montre qu'une adaptation est nécessaire. Il est impossible dans ce cas de remonter à la phase conceptuelle pour faire les modifications et les soumettre à un organe de décision quelconque, puis de revenir au niveau du logiciel SIG avec une structure de données modifiée en conséquence. Il faudra alors refaire fonctionner le translateur et probablement recommencer une bonne partie de l'intégration de données.

Par conséquent, la seule voie envisageable est celle du profil d'un utilisateur qui ne peut, pour des raisons économiques principalement, et de temps disponible dans une moindre mesure, se permettre l'acquisition de solutions lourdes et onéreuses, et qui juge que les problèmes de fonctionnement d'Access sont rédhibitoires. On peut en effet devoir gérer des projets relativement importants du point de vue du nombre de sources de données et d'objets, mais dont la structure n'est pas trop complexe. Mais il faut bien être conscient ici qu'en s'adressant à ce genre de profil utilisateur, un translateur du type MADSTOMI ne pourra pas prétendre à une diffusion étendue, loin de là. C'est une constatation qui vient refroidir toute perspective de commercialisation, à moins qu'une étude de marché un peu plus complète (!) ne vienne contredire cette affirmation. Par contre, toujours dans cette perspective de commercialisation, les autres fonctionnalités envisagées dans le cadre de l'application MADSTOMI (aide à l'intégration de données, voir partie 2 des conclusions et perspectives), attractives pour un plus grand nombre d'utilisateurs potentiels, pourraient venir compenser la probable faible demande pour le translateur.

Une application de ce type peut également avoir un effet éducatif. Le fait de diffuser un produit de ce type dans le domaine du logiciel libre par exemple pourrait s'avérer bénéfique dans le milieu de la conception des bases de données à composante spatiale. En effet, la pratique montre que dans la plupart des projets SIG, la phase de modélisation est trop souvent laissée de côté. De manière analogue à ce qu'il se produit avec les métadonnées que l'on néglige (perte de temps apparente), l'utilisateur ne veut pas perdre de temps et commence à exploiter les données sans prendre le temps de mettre à plat les différents objets concernés et les relations qui existent entre eux. Et cela même dans le cadre de projets relativement importants. On aboutit souvent à de grosses pertes de temps évidemment quand il s'agit de revoir de fond en comble la structure de ces données.

L'essentiel à retenir de cette partie sur la pertinence de l'application reste que la chaîne d'applications MADS – translateur MADSTOMI – MapInfo (ou Arcview) peut constituer une solution simple et économique pour des

utilisateurs au fait des méthodes de modélisation et désireux de structurer au mieux un projet. Tout ceci soulève évidemment un certain nombre de problèmes du côté de la diffusion de l'éditeur de schémas lui-même, mais ce n'est pas l'objet de ce papier.

Conclusions et perspectives

a) Dans un premier temps, il faut développer les fonctionnalités existantes du translateur. Comme cela a été évoqué dans la description du fonctionnement actuel de l'application, il existe un certains nombres de fonctions pour lesquelles des adaptations sont indispensables, et il y a encore des aspects à perfectionner (compilation du script ".fmi" ou lancement du logiciel MapInfo suite à la création de la structure par exemple).

b) Le module proposé effectue l'importation d'une structure de données MADS et permet de générer les tables nécessaires, des tables de métadonnées, ainsi que de formater les champs de données. Ce sont bien les opérations pour lesquelles son développement a été envisagé. Cependant, si l'on se place maintenant dans un contexte d'exploitation des données, ces fonctions conceptuelles ne suffisent évidemment pas et il est indispensable de prévoir les opérations nécessaires afin de concrétiser le développement d'un outil utile. Nous ne voulons naturellement pas dire par là que la chaîne élaborée dans le cadre de ce projet de recherche (de l'éditeur de schémas MADS aux applications SIG cibles) n'est pas pertinente, mais elle n'aboutira réellement que le jour où les translateurs évolués permettront d'importer des données dans la structure. Dans l'hypothèse où le groupe de recherche peut entrevoir quelques perspectives commerciales, il n'est en effet pas possible de se passer de modules permettant l'intégration de fonctions susceptibles de faciliter l'importation de données attributaires et spatiales[3]. Cela peut sembler a priori évident, mais les travaux de développement nécessaires pour y parvenir ne sont pas simples à concevoir. Conformément à ce qui est énoncé dans la partie "Pertinence de l'application", les développements envisagés devraient déboucher sur l'élaboration d'un véritable intégrateur de données, parallèlement au travail de translateur effectué sur les fichiers de structure générés par MADS. Pour alimenter la structure vide issue de la phase de modélisation, plusieurs problèmes qui dépendent pour la plupart de l'existence ou non de méta-information, doivent être résolus: la localisation des données n'est pas connue, les données peuvent exister sous forme numérique ou analogique, les formats de fichiers sont disparates, certains fichiers ne peuvent être composés que de données attributaires, d'autres exclusivement de la géométrie, etc. Nous sommes confrontés à une problématique d'importation composée qui techniquement ne devrait pas être difficile à résoudre, mais pour laquelle des soucis de composition d'interface pourraient surgir. Il serait donc effectivement utile de planifier la mise au point d'un véritable assistant à l'intégration de données accompagné d'une boîte à outils permettant de résoudre les différents problèmes.

c) En partie pour des raisons économiques que nous ne développerons pas ici [1][5], parce que la tendance générale est de fournir des services logiciels via Internet (meilleure accessibilité), et finalement pour assurer une meilleure diffusion des applications réalisées, un pan des recherches futures liées à COBALT devrait être consacré au développement de services online. Est-il envisageable de reproduire l'éditeur de schémas MADS sur interface Web ? Dans quelle mesure ? Est-ce le cas pour les translateurs vers les applications cibles, voire même pour les services complémentaires envisagés autour des translateurs ? Cela devrait constituer à notre avis un axe de recherche et de développement d'une éventuelle phase ultérieure, et mener à l'élaboration d'un ou plusieurs prototypes.

Bibliographie

[1] O. Ertz, S. Joost, D. Rappo, **Towards Geoservices Portals, WGIS trends for Business Applications**, WISE 2001, Proc. of the second Int. Conference on Web Information Systems Engineering, Kyoto, December 2001.

[2] C. Parent, S. Spaccapietra, E. Zimanyi, P. Donini, C. Plazanet, and C.Vangenot, **MADS ou l'information spatio-temporelle à portée de ses utilisateurs**, Quatrièmes rencontres de ThéoQuant, Besançon, France, Février 1999.

[3] C. Parent, S. Spaccapietra, E. Zimanyi, P. Donini, C. Plazanet, and C.Vangenot. **Modeling Spatial Data in the MADS Conceptual Model** Proc. of the 8th Int. Symp. on Spatial Data Handling, SDH'98, Vancouver, Canada, July 1998.

[4] C. Parent, S. Spaccapietra, E. Zimanyi, P. Donini, C. Plazanet, C.Vangenot., N. Rognon, J. Pouliot. and P.-A. Crausaz. **MADS: un modèle conceptuel pour des applications spatio-temporelles**, Revue Internationale de Géomatique,vol. 7, no 3-4, 1997.

[5] P. G. Wenzi, S. Winter, **Geomarketing on a Pay-per-Use Basis**, Department of Geoinformation, Technical University of Vienna, 2000.

[3] Données stockées dans des fichiers de format .shp, .tab, .xls, .dxf, etc.