

UNIVERSITE LIBRE DE BRUXELLES
Faculté des Sciences appliquées

Année académique 2001-20002

**Développement d'outils pour l'intégration
et
évolution de bases de données spatio-temporelles**

DIRECTEUR DE MEMOIRE :
M. Esteban Zimányi

VUE

TRAVAIL DE FIN D'ETUDES
PRESENTE PAR
Katumwa Ngituka, Didier
EN
DE L'OBTENTION DU GRADE
D'INGENIEUR CIVIL INFORMATICIEN

UNIVERSITE LIBRE DE BRUXELLES
Faculté des Sciences appliquées

Année académique 2001-2002

**Développement d'outils pour l'intégration
et
évolution de bases de données spatio-temporelles**

DIRECTEUR DE MEMOIRE :
M. Esteban Zimányi

VUE

TRAVAIL DE FIN D'ETUDES
PRESENTE PAR
Katumwa Ngituka, Didier
EN

DE L'OBTENTION DU GRADE
D'INGENIEUR CIVIL INFORMATICIEN

*Ce peuple, assis dans les ténèbres, a vu une grande lumière; et sur ceux qui étaient assis dans la région et l'ombre de la mort La lumière s'est levée.
A tout Seigneur, tout honneur...*

Reconnaissance et remerciements...

A Monsieur Esteban, Zimányi, mon Directeur de mémoire, pour sa disponibilité et son soutien

A Ir. Louis Jacomet et Ir. Olivier Samyn, chercheurs au service informatique de la faculté pour le temps qu'ils m'ont consacré.

A ma Mère, Lucie Kitimini : à femme d'exception, affection exceptionnelle !

A Godelieve, ma sœur, Papy et Serge mes frères, à Christian mon petit neveu, à Lwamba Masunga (Mami) ma grand-mère : l'amour n'est altéré ni par le temps ni par l'espace.

A la mémoire de Pierrot Katumwa, mon père.

A Innocent Mboma BwaKand, à Marie-Thérèse Kitimini son épouse, ainsi qu'à tous leurs enfants.

A Eric Mboma (Da) mon cousin et ami,

A Joseph Nsele, mon oncle, à son épouse Elise ainsi qu'aux « Filles ».

A toutes les familles Katumwa, Kitimini, Daba, Oliona et Dibata.

Au Rev. Laurent Nzioloko.

Au Rev. Martin Mutyebele et à toute sa famille

Au Père Karel Stuer, à la Sr. Françoise Gildemyn, à Alexandre Kawanda (VDB) et à tout le staff du Centre Amani.

A célestin Katumwa, mon cousin ainsi qu'à toute sa famille,

A Masudi Opese, Simon Beya, Gabin Kayumbi et Fernand Ngoy mes frères et amis

A Ir. Graphèle Paluku et à toute sa famille,

Aux familles Tamasala, Malungu et Mukendi.

A tous mes Amis et Frères.

Table des matières

1. Introduction	9
2. MADS et la modélisation spatio-temporelle	13
2.1 Structure d'une base de données	14
2.1.1 Le schéma Conceptuel	14
2.1.2 Le schéma Logique	14
2.1.3 Le modèle physique	14
2.2 Modélisation de données spatio-temporelles	15
2.3 Le modèle MADS	16
2.3.1 MADS et la modélisation de données classiques	17
2.3.2 MADS et la modélisation de données spatiales	17
2.3.3 MADS et la modélisation de données	21
3. Transformation de schémas classiques	24
3.1 Définition d'une transformation	25
3.2 Réversibilité d'une transformation	25
3.3 Types de transformation	25
3.4 Etude des transformations – Transformation d'attribut d'un type d'entité en type d'entité.	26
3.4.1 Représentation par instance	26
3.4.2 Représentation par valeur	28
3.4.3 Exemple de transformation de schéma classique	30
4. Transformation de schémas spatio-temporels	33
4.1 Introduction	34
4.2 Transformation d'attribut temporel	35
4.2.1 Attribut simple monovalué	35
4.2.2 Attribut simple multivalué	39
4.2.3 Attribut complexe monovalué – la temporalité sur l'attribut parent	43
4.2.4 Attribut complexe monovalué – la temporalité sur un composant	47
4.2.5 Attribut complexe multivalué	50
4.3 Transformation d'attribut spatial	53
4.3.1 Attribut simple monovalué	53
4.3.2 Attribut simple multivalué	56
4.3.3 Attribut complexe	58
4.3.3.1 Attribut complexe monovalué	58
4.3.3.2 Attribut complexe multivalué	59
4.3.4 Attribut variable	60

5. Implémentation	62
5.1 Présentation de MADS Editor	63
5.2 Implémentation des Transformations	64
5.3 Réalisation d'une transformation dans MADS Editor	64
6. Conclusion	73
Bibliographie	71

Avertissements et Conventions.

Ce travail traite un aspect des bases de données. Il est dès lors admis que le lecteur possède les notions du modèle « Entité-Association », même si certains rappels théoriques sont faits tout au long de l'ouvrage.

Toute coïncidence de noms ou valeurs contenus dans le présent document avec des éléments réels est tout à fait fortuite.

Le terme « temporalité » désigne le caractère temporel.

Le terme « spatialité » désigne le caractère spatial.

Le terme « spatio-temporalité » désigne le caractère spatio-temporel.

Le mot « classique » désignera souvent tout ce qui n'est ni temporel, ni spatial.

Tous les logiciels cités dans le présent ouvrage sont la propriété de leurs éditeurs respectifs.

Chapitre 1 : Introduction

Chapitre 1 : Introduction

De nos jours, il y a de grands volumes de données géographiques disponibles dans des bases de données gérées de manière indépendante. Le besoin d'intégrer ces bases de données fédérées est apparu, notamment à cause du coût de la récolte des informations (qui exige souvent l'envoi des hommes sur le terrain). Cependant, l'opération d'intégration de schémas nécessite généralement une transformation préalable des schémas. Notre travail va se focaliser sur l'aspect « Transformation » de schémas. Du point de vue pratique, l'outil développé dans le présent travail vient étendre les fonctionnalités de l'Editeur de schémas MADS (MADS Editor). MADS Editor a été développé dans le cadre du projet MurMur (MULTi-Representations et Muli-Resolutions).

Après avoir décrit brièvement le contexte, à savoir le projet MurMur, ainsi que le modèle MADS. Nous procéderons à une étude approfondie des opérations de transformation de schéma conceptuel. Cette étude débutera par l'analyse d'une transformation de schéma classique, pour ensuite déboucher sur l'étude des transformations de schémas spatio-temporels. Enfin, nous nous attèlerons à l'aspect pratique du travail, qui consiste à implémenter ces transformations au sein de l'outil MADS Editor.¹

¹ MADS Editor est un éditeur de schémas suivant le modèle MADS. (cfr chapitre 5 et bibliographie)

Le Projet MurMur

MurMur est un projet de recherche financé par le programme IST² de l'Union Européenne. Plusieurs partenaires dont l'Université Libre de Bruxelles (ULB), sont associés au projet. La proposition Murmur a trouvé sa motivation dans le fait que les systèmes actuels ne fournissaient pas des schémas de représentation qui prennent en compte la diversité des besoins des utilisateurs. Il est en effet courant, via le web, de fournir des informations en fonction du profil de l'utilisateur. Malheureusement cet aspect n'est pas parfaitement intégré dans le stockage des données.

MurMur a pour objectif d'étendre les fonctionnalités actuellement fournies par les logiciels commerciaux de gestion de données (SGBD³ et SIG⁴), pour supporter des schémas de représentation plus flexibles, comme la coexistence de représentations multiples du même phénomène du monde réel. Par exemple, dans une base de données relationnelle ou objet-relationnelle, la même voiture peut être représentée, d'une part comme instance d'une table Véhicule et d'autre part, comme instance d'une table Pièce de Collection, où les deux tables peuvent ne pas avoir d'attribut commun. Dans une base de données géographique multi-échelle, le même terrain peut avoir une représentation géométrique précise de sa surface à une échelle 1/10000 et une représentation moins précise à une échelle 1/50 000. Par rapport à la dimension temporelle, une rivière peut être associée à plusieurs géométries pour représenter l'évolution de son cours au travers des années.

Le projet MurMur s'est spécifiquement intéressé à *fournir des outils pour la gestion de représentations multiples qui sont adéquats pour des données géographiques*. La motivation du projet se justifie également par l'importance sociale des données géographiques et par la diversité des profils utilisateurs. Ces profils étant beaucoup plus larges dans les applications géographiques (allant de la gestion de l'environnement à la planification des villes, les administrations cadastrales, les sociologues, etc.) que dans les applications de bases de données conventionnelles, où une base de données sert généralement une communauté d'utilisateurs appartenant à la même organisation. Beaucoup d'applications sont concernées par le problème des représentations multiples,

² Information Society Technologies

³ Systèmes de Gestion de Bases de Données

⁴ Systèmes d'Information Géographique

comme la gestion de l'environnement, la prévention des risques naturels, les systèmes multi-échelles de production de cartes, la gestion cadastrale, et les applications de navigation embarquées.

Du point de vue scientifique, le principal objectif du projet est d'adresser les problèmes dus à la coexistence de plusieurs représentations. MurMur s'envisage sous deux perspectives:

- les bases de données multi-échelles, où diverses représentations à des niveaux de résolution différents sont stockées dans une base de données unique, typiquement en corrélation avec des fonctions de généralisation cartographique ;
- les bases de données temporelles qui visent la gestion de représentations multiples prises à des différents points du temps.

Chapitre 2 : MADS et la modélisation Spatio-temporelle

Chapitre 2 : MADS et la modélisation Spatio-temporelle

2.1 Structure d'une base de données :

Une base de base de données est conventionnellement (ANSI⁵) décrite sur trois niveaux. Il s'agit des niveaux conceptuel, logique et physique. Chaque niveau possède un ou plusieurs modèles de représentation.

2.1.1 Le schéma conceptuel

Le schéma conceptuel est le résultat de la phase d'analyse pour la réalisation d'une base de données. Il permet grâce à son formalisme de lever toute ambiguïté sur la définition de la base de données. Etant indépendant de la technologie à utiliser (ou utilisée) pour l'implémentation, il est un puissant outil pour l'échange d'information sur les bases de données. Le modèle conceptuel le plus utilisé est le modèle Entité-Association. Ce dernier ainsi que l'une de ses variantes (Objet – Relation) seront utilisés dans ce document.

2.1.2 Le schéma logique

Le schéma logique est la traduction du schéma conceptuel selon un modèle existant (objet ou relationnel). Il servira de référence aux différentes vues particulières. Ces schémas sont totalement indépendants de la technologie utilisée (matérielle et logicielle). Il est néanmoins moins passif que le modèle conceptuel en ce qu'il permet de définir certaines contraintes que les données doivent satisfaire.

2.1.3 Le modèle physique

Quant au modèle physique, il est un ensemble de caractéristiques techniques d'une structure de données implémentée, à savoir : les adresses de stockage des données, la taille du tampon, la taille d'une page, les pointeurs,...le modèle physique est donc fortement lié au SGBD. Il est conçu à partir du modèle logique et des caractéristiques et paramètres techniques, grâce au LDD (Langage de Définition de Données) du SGBD

⁵ American National Standards Institute

(Oracle, SQL-Server, Informix, DB2,...). Le modèle physique est fortement dépendant du SGBD.

2.2 Modélisation de données spatio-temporelles

Un modèle conceptuel spatio-temporel se doit d'offrir à ses utilisateurs une excellente richesse et une grande liberté d'expression tout, en proposant une description des schémas de données lisible intuitive. Pour répondre à ces exigences, le modèle doit :

- Se baser sur l'orthogonalité des dimensions structurelles, temporelles et spatiales. Le principe d'orthogonalité vise à assurer l'indépendance de données par rapport aux dimensions structurelles, spatiale et temporelles. Par exemple, On doit pouvoir définir d'un côté des données de manière libre sans contrainte spatio-temporelle, et de l'autre côté une donnée temporelle indépendamment de sa spatialité éventuelle.
- Etre compatible avec la modélisation traditionnelle (non spatio-temporelle) un modèle spatio-temporel doit aussi permettre la modélisation des données non spatio-temporelles. En effet, toutes les données ne sont pas forcément temporelles et/ou spatiales.
- Pour les SIG en particulier, le modèle doit offrir la connaissance des liens topologiques. Cette caractéristique est une grande importance pour les SIG. A ce jour, plusieurs SIG génèrent automatiquement les liens topologiques en les calculant à partir des coordonnées physiques des objets.

Dans les modèles existants, des concepts tels que l'agrégation, la généralisation (ou spécialisation) ainsi que la modélisation des champs continus ont été tant bien que mal proposés. Néanmoins, trois besoins majeurs restent à satisfaire :

- a. Offrir la possibilité de contraindre les objets spatiaux par rapport à un lien topologique.
- b. Pour ces liens topologiques, chaque application doit avoir la possibilité de définir une sémantique propre.
- c. Permettre la description explicite dans un schéma d'un lien topologique important pour les applications, afin de pouvoir y effectuer certaines manipulations telles que le nomage, l'affectation des attributs,..

Sur le plan structurel les modèles proposés sont soit d'une approche entité-association tels que MODUL-R⁶ et GéO2⁷ soit de l'approche orientée objet tels que GeoOM⁸, POLLEN⁹ et CONGOO. Tous ces modèles ont leur façon d'intégrer la spatialité et la temporalité.

Nous allons à présent approfondir l'étude de la modélisation spatio-temporelle par l'analyse du modèle MADS.

2.3 Le Modèle MADS.

Le Modèle MADS (Modélisation d'Applications à Données Spatio-temporelles) a été initialement conçu dans un contexte applicatif de gestion et d'aménagement du territoire, et certaines de ses orientations. De nombreuses applications telle que la gestion pétrolière rentrent dans ce cadre particulier. MADS a puisé sa motivation dans toutes les lacunes et imperfections des modèles existant. Ses principales caractéristiques sont les suivantes :

- L'orthogonalité des dimensions structurelle, spatiale et temporelle
Cette propriété est sans doute la plus importante, car elle permet de mélanger dans une même application les données classiques et les données spatio-temporelles. Héritée de l'orthogonalité, la propriété d'indépendance permet de bâtir des modèles simples et puissants(par combinaison de différentes dimensions).
- la possibilité de décrire des relations topologiques ou temporelles entre les entités(objets) de manière explicite,
- une définition formelle des concepts tels l'agrégation spatiale ou temporelle,
- la généralisation/spécialisation et les mécanismes d'héritage associés,
- la possibilité de décrire des champs spatiaux continus, grâce au concept d'attribut variable.

⁶ Y. BÉDARD, C. CARON, Z. MAAMAR, B. MOULIN, D. VALIÈRE, *Adapting Data Models for the design of Spatio-Temporal Databases*, Computer Environment and Urban Systems,20(1), 1996.

⁷ B. DAVID, L. RAYNAL, G. SCHORTER, *GeO2: Why objects in a geographical DBMS?*, Proceedings of the 3rd International Symposium on Advances in Spatial Databases, SSD'93, p. 264-276. Springer-Verlag, Lecture Notes in Computer Science 692, 1993.

⁸ N. TRYFONA, D. PFOSE, T. HADZILACOS, *Modeling Behavior of Geographic Objects: An Experience with the object modeling technique*, Proceedings of the 9th International Conference on Advanced Information Systems Engineering, CAiSE'97, Barcelone, Espagne. Springer-Verlag, 1997.

⁹ O. GAYTE, T. LIBOUREL, J.P. CHEYLAN, S. LARDON, *Conception des systèmes d'information sur l'environnement*, Hermès, Paris, 1997.

- la provision de types spatiaux génériques au-delà des types de base,
- des notations visuelles intuitives à l'aide des pictogrammes, améliorant ainsi l'ergonomie. Les aspects ergonomiques sont essentiels pour le développement d'outils CASE d'édition visuelle d'un schéma spatio-temporel ou de formulation visuelle interactive de requêtes d'interrogation et de mise à jour.

2.3.1 MADS et la Modélisation des données classiques.

Fondé sur le modèle Objet + Relation¹⁰, MADS offre tous les concepts d'usages pour la description des données classiques. Parmi ces concepts il y a notamment :

- Un type d'objet : qui représente un ensemble d'entités du monde réel ayant une structure et un comportement similaires.
- Un objet : qui représente une entité du monde réel que l'on veut modéliser
- Un type d'association : qui représente un ensemble de liens avec des caractéristiques similaires.
- Une association : qui représente un lien entre deux ou plusieurs objets.
- Un attribut : porté soit par un type d'objet ou un type d'association, il représente une propriété du monde réel.
- Les liens de généralisation et d'agrégation ainsi que des contraintes d'intégrité associée. A cela il faut ajouter le concept de « méthode ». une méthode est une opération qu'on peut effectuer les occurrences d'un type d'objet.

2.3.2 MADS et la Modélisation des données spatiales.

Il existe à ce jour, dans le monde des SIG, deux façons décrire les relations entre les objets et l'espace. Il s'agit premièrement de l'approche « discret » et de l'approche « continu ».












Dans l'approche « discret », la base de données contient les objets localisables dans l'espace. Quant à l'approche « continu », la base de données contient des régions de l'espace sur lesquelles des variables sont définies comme des champs continus des valeurs. Comme dit plus haut, le modèle MADS a été développé dans un contexte

¹⁰ S. SPACCAPIETRA AND C. PARENT, *ERC+ : An Object based Entity Relationship Approach*, In *Conceptual modeling, databases, and CASE*, P. Loucopoulos and R. Zicari (Eds.), John Wiley & Sons, 1992, p. 69-86.

d'aménagement du territoire, des eaux,... Dans ce milieu, c'est l'approche « discret » qui est le plus utilisé. Aussi, tout en offrant des concepts pour décrire une vue continue, MADS a pleinement adopté l'approche « discret »

La spatialité couvre deux aspects : l'emprise et la localisation. L' « emprise » permet de décrire la forme (point, surface,...) associée à la représentation et la « localité » donne la position de la forme dans l'espace. MADS propose des types abstraits spatiaux génériques à partir desquels l'on peut décrire l'emprise. Le Tableau¹¹ 1. donne ces différents types :

Tableau 1

type spatial	picto.	dimension	définition
géo		0, 1 ou 2	tout type spatial défini ci-dessous dans ce tableau
géosimple		0, 1 ou 2	tout type spatial simple (un point, une ligne, une ligne orientée ou une surface simple)
point		0	un point
ligne		1	toute ligne qu'elle soit droite, courbe, brisée, fermée ou non, orientée ou non
ligne orientée		1	toute ligne orientée qu'elle soit droite, courbe, brisée, fermée ou non
surface simple		2	toute surface connexe (avec éventuellement des trous)
géocomposé		0, 1 ou 2	toute composition de types spatiaux simples
semis		0	une collection de points
graphe		1	une collection de lignes
digraphe		1	une collection de lignes orientées
surface complexe		2	une collection de surfaces simples

¹¹ Source : MADS, modèle conceptuel spatio-temporel (cfr. bibliographie)

Ces types abstraits sont organisés en la hiérarchie suivante (Figure¹² 1):

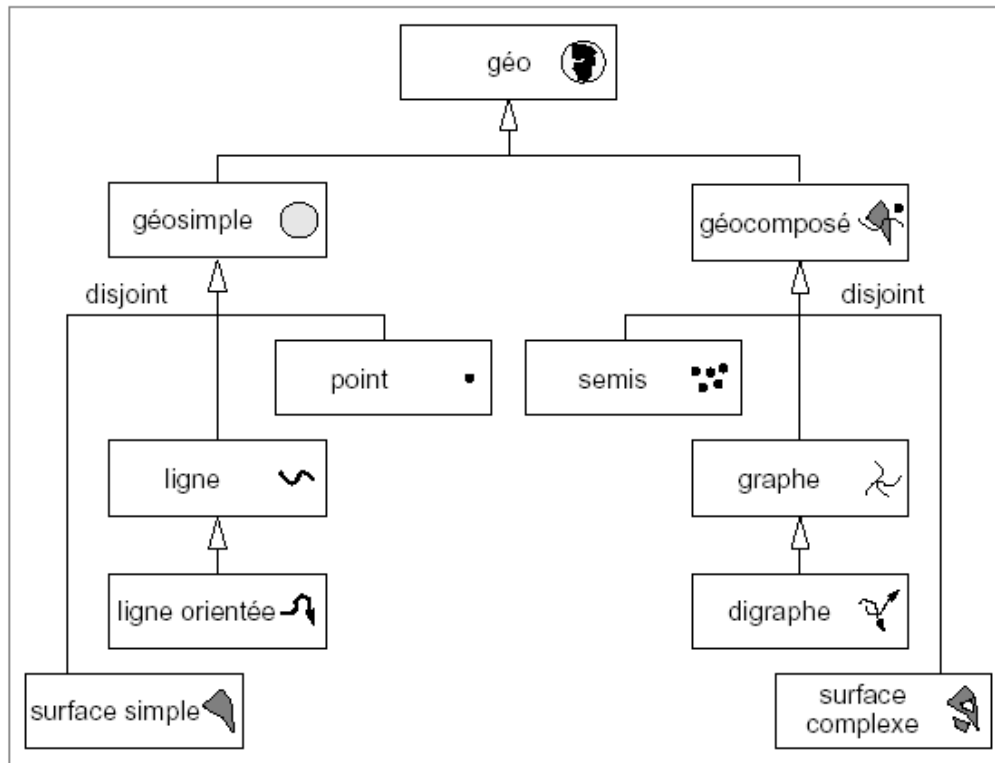


Figure 1

La spatialité peut être attachée à un attribut à un type d'objet ou à un type d'association.

Un **attribut** (simple, complexe monovalué ou multivalué) est caractérisé par une spatialité lorsque son domaine de valeurs est un type abstrait spatial. Il existe en MADS des attributs dits « variables ». Il s'agit d'attribut dont la valeur est définie par une fonction dont le domaine est l'ensemble d'éléments géométriques dans lesquels la géométrie de l'objet (ou attribut) considérée a été décomposée. Les attributs variables permettent de définir des champs spatiaux continus.







Un **type d'objet** est caractérisé par une spatialité lorsque le domaine de valeur de son attribut prédéfini « géométrie » est un type abstrait spatial (prédéfini ou défini explicitement pour l'application).

Sur les **types d'associations**, la spatialité exprime une contrainte spatiale topologique métrique, d'orientation ou d'agrégation. MADS propose deux types

¹² Source : MADS, modèle conceptuel spatio-temporel (cfr. bibliographie)

prédéfinis représentant les types d'associations les plus courantes dans les applications. Il s'agit des *associations topologiques* et des *associations d'agrégation*. Les premières décrivent une relation topologique et les secondes permettent de décrire la composition d'une entité du monde réel (s'il est décomposable en d'autres entités). Le Tableau¹³ 2 donne les différents types de contraintes topologiques définies dans MADS.

Tableau 2

type spatial	picto.	définition
disjonction		aucun partage (valable pour tout type d'objet spatial)
adjacence		partage sans intérieur commun
croisement		partage d'une partie de l'intérieur tel que la dimension de l'objet partagé est strictement inférieure à la plus grande dimension des deux objets en relation.
recouvrement		partage d'une partie de l'intérieur tel que la dimension de l'objet partagé est égale à la plus grande dimension des deux objets en relation (valable pour des types spatiaux de même dimension)
inclusion		la totalité de l'intérieur de l'un correspond à une partie de l'intérieur de l'autre
égalité		partage de la totalité de l'intérieur et de la totalité de l'enveloppe (valable pour des types spatiaux de même dimension)

Exemple de schéma spatial :

Soit une base de données contenant les informations des villes et des parcs qu'ils possèdent. Le schéma conceptuel de la base de données est la suivante :

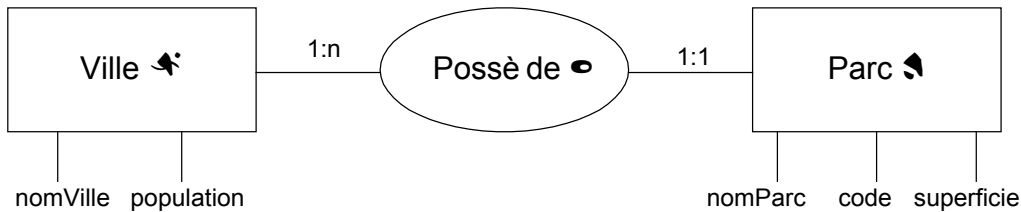


Figure 2

Dans ce schéma, les types d'objet "Ville" et "Parc" sont spatiaux, respectivement un « géocomposé » et une « surface complexe ». L'association « Possède » est également spatiale. Sa spatialité exprime une contrainte topologique d'inclusion entre « Ville » et

¹³ Source : MADS, modèle conceptuel spatio-temporel (cfr. bibliographie)

« Parc », c'est-à-dire que « Parc » *doit être contenu dans* « Ville ».






2.3.3. MADS et la Modélisation des données temporelles

Pour la modélisation temporelle, MADS travaille sur deux aspects qui sont l'estampillage et la modélisation des liens dynamiques entre objets.

L'estampillage permet de localiser dans le temps la pertinence d'une information, cet aspect est fondamental, puisqu'il permet de définir le cycle de vie ¹⁴des attributs, des objets ou des associations. La modélisation des liens dynamiques entre objets, modélisent les liens dont la sémantique possède une composante temporelle intrinsèque.

MADS définit également des types abstraits temporels qui sont donnés dans le tableau ci-dessous :

Tableau 3

Type temporel	Picto.	Définition
Temporel		Désignation générale d'un élément temporel
Instant		Désigne un instant particulier
Collection d'instant		Désigne une collection d'instant définis
Intervalle de temps		Désigne un intervalle de temps défini.
Collection d'intervalles de temps		Désigne une collection d'intervalles de temps.

¹⁴ Pour les attributs, on parlera plutôt de « période de validité », cfr. La définition d'attribut temporel

Ces types abstraits temporels sont structurés suivant la hiérarchie suivante (Figure¹⁵ 3) :

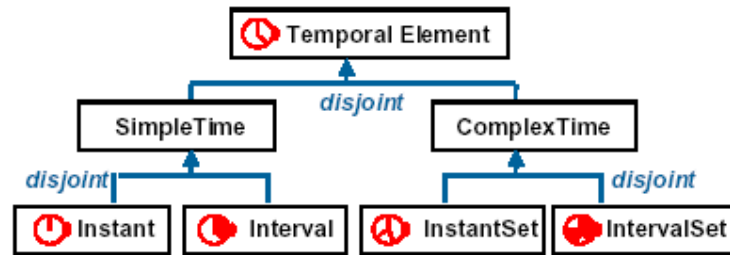


Figure 3

La temporalité peut-être attachée à un type d'objet, à un type d'association ou à un attribut.

Sur l'**attribut** (simple, complexe, monovalué ou multivalué) la temporalité exprime sa période de validité. Elle correspond à la période pendant laquelle l'information, portée par l'attribut, est valide dans le monde actif. La période de validité est définie par l'utilisateur et peut couvrir le passé, le présent et le futur (si des prévisions sont faites).

Sur un **type d'objet**, la temporalité définit son cycle de vie. Un objet est une instance d'un type d'objet, il peut migrer vers un autre type d'objet, il peut être suspendu et enfin de compte il peut être détruit. L'histoire de l'objet ainsi définie est appelée « cycle de vie ». Un type d'objet temporel garde la trace du cycle de vie de ses instances.

Sur un **type d'association**, la temporalité exprime soit son cycle de vie soit une contrainte temporelle entre les objets qu'elle lie. Une association temporelle peut relier des objets non-temporels.

Le schéma de la Figure 4. illustre un schéma conceptuel temporel. La sémantique temporelle exprime que pour chaque étudiant on désire garder son cycle de vie ainsi son cursus (une trace de ses résultats finaux, en 1^{ère} session ou éventuellement en seconde session).

¹⁵ Source : Designing Spatio-Temporal Databases (Cfr. Bibliographie)

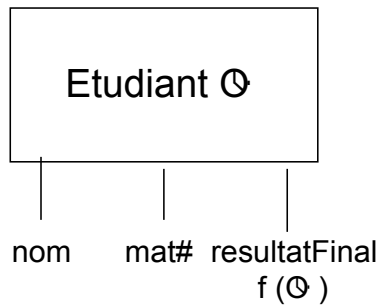


Figure 4

f (⊙) sur l'attribut « resultatFinal » indique que l'attribut est temporel. Cette notation est une convention de MADS pour la temporalité des attributs.

Bien que nous ayons défini indépendamment, suivant le principe de l'orthogonalité, les concepts de spatialité et de temporalité peuvent être intégrés dans un même schéma. Par exemple, le schéma de la Figure 5 ce schéma serai par exemple celui d'une association gérant des terrains de football. Ils aimeraient la trace des périodes de location de chaque terrain, la trace des responsables ainsi que l'historique des trophées éventuellement remportés par chaque équipe.



Figure 5

Chapitre 3 : Transformation de schémas Classiques

Chapitre 3 : Transformation de schémas classiques

Dans cette partie nous allons introduire la notion de transformation de schéma avant de décrire une transformation de schéma classique. L'étude d'une transformation classique nous donnera tous les éléments d'ordre structurel utiles pour la suite. Le modèle conceptuel utilisé dans cette partie est le modèle Entité-Association.

3.1 Définition d'une transformation :

Une transformation T est un opérateur qui remplace une construction source C d'un schéma S par une autre construction C' . C' est l'image de C par T , on notera cela par $C'=T(C)$. T est défini par des préconditions que toute construction C doit satisfaire pour pouvoir être transformée en C' .

3.2 Réversibilité d'une transformation

Une transformation T_a est réversible s'il existe une transformation T_b telle que pour toute construction C , $T_b(T_a(C)) = C$. La réversibilité est une propriété très importante. En effet, si une transformation est réversible, alors nous avons la garantie que les schémas source et cible ont la même capacité de description et décrivent le même univers du discours bien que représentés différemment.

3.3 Types de transformation

Il existe de différents types de transformation. Sur les éléments de base d'un schéma conceptuel on peut envisager les transformations suivantes :

3.3.1 Pour un attribut

- Transformation d'un attribut d'un type d'entité en type d'entité
- Transformation d'un attribut d'un type d'association en un type d'entité.

3.3.2 Pour un type d'entité

- Transformation d'un type d'entité en un type d'association.
- Transformation d'un type d'entité en un attribut d'un type d'entité
- Transformation d'un type d'entité en un attribut d'un type d'association.

3.3.3 Pour un type d'association

- Transformation d'un type d'association en un type d'entité
- Transformation d'un type d'association en un attribut

3.4 Etude des Transformations - Transformation d'attribut d'un type d'entité en type entité

L'étude¹⁶ de la transformation de schéma classique, dont nous ne présenterons ici que les résultats ainsi que les principales remarques, nous permettra de dresser cadre structurel (cardinalité des rôles, les transformations non permise,...) pour l'étude spatio-temporelle que nous envisageons. Pour être transformé, un schéma doit satisfaire les préconditions, et le schéma transformé remplit les postconditions.

Pour transformer un attribut d'un type d'entité en type d'entité il existe deux possibilités de représentation du nouveau type d'entité : il s'agit de la représentation par instance et de la représentation par valeur. Les principes de ces deux représentations vont être donnés dans les lignes qui suivent.

3.4.1 Représentation par instance

Principe :

Chaque instance de l'attribut A_i de l'entité type E_1 est représentée par une entité type E_2 .

Soit A_i l'attribut à transformer. A_i peut être simple (avec des valeurs atomiques) ou complexe (composé d'autres attributs simples ou complexes).

¹⁶ J-L Hainaut, cfr. l'aide du logiciel DB-Main .

La transformation de l'attribut simple (monovalué ou multivalué) A_i de E_1 est donnée par les schémas suivants :

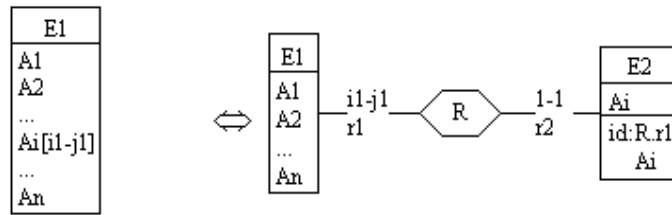


Figure 6

Si A_i est complexe (monovalué ou multivalué) la transformation se fera de la manière suivante :

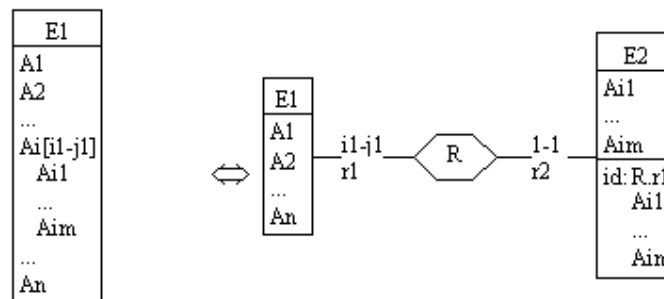


Figure 7

Préconditions

- A_i est un attribut direct (avec la cardinalité $[i_1-j_1]$), puisqu'un lien doit exister entre la nouvelle entité et l'entité-parent.
- A_i ou un sous-ensemble de ses composants (si A_i est complexe) peut être un identifiant de E_1 , mais cet identifiant ne peut pas être complexe
- A_i ou un sous-ensemble de ses composants (si A_i est complexe) ne peut faire partie d'un groupe impliqué dans une contrainte (référentielle, d'inclusion, d'égalité,...), à moins que seul A_i ou ses composants ne constitue(nt) le groupe en question.
- Si A_i est un attribut multivalué, A_i ou un sous-ensemble de ses composants ne peut pas être un identifiant de E_1 .
- Si A_i est un attribut multivalué, il ne peut être une collection ordonnée (liste ordonnée, ou autre), car on ne peut conserver cet ordre après transformation en

essayant d'avoir par exemple une liste ordonnée d'associations.

- Si A_i est un attribut multivalué, il ne peut être une collection acceptant de doublons. Il s'agit de la transformation par instance, après transformation A_i devient identifiant de E_2 , or cela n'est pas possible si deux instances de A_i ont la même valeur.
- Si A_i est un attribut multivalué complexe et qu'un attribut de A_i est un identifiant de E_1 , cet attribut doit aussi être un identifiant de A_i .

Postconditions

- Créer une entité type E_2 , un type d'association R , un rôle r_1 [i_1-j_1] entre E_1 et r , et un rôle r_2 [1-1] entre E_2 et R .
- Les groupes de E_1 qui contiennent seulement A_i (ou un sous-ensemble de ses composants) sont transférés dans E_2 .
- Dans les autres groupes de E_1 , remplacer A_i (ou un sous-ensemble de ses composants) par r_2
- Si A_i (ou un sous-ensemble de ses composants) un identifiant secondaire de E_2 et que E_2 n'a aucun identifiant primaire, l'identifiant secondaire devient primaire.
- A_i est transféré dans E_2 (avec la cardinalité [1-1). Si A_i est complexe, ses valeurs des méta-propriétés sont propagées dans E_2 .

3.4.2 Représentation par valeur.

Principe :

Chaque valeur unique de l'attribut A_i dans l'entité type E_1 est représentée par une entité type E_2 .

La transformation de l'attribut simple (monovalué ou multivalué) A_i de E_1 est donnée par les schémas suivants :

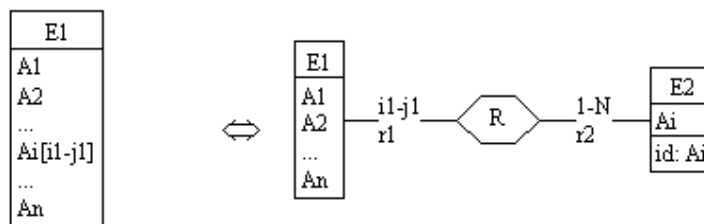


Figure 8

Si A_i est complexe (monovalué ou multivalué) la transformation se fera de la manière suivante :

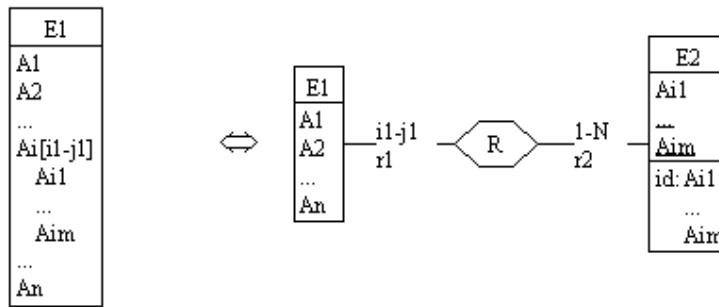


Figure 9

Préconditions

- Ai doit être un attribut direct (avec la cardinalité [i1-j1]), puisqu'un lien doit exister entre la nouvelle entité et l'entité-parent.
- Ai ou un sous-ensemble de ses composants (si Ai est complexe) ne peut faire partie d'un groupe impliqué dans une contrainte (référentielle, d'inclusion, d'égalité,...), à moins que seul Ai ou ses composants ne constituent le groupe en question.
- Ai ne peut avoir un composant identifiant de E1.
- Si Ai est un attribut multivalué, il ne peut être une collection ordonnée (liste ordonnée, ou autre), car on ne peut conserver cet ordre après transformation en essayant d'avoir par exemple une liste ordonnée d'associations.

Postconditions

- Créer une entité type E2, un type d'association R, un rôle r1 [i1-j1] entre E1 et r et un rôle r2 [1-N] entre E2 et R .
- Les groupes de E1 qui contiennent seulement Ai (ou un sous-ensemble ses composants) sont transférés dans E2.
- Dans les autres groupes de E1, remplacer Ai (ou un sous-ensemble de ses composants) par r2
- Si E2 n'a pas d'identifiant primaire, transformer l'identifiant secondaire existant en identifiant primaire ou créer un identifiant primaire contenant Ai.
- Si Ai est transféré dans E2 (avec la cardinalité [1-1). Ses valeurs des méta-propriétés sont propagées dans E2.

3.4.3 Exemple de transformation de schéma classique.

Nous allons illustrer la transformation de schéma classique à l'aide de l'outil CASE¹⁷ DB-MAIN¹⁸.

Nous nous proposons de transformer l'attribut « adresse » de l'entité-type « Employé » en entité-type par a représentation par instance. Chaque principale étape d la transformation est illustrée par une capture d'écran.

Le schéma conceptuel initial, où tous les attributs sont simples monovalués, es le suivant :

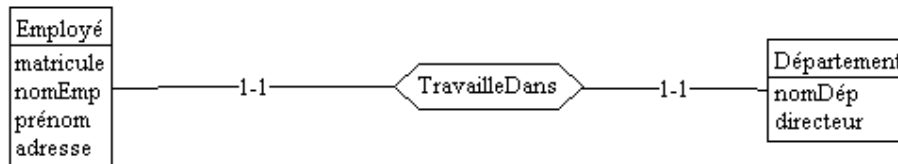


Figure 10

Etape 1.

Nous cliquons sur l'attribut « adresse » de «Employé » et ensuite, nous allons sur le menu « Transform/Attribute/--To Entity-type ». une boîte de dialogue est alors affichée. La boîte de dialogue propose le choix du principe de représentation (par instance ou par valeur). Nous choisissons « instance representation ».

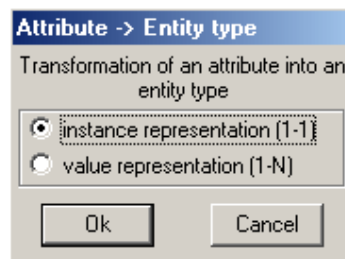


Figure 11

¹⁷ Computer Assisted Software Engineering : outil d'aide à l'ingénierie.

Etape 2.

Une nouvelle boîte de dialogue est alors affichée : elle permet de modifier les propriétés de base du nouveau type d'entité. Nous nommons le nouveau type d'entité « Adresse », et cliquons ensuite sur « OK ».

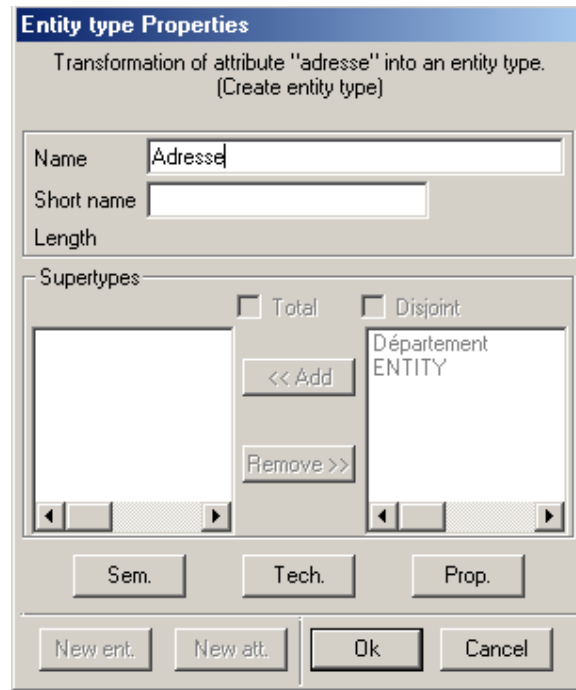
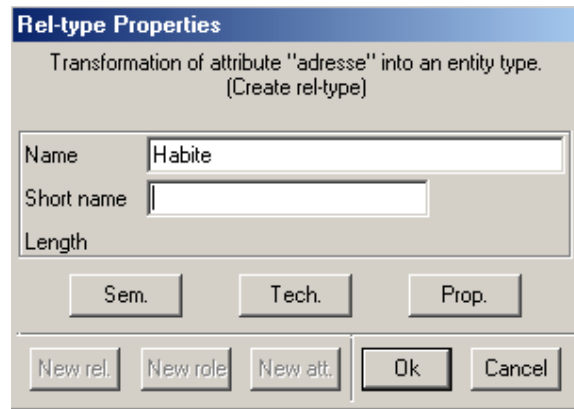


Figure 12

Etape 3.

Une dernière boîte de dialogue est affichée : elle permet de modifier les propriétés de base du nouveau type d'association. Nous donnons à la nouvelle association le nom de « Habite », et cliquons sur « OK ».

¹⁸ cfr. la partie bibliographique pour une référence complète



The dialog box is titled "Rel-type Properties" and contains the following elements:

- Title: Transformation of attribute "adresse" into an entity type. (Create rel-type)
- Name: Habite
- Short name: (empty)
- Length: (empty)
- Buttons: Sem., Tech., Prop.
- Buttons: New rel., New role, New att., Ok, Cancel

Figure 13

Et finalement le schéma transformé (Figure 14.) est affiché en lieu et place de l'ancien.

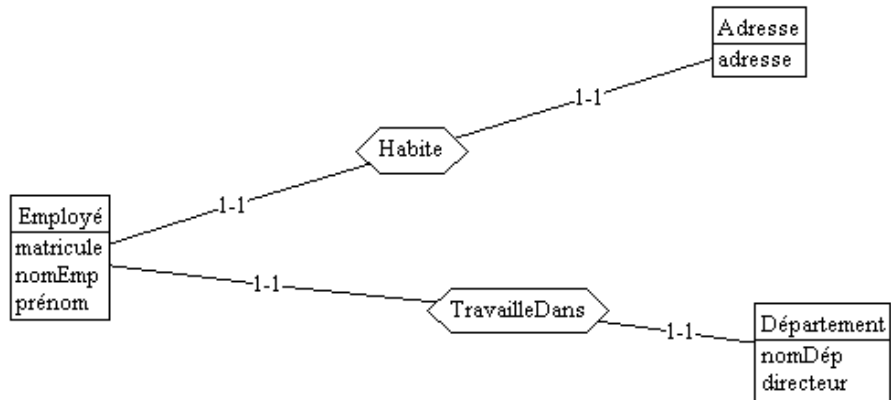


Figure 14

Chapitre 4 : Transformation de schémas
spatio-temporels

Chapitre 4 : Transformation de schémas spatio-temporels

4.1 Introduction

Dans le précédent chapitre nous avons fait le point sur la transformation de bases de données, ou plus exactement des schémas dits « classiques ». A présent, nous allons entreprendre une étude sur la transformation de bases de données spatio-temporelles. Cette étude est l'un des objectifs-clés de ce travail, tout d'abord parce qu'elle n'a jamais été entreprise antérieurement, ensuite parce qu'elle est nécessaire à l'implémentation des fonctionnalités de Transformation. Plusieurs éléments de conclusion du cas classique peuvent être appliqués au cas spatio-temporel du fait de l'indépendance des dimensions (structurelle, spatiale et temporelle)

Cependant, la « spatio-temporalité » élargit considérablement la sémantique des schémas. De ce fait, des précautions particulières doivent être prises afin de conserver l'expression du schéma initial, tout en offrant un schéma transformé clair et conforme à l'esprit et à la sémantique du modèle MADS.

L'orthogonalité des dimensions structurelle, spatiale et temporelle du modèle MADS, nous permet de mener séparément une étude temporelle et une étude spatiale de la Transformation.

De prime à bord, certaines questions peuvent déjà être posées, notamment ce qu'il en adviendra des contraintes sur un élément transformé, de ses méthodes et surtout la question clé : « **que fait-on de ses propriétés spatio-temporelles ?** »

Pour y arriver, nous adopterons une approche pragmatique, c.-à-d. partant des cas pratiques, nous en déduirons des règles générales ainsi les conditions qui nous y ont conduit. Notre réflexion sera guidée par le souci de conserver la sémantique du schéma.

Pour rappel, les conventions suivantes ont été adoptées :

Les noms des attributs commencent par des minuscules et ceux des types d'objets ou d'associations par des majuscules.

Le terme « classique » désigne tout ce qui n'est ni spatial, ni temporel.

Tous les exemples sont indépendants les uns des autres, à moins que la référence à un exemple ne soit explicitement signalée.

La coïncidence de tout nom, de toute valeur ou de tout chiffre d'un exemple à un autre est tout à fait fortuite et n'a donc aucune implication. Cette étude sera faite sur base du modèle spatio-temporel MADS, fondé sur le modèle « Objet + Relation ».

Nous allons étudier la **transformation d'attribut d'un type d'objet en type d'objet**. Nous avons choisi ce cas parce qu'il fait intervenir plusieurs concepts permettant d'aborder assez aisément tous les autres types de transformation.

Nous admettrons les deux types de représentation (par instance et par valeur) énoncées dans le cas « classique ». L'étude se fera progressivement, partant d'attribut simple monovalué pour aboutir à l'attribut complexe multivalué.

4.2 Transformation d'attribut temporel

4.2.1 Attribut simple monovalué

Soit le schéma ci-contre :

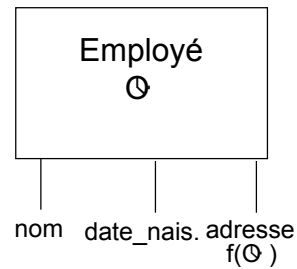


Figure 15

Admettons que nous disposions des données (deux objets e1 et e2) structurées de la manière suivante :

e1	Jean 11/09/1967	08/1993-06/1998
	bld St Michel [08/1993-12/1994] rue des éléphants [01/1995-05/1996] rue des oliviers [06/1996-06/1998]	
e2	Pierre 5/06/1962	05/1990-01/1994
	avenue des celtes [05/1990-08/1991] bld St Michel [09/1991- 01/1994]	

Figure 16

Proposons-nous de transformer l'attribut « adresse » en type d'objet. Par rapport à l'héritage des propriétés temporelles de « adresse », il y a deux solutions. La première consiste à affecter les propriétés temporelles de « adresse » sur l'association et la seconde à affecter les propriétés temporelles sur le nouveau type d'objet.

Dorénavant, nous appellerons respectivement « solution 1 » et « solution 2 », la solution qui consiste à affecter les propriétés temporelles au type d'association créée et celle qui consiste à affecter les propriétés temporelles sur le nouveau type d'objet.

Solution 1.

a. Représentation par instance

Le schéma transformé est le suivant :

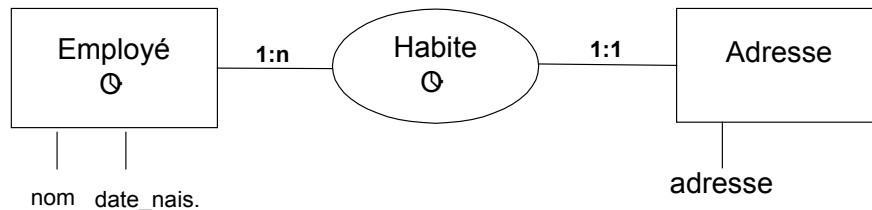


Figure 17

Conformément à la sémantique du schéma transformé les données peuvent être représentées de la manière suivante :

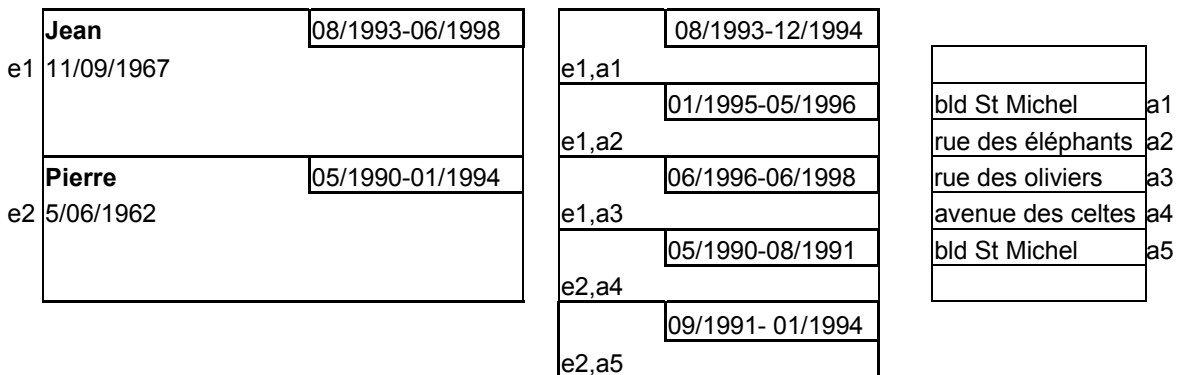


Figure 18

Comme le montre la Figure 18., cette transformation conserve la sémantique mais en introduisant des redondances des valeurs de « adresse » (dans « Adresse »).

b. Représentation par valeur

Le schéma transformé est le suivant :

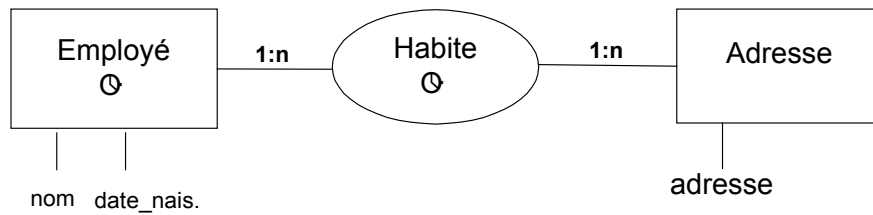
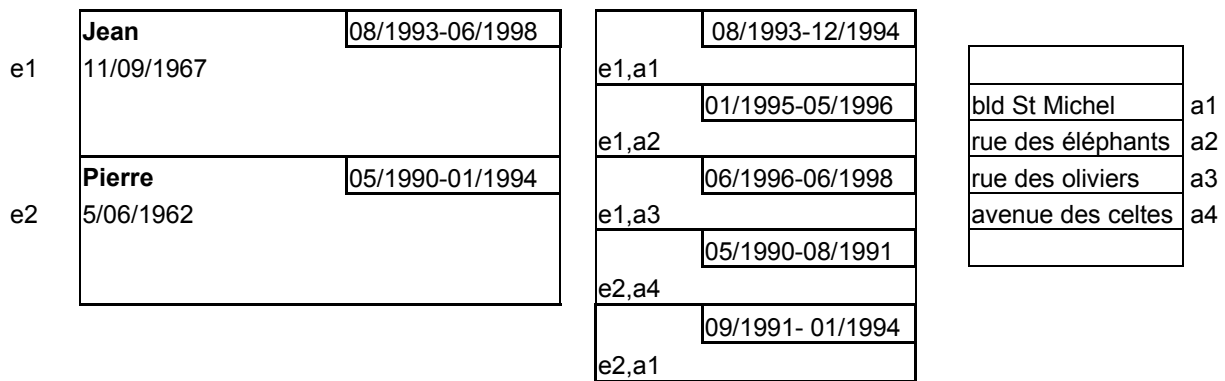


Figure 19

Les données seront structurées de la manière suivante :

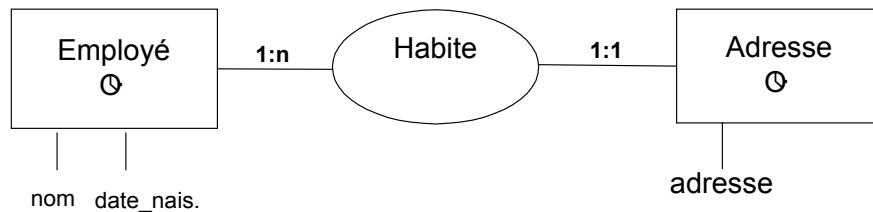


Cette représentation conserve la sémantique et possède l'avantage d'être plus économique, puisqu'on ne prend que les valeurs uniques des instances d'«adresse ».

Solution 2

a. Représentation par instance

Le schéma transformé est représenté ci-dessous :



Et les données seront structurées de la manière suivante :

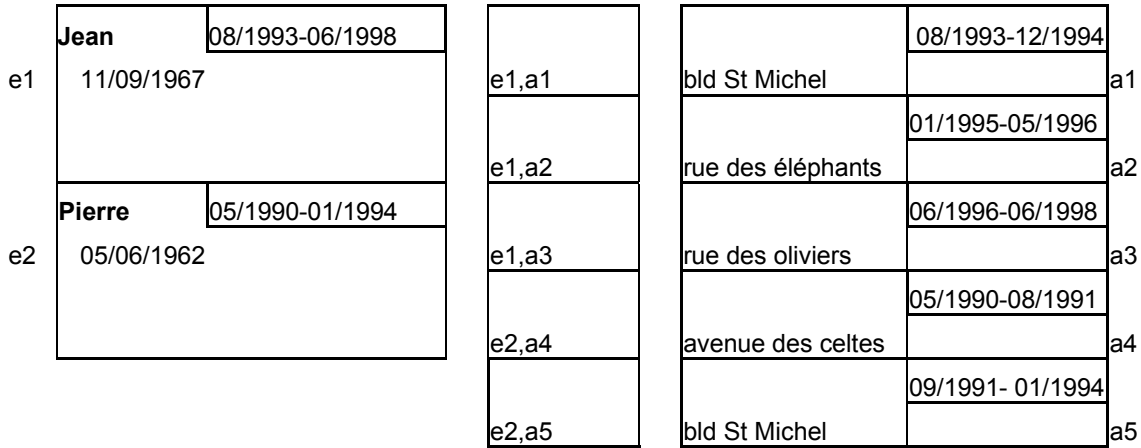


Figure 20

Nous voyons que cette transformation conserve également la sémantique.

b. Représentation par valeur

Le schéma transformé suivant la représentation par valeur est la suivante :

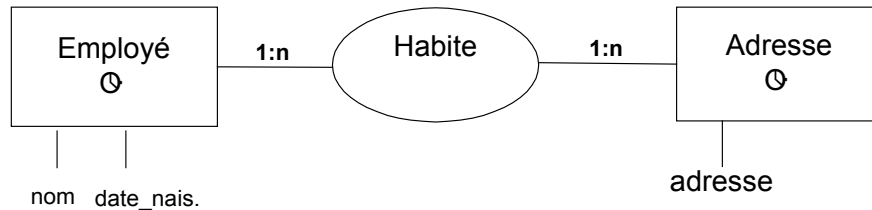


Figure 21

Cette solution pose problème. Pour deux valeurs identiques d' « adresse », quelle période de validité représentera le cycle de vie du nouvel objet ? Dans notre exemple il existe deux instances avec la valeur « bld St Michel » mais chacune avec sa propre période de validité. Une solution serait de faire pour chaque valeur unique, l'union de ses périodes de validité. L'union des périodes de validité ainsi obtenue, donnerait le cycle de vie de l'objet représentant ladite valeur. Bien évidemment, *par rapport au schéma initial, la sémantique n'est plus la même*, le cycle de vie représente maintenant le moment où un employé (quelconque) a habité à l'adresse dont il est question. Les données peuvent alors être représentées sous la structure suivante :

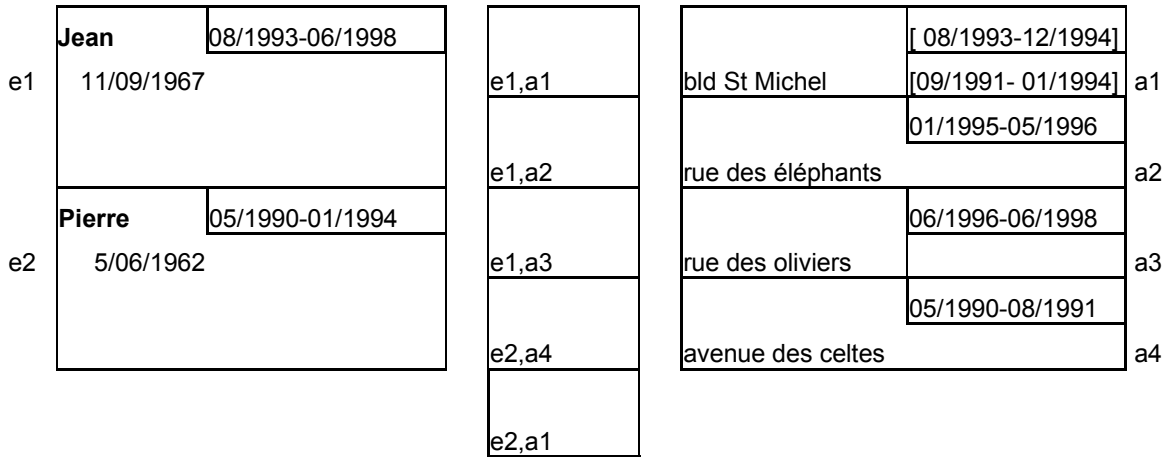


Figure 22

Cette transformation modifiant la sémantique, l'utilisateur devra en être averti !

4.2.2 Attribut simple multivalué

Soit le schéma ci-contre:

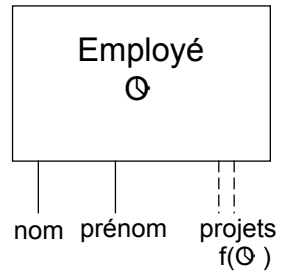


Figure 23

Admettons que nous ayons des données structurées de la manière suivante :

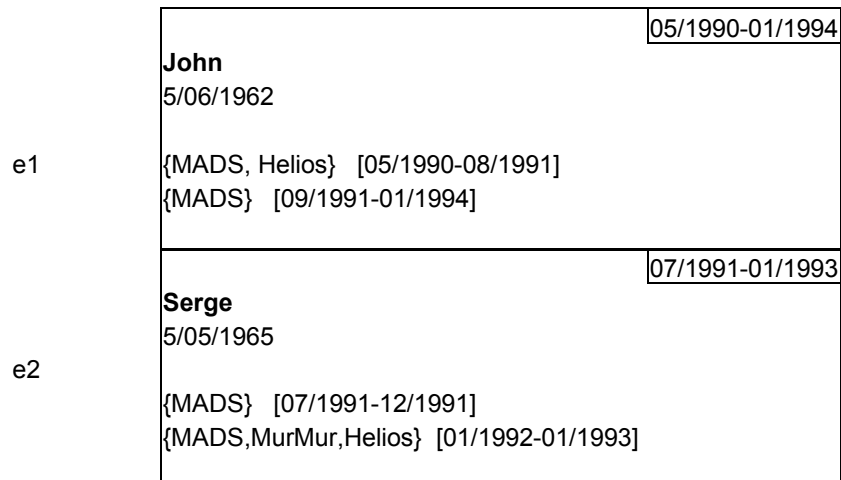


Figure 24

Proposons-nous de transformer l'attribut « projets » en type d'objet.

Solution 1

a. Représentation par instance

Le schéma transformé est me suivant :

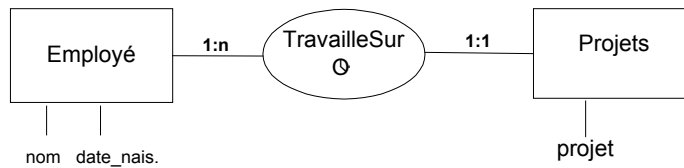


Figure 25

Et les données seront structurées de la manière suivante :

e1	John 5/06/1962	05/1990-01/1994	e1,p1	05/1990-08/1991	MADS	p1
			e1,p2	05/1990-08/1991	Helios	p2
e2	Serge 5/05/1965	07/1991-01/1993	e1,p3	09/1991-01/1994	MADS	p3
			e2,p4	07/1991-12/1991	MADS	p4
			e2,p5	01/1992-01/1993	MADS	p5
			e2,p6	01/1992-01/1993	Murmur	p6
			e2,p7	01/1992-01/1993	Helios	p7

Figure 26

Cette solution préserve bien la sémantique. Cependant, elle n'est pas de plus économique : il y a redondance des valeurs de « projets ». Cette redondance est d'autant plus dramatique que l'attribut « projets » (dans « Employé ») est multivalué.

b. Représentation par valeur

Le schéma transformé selon la représentation par valeur est la suivante :

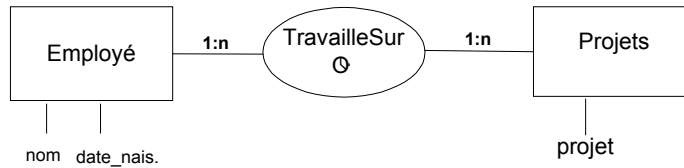


Figure 27

La structure de données correspondante et la suivante :

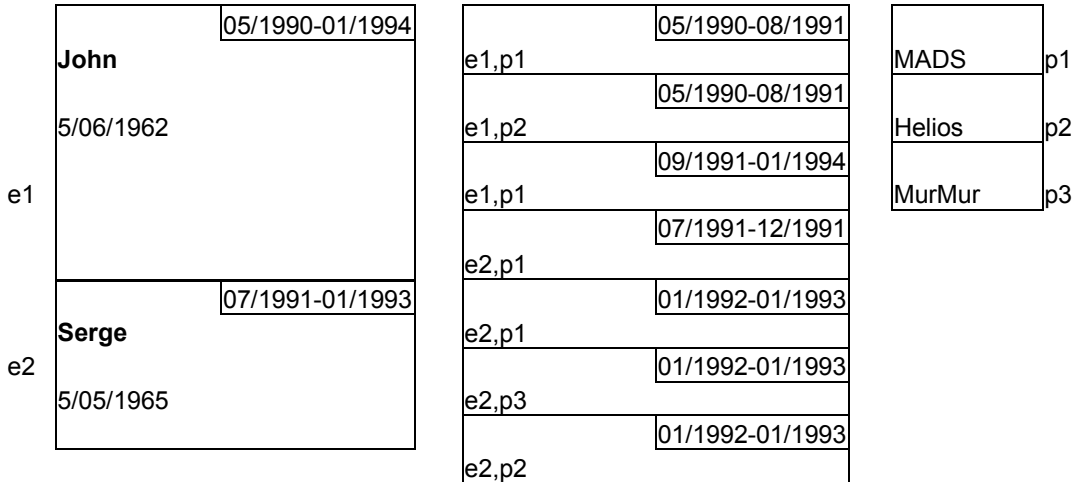


Figure 28

Cette transformation préserve la sémantique et a en plus l'avantage d'être économique. Une amélioration pourrait consister à réunir au niveau de l'association, les cycles de vie des couples (Employé, Projets) identiques. En appliquant ce raisonnement on obtient la structure suivante :

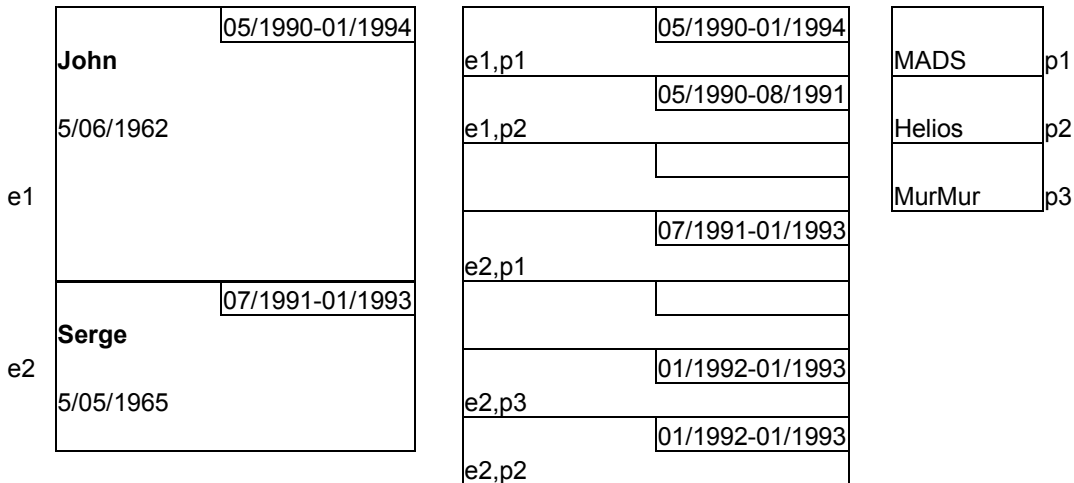


Figure 29

Cette structure est encore plus compacte. Mais elle exige, comme on vient de le voir, une opération supplémentaire.

Solution 2

a. Représentation par instance

Le schéma transformé est le suivant :

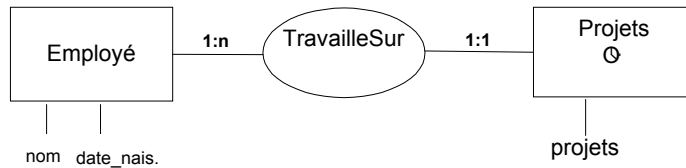


Figure 30

La structure de données correspondante est la suivante :

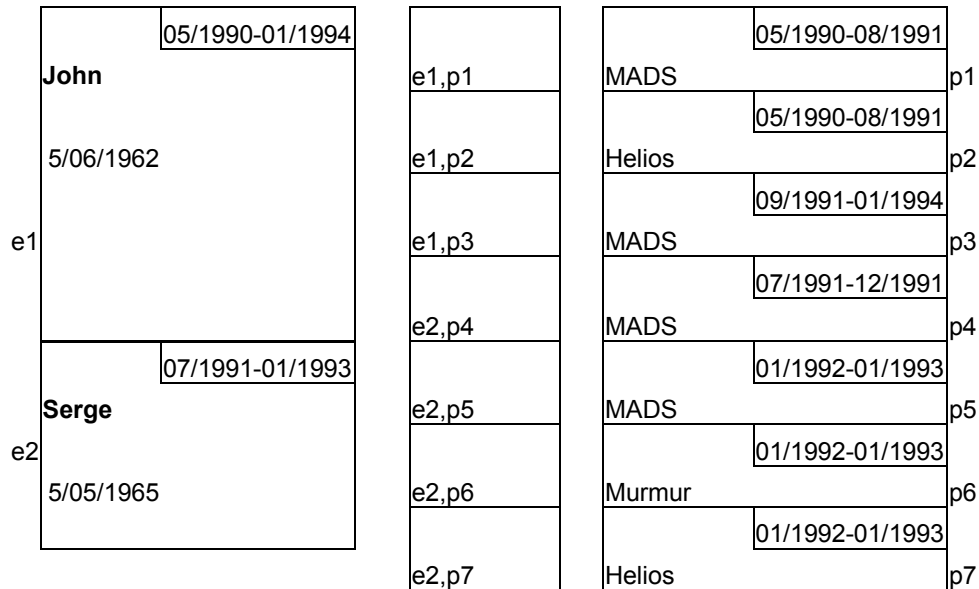


Figure 31

La sémantique est préservée, mais il y a redondance des valeurs de « projets » (dans « Projets »). Une amélioration serait de regrouper (union des cycles de vie de « Projets ») les valeurs des projets identiques associés au même employé.

b. Représentation par valeur

Le schéma transformé est donné ci-dessous :

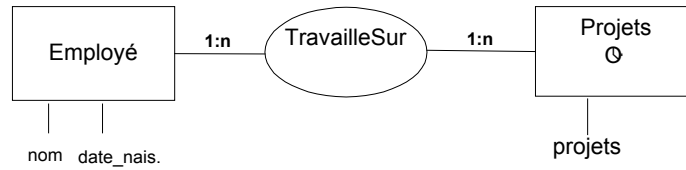


Figure 32

Pour la représentation par valeur avec la « solution 2 » , nous rencontrons la même difficulté que dans le cas de l'attribut simple. Au prix d'un changement de sémantique, nous proposons la solution qui consiste pour chaque projet, à unir les périodes de validité qui lui ont été associées . Le cycle ainsi obtenu constituerait le cycle de vie du « projet », dans le sens où ce cycle correspond à toutes les périodes où au moins un employé était affecté au projet.

La structure de données suivante donne le résultat obtenu par cette solution alternative:

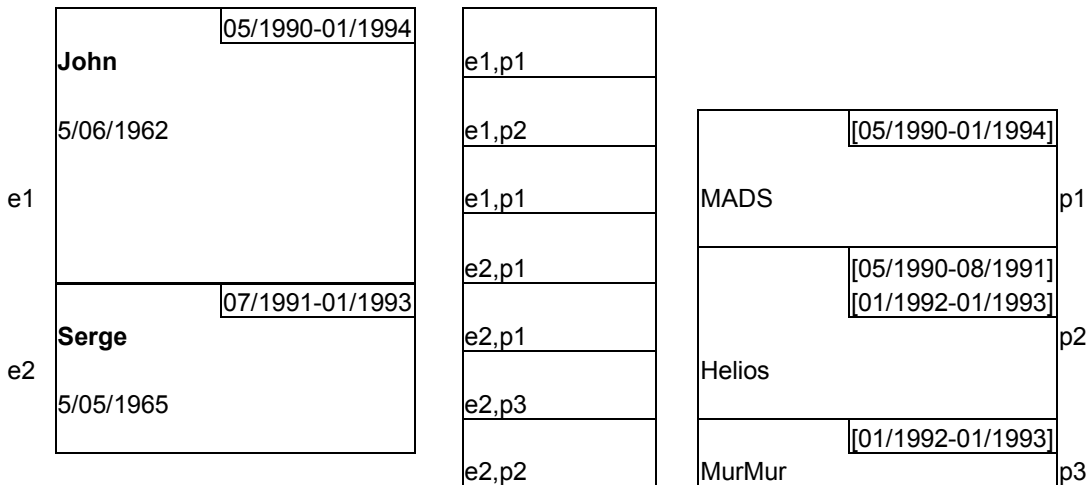


Figure 33

4.2.3. Attribut complexe monovalué – la temporalité sur l'attribut « parent »

Nous nous servons du schéma ci-dessous, où l'on désire garder la trace des services (nom du service, nom du manager).

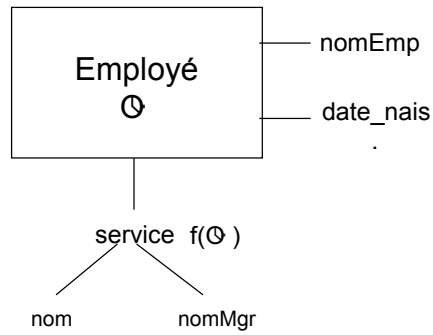


Figure 34

Admettons que nous possédons les données suivantes :

e1	Jean	01/1995-06/1998
	10/06/1970 {Marketing,Didier} [01/1995-07/1994] {HR,Robert} [08/1994-10/1995]	
e2	Luc	06/1993-07/1998
	19/02/1967 {HR,Robert} [06/1993-12/1995] {HR,Alain} [01/1996-07/1998]	

Figure 35

Nous nous proposons de transformer l'attribut temporel « service » en type d'objet.

Solution 1

a. Représentation par instance

Le schéma transformé se présente de la façon suivante :

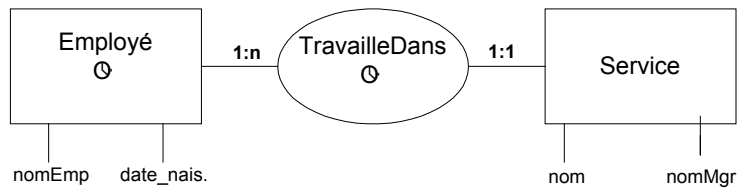


Figure 36

Et la structure des données correspondante est :

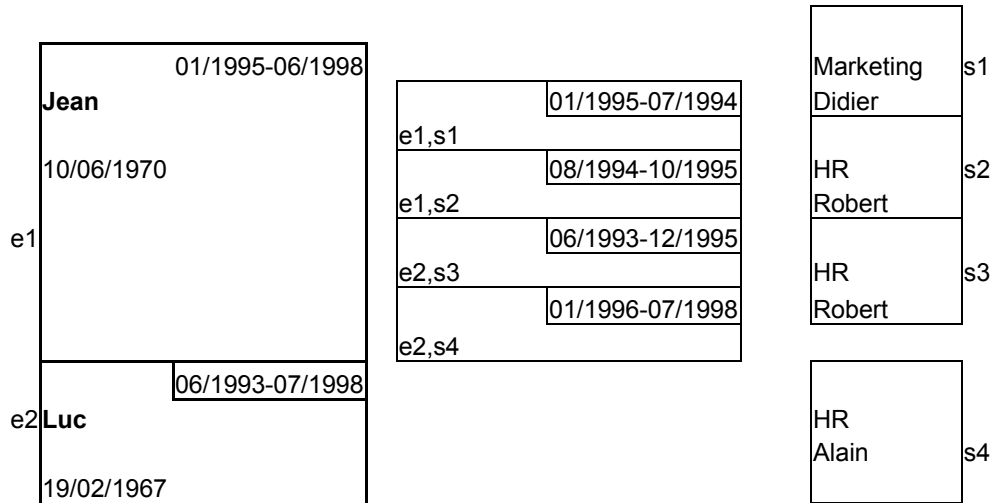


Figure 37

La sémantique est conservée, mais il existe une double (parce que l'attribut complexe possède deux composants) redondance de « service ». Par exemple, le couple (HR, Robert) apparaît deux fois.

b. Représentation par valeur

Le schéma transformé est le suivant :

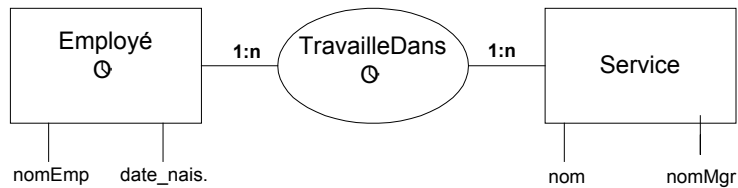


Figure 38

La structure de données ainsi obtenue est :

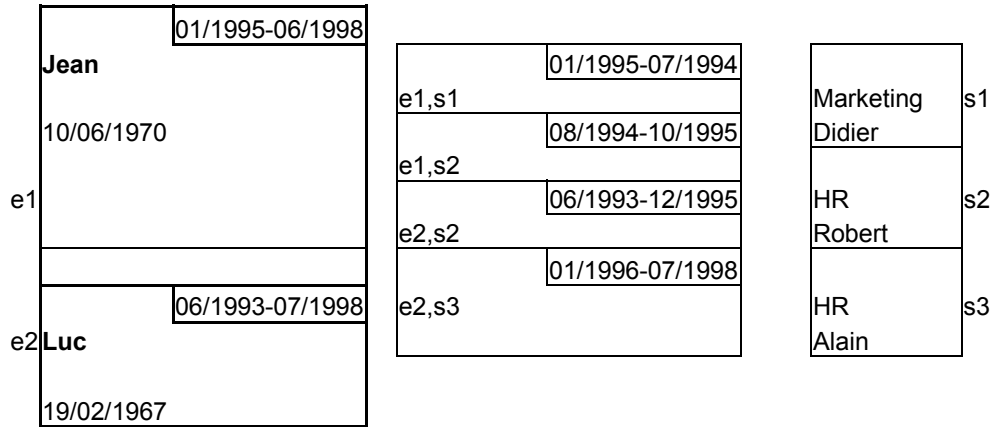


Figure 39

L'efficacité de la représentation par valeur se vérifie encore une fois par l'élimination des redondances. De plus, la sémantique est conservée.

Solution 2

a. Représentation par instance

Le schéma transformé est le suivant :

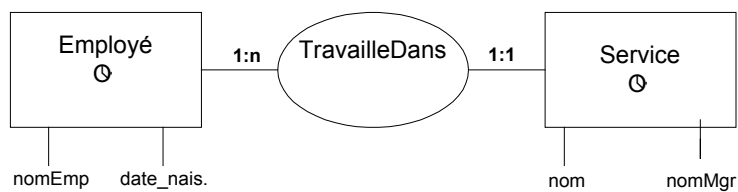


Figure 40

La structure de données est :

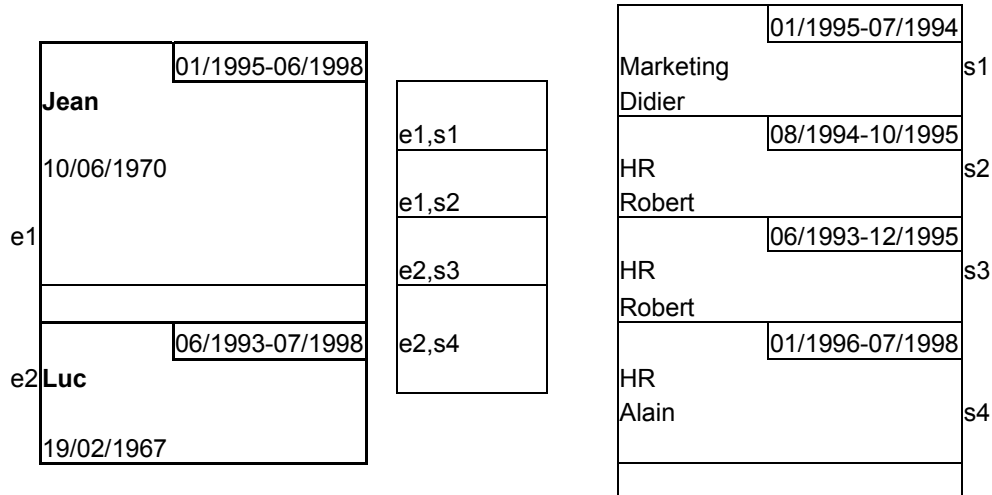


Figure 41

La sémantique est conservée malgré les redondances .

b. Représentation par valeur

Le schéma transformé est le suivant :

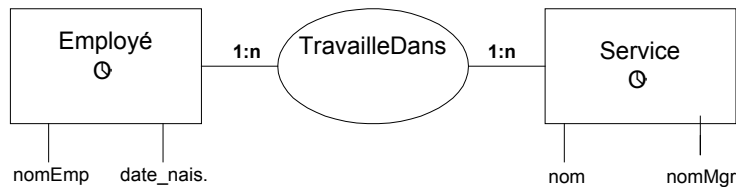


Figure 42

Comme vu dans les cas précédents, la représentation par valeur suivant la « solution 2 » ne respecte pas la sémantique. Des solutions alternatives peuvent être proposées suivant les objectifs poursuivis. Nous ne détaillerons donc pas la structure de données dans ce cas-ci.

4.2.4 Attribut complexe monovalué – la temporalité sur un composant

Nous allons à présent examiner le cas où la temporalité est associée à un attribut

composant de l'attribut complexe. Nous nous servirons du schéma suivant (où l'on désire garder l'historique des « managers » des services):

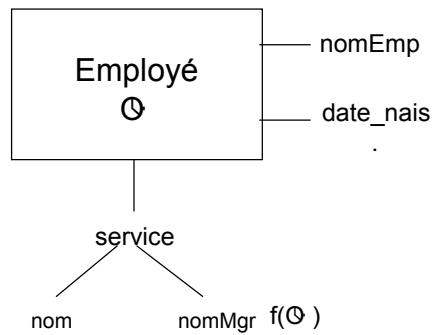


Figure 43

Admettons que nous possédions les données suivantes :

e1	Jeanine	01/1995-06/1998
	18/06/1971	{HR, Stef [01/1995-08/1996] John [08/1996-06/1998]}
e2	Marie	06/1993-07/1998
	7/02/1967	{HelpDesk Chris [06/1993-07/1996] Yan [08/1996-07/1998]}

Figure 44

Solution 1

a. Représentation par instance

Le schéma transformé serait le suivant :

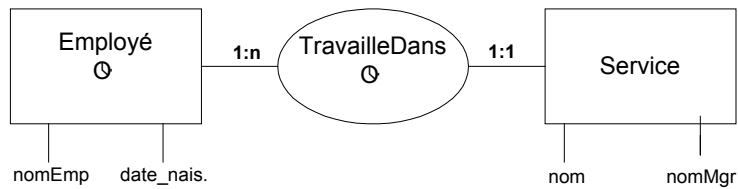


Figure 45

Avec la structure de données ci-après :

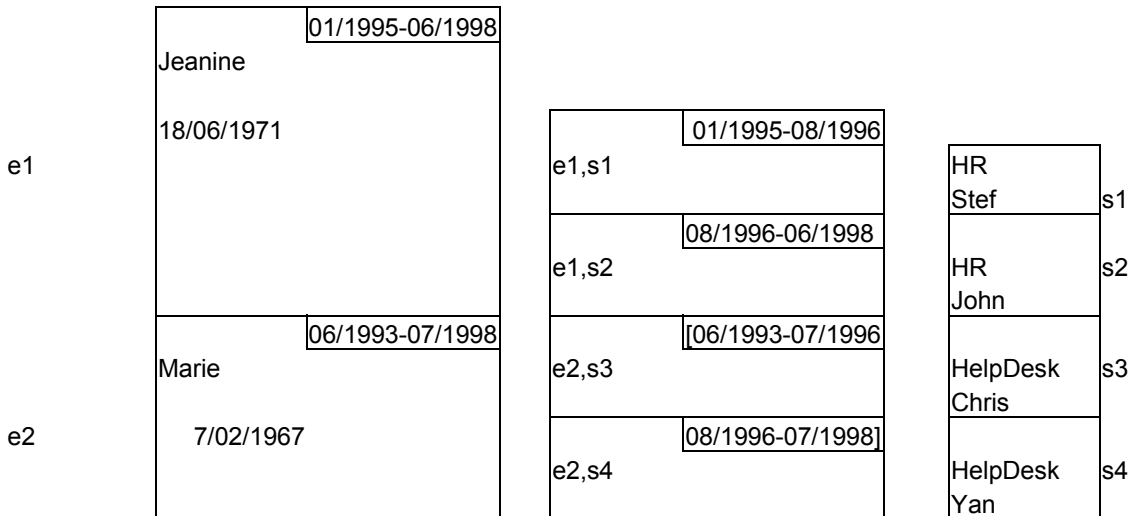


Figure 46

L'information n'est plus du tout la même qu'initialement. Ici, le cycle de vie donne la période pendant laquelle l'employé a travaillé dans « service » qui était dirigé par « nomMgr ». Ce problème ne sera résolu ni par la représentation par valeur, ni par la « solution 2 ».

La seule alternative serait d'affecter la temporalité à l'attribut « nomMgr » (de « Service »). Nous appèlerons cette issue « solution 3 »

Solution 3

D 'après ce que nous venons d'énoncer, le schéma transformé serait le suivant :

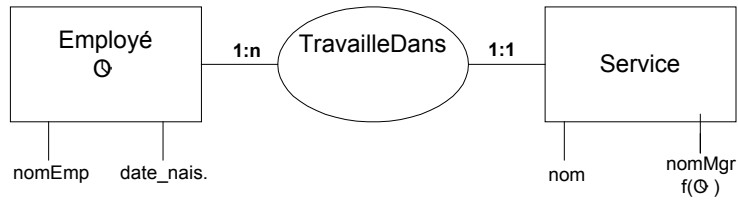


Figure 47

Cette dernière solution respecte la sémantique, puisque l'attribut temporel « nomMgr » est reproduit avec ses propriétés temporelles initiales.

4.2.5 Attribut complexe multivalué

Soit le schéma ci-dessous :

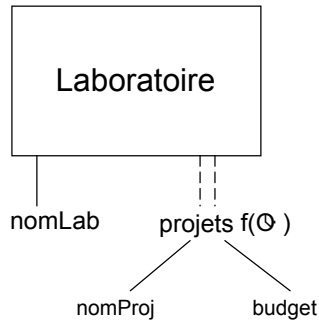


Figure 48

Supposons que nous disposions des données suivantes :

	DataLab
lab1	{(MADS,1500) [1/1996-08/1998], (MADS,1500),(Helios,3000) [09/1998-now]}
lab2	SOM {(EUROPA1,3000) [11/1997-10/2000], (EUROPA1,3000),(BO7,2000) [11/2000-now]}

Figure 49

Solution 1

a. Représentation par instance

Le schéma transformé est le suivant :

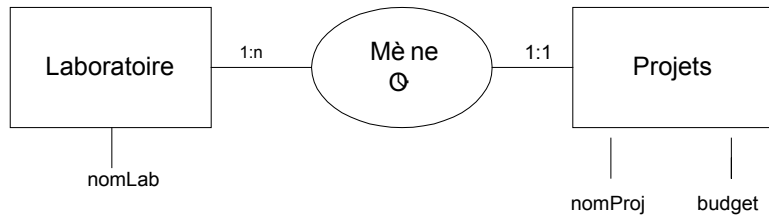


Figure 50

Les données seront structurées de la manière suivante :

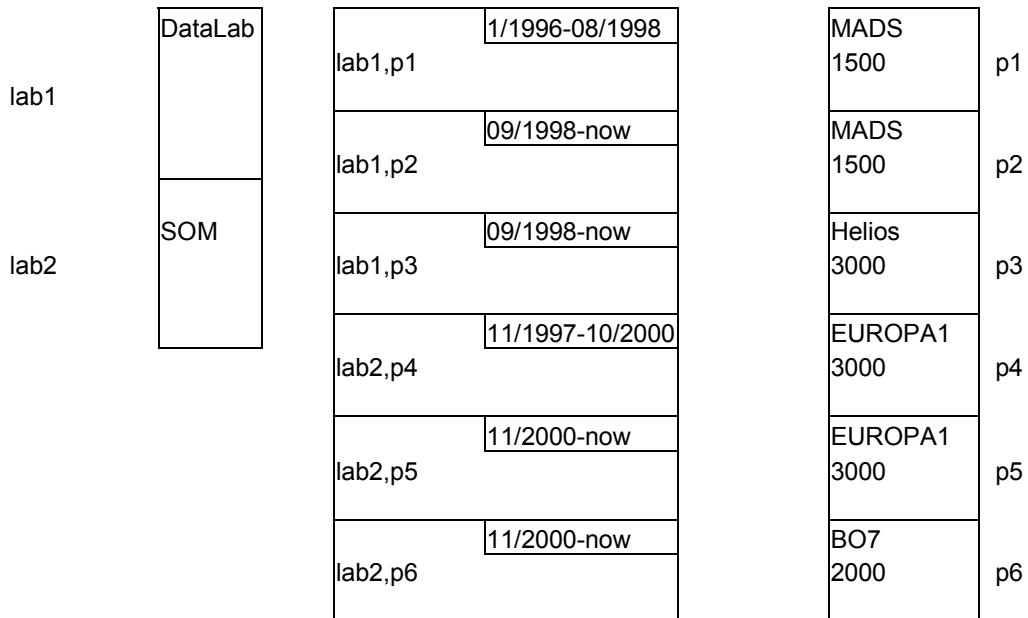


Figure 51

Cette transformation conserve la sémantique, mais avec des redondances dans « projets ». Comme précédemment ces redondances peuvent être évitées avec la représentation par valeur.

La « solution 2 » ainsi que ses variantes sont qualitativement similaires au cas de l'attribut complexe monovalué. Nous ne les détaillerons donc pas.

Remarques générales

Un problème se pose pour la transformation d'attribut temporel de granularité de type « continu ». En effet, pour ce type d'attribut, sa valeur est déterminée par une fonction d'interpolation. Par rapport à l'approche que nous avons adoptée, nous ne pouvons dans ce cas définir explicitement (directement) c'est qu'une valeur. D'où la difficulté de procéder aux transformations.

Ce serait le cas par exemple d'une Station de mesure décrite par le schéma suivant :

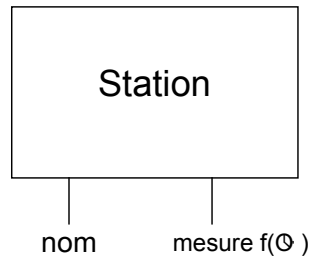


Figure 52

Pour transformer l'attribut « mesure », une solution serait, si la fonction d'interpolation ne dépend que de l'attribut « mesure » de placer la temporalité sur l'attribut « mesure » de « Mesure ». Le schéma transformé serait le suivant :

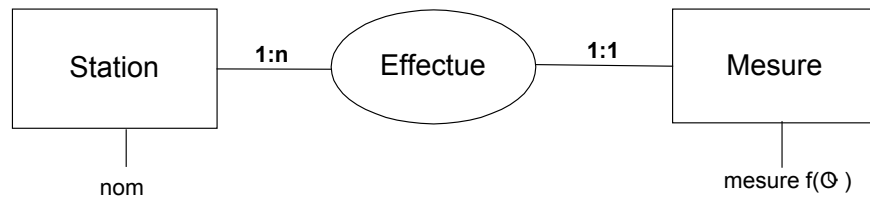


Figure 53

La décision d'effectuer cette transformation sera laissée à l'utilisateur.

4.3 Transformation d'attribut spatial

Nous allons à présent étudier la transformation d'attribut spatial, en se basant toujours sur les deux principes de représentation (par instance et par valeur).

A la différence de la temporalité, la spatialité sur un type d'objet et sur un type d'association n'ont pas la même sémantique. Alors que sur un type objet elle exprime une propriété spatiale intrinsèque, sur un type d'association elle exprime plutôt une contrainte d'intégrité spatiale (topologique, métrique, d'orientation et d'agrégation).

De plus, un type d'association spatiale doit relier au moins deux types d'objet spatiaux. Donc si l'attribut à transformer est spatial et le type d'objet le portant n'est pas spatial, après transformation la spatialité ne pourra être affectée au type d'association (parce que le type d'objet-parent n'est pas spatial). Pour toutes ces raisons, la spatialité sera toujours affectée au nouveau type d'objet.

Mise à part l'aspect structurel (représentation par instance ou par valeur), les propriétés spatiales d'un attribut transformé sera par défaut affecté au type d'objet, et l'on reportera les éventuelles contraintes spatiales sur le type d'association.

Voyons à présent quelques transformations d'attributs spatiaux en type d'objet.

4.3.1 Attribut simple monovalué

Soit le schéma conceptuel ci-dessous, où le type d'objet « Pays » et l'attribut « capitale » sont spatiaux.

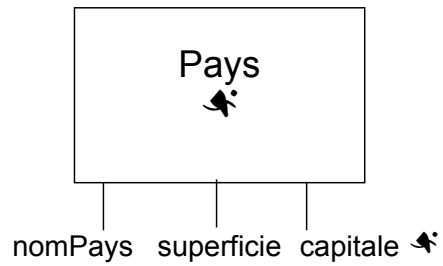


Figure 54

Admettons que nous ayons les données ci-contre :

	Belgique
p1	30 500
	Bruxelles
	France
p2	544 000
	Paris

Figure 55

Nous nous proposons de transformer l'attribut « capitale » en type d'objet

a. Représentation par instance

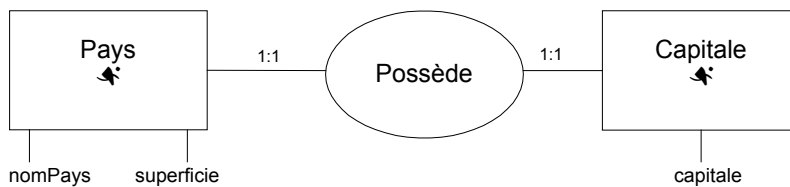


Figure 56

Les éventuelles contraintes entre l'attribut « capitale » et le type d'objet « Possède » seraient reportées sur le type d'association « Possède ».

Les données sont alors restructurées de la manière suivante :

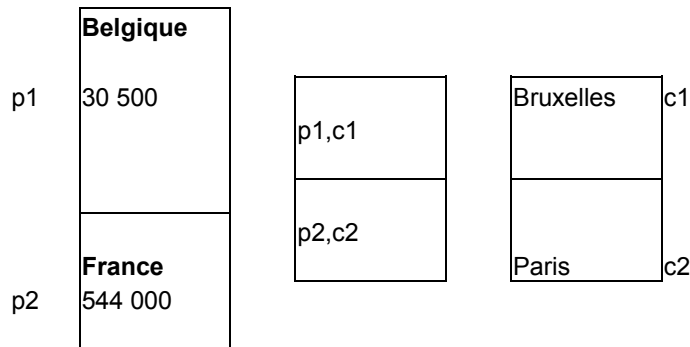


Figure 57

Cette transformation préserve la sémantique, avec l'avantage que les nouveaux objets gardent les propriétés spatiales propres à chaque instance d'attribut, même si leurs valeurs sont les mêmes. Si dans notre exemple, deux capitales avaient le même nom, elles seraient représentées chacune par un objet : cette approche est donc réaliste à cet égard. Par contre, cette transformation pourrait paraître redondante, si deux instances d'attribut ayant deux valeurs identiques représentent exactement les mêmes objets du monde réel.

b. Représentation par valeur

Le schéma transformé est :

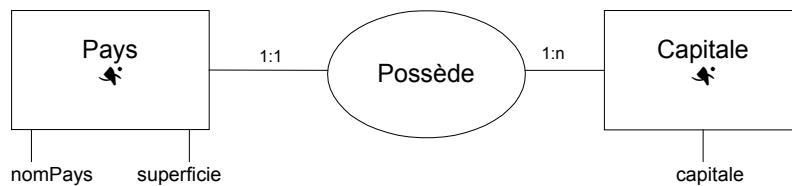


Figure 58

Par rapport à l'exemple choisi, les deux instances de l'attribut « capitale sont « uniques », si bien que la structure des données est identique à celle de la représentation par valeur.

Cependant, pour deux valeurs identiques d'instance de l'attribut « capitale » de « Pays » (en d'autres termes, si deux capitales portaient le même nom), on ne saurait choisir quelle spatialité (plus exactement la spatialité de quelle instance) attribuer au nouvel objet : ceci constitue une limitation.

Une solution, partielle tout de même, serait d'examiner préalablement toutes les instances de l'attribut à transformer, et s'assurer que deux instances ayant des valeurs identiques sont caractérisées par les mêmes propriétés spatiales. On pourrait alors transformer chaque valeur unique en type d'objet spatial.

4.3.2 Attribut simple multivalué

Supposons que nous ayons le schéma suivant :

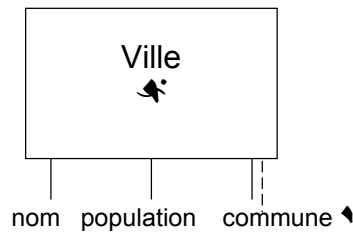


Figure 59

Le type d'objet « Ville » ainsi que l'attribut « commune » sont spatiaux.

Remarque : Les types spatiaux abstraits choisis pour l'un et l'autre dépendent de la nature de l'information que l'on désire stocker (traiter).

Admettons que nous disposions des données suivantes :

	Liège
v1	2000000 { St Germain, Outre-Meuse, Gentil}
	Namur
v2	150000 {Outre-Meuse, Manceau}

Figure 60

Proposons-nous de transformer l'attribut « commune » en type d'objet. Les solutions sont les suivantes :

a. Représentation par instance

Le schéma transformé est le suivant :

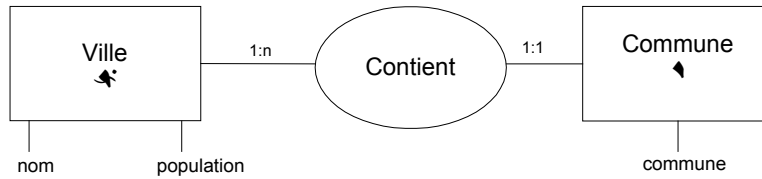


Figure 61

Nous avons appelé l'association « Contient » au lieu de « ComposéDe », car en MADS, ce dernier désigne souvent une association d'agrégation clairement définie. Ce qui n'est pas le cas dans notre exemple. Les éventuelles contraintes spatiales entre l'attribut spatial « commune » et le type d'objet « Ville » sont reportées sur le type d'association « Contient ».

La structure de données obtenue après transformation est la suivante :

v1	Liège	v1,c1	St Germain	c1
	2000000	v1,c2	Outre-Meuse	c2
		v1,c3	Gentil	c3
v2	Namur	v2,c4	Outre-Meuse	c4
	150000	v2,c4	Manceau	c5

Figure 62

Cette transformation conserve la sémantique. La redondance d'instances de « commune » n'est que apparente, puisque dans notre exemple les deux valeurs « Outre-Meuse » correspondent à des objets du monde réel différents, ayant chacun ses propres propriétés spatiales.

b. Représentation par valeur

Le schéma transformé est le suivant :

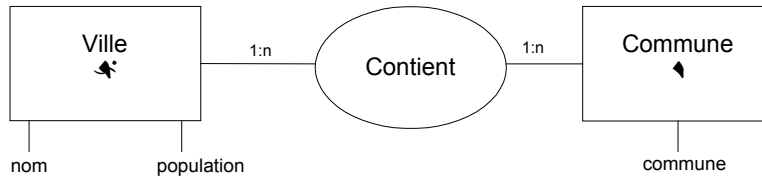


Figure 63

Cette représentation présente les mêmes limitations que dans le cas l'attribut monovalué. (Pour les valeurs identiques d'instances de l'attribut « commune », c.-à-d., dans le cas où deux communes porteraient le même nom)

4.3.3 Attribut complexe

Nous allons à présent étudier la transformation d'attribut spatial complexe. Les remarques étant qualitativement similaires à l'un des cas précédents, nous ne donnerons que les schémas initiaux et transformés.

4.3.3.1 Attribut complexe monovalué

Soit le schéma suivant :

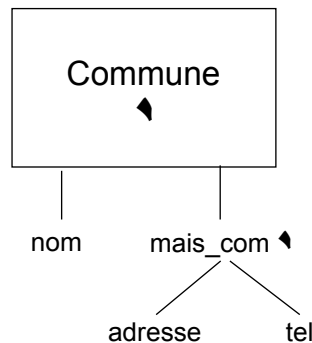


Figure 64

Nous nous proposons de transformer l'attribut « mais_com » (maison communale) en type d'objet. Les schémas transformés sont les suivants :

a. Représentation par instance

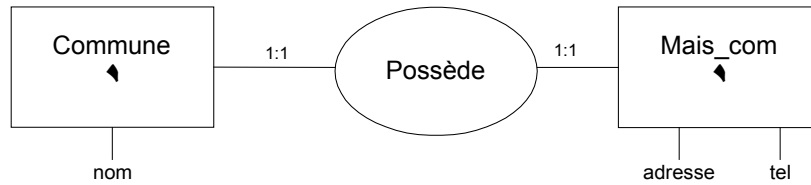


Figure 65

Cette transformation conserve la sémantique

b. Représentation par valeur

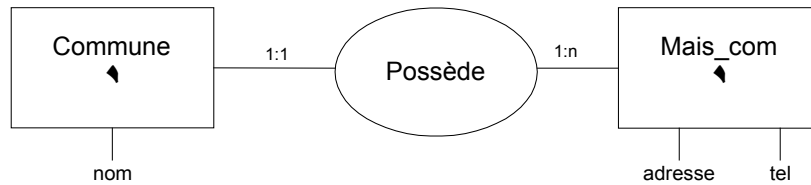


Figure 66

Cette transformation présente les limitations propres à la représentation par valeur. En outre quoique exceptionnel, une maison communale peut gérer plus d'une commune (Par exemple, la maison communale de Bruxelles-ville, gère les communes de Bruxelles-ville et de Laeken). C'est la raison pour laquelle nous avons conservé la cardinalité « 1:n » pour le second rôle.

4.3.3.2 Attribut complexe multivalué

Soit le schéma suivant :

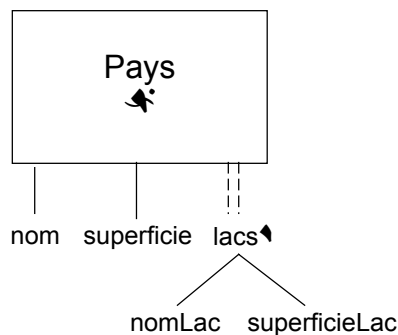


Figure 67

Nous nous proposons de transformer l'attribut « lacs » en type d'objet. Les schémas transformés seront les suivants :

a. Représentation par instance

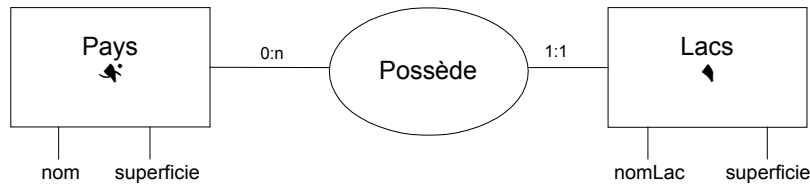


Figure 68

Cette transformation conserve la sémantique.

b. Représentation par valeur

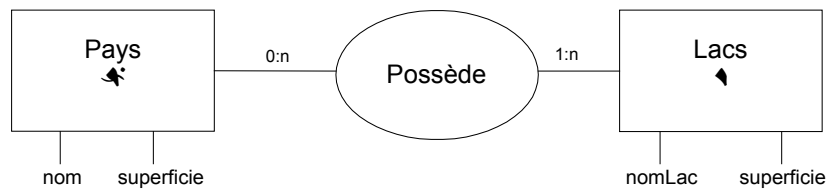


Figure 69

Cette transformation présente les limitations propres à la représentation par valeur.

4.3.4 Attribut variable.

Une dernière catégorie concerne les attributs variables. La difficulté de leur transformation provient du fait que leur valeur est inférée par le système sur base d'une fonction dont le domaine peut s'étendre au-delà de l'attribut. Transformer cet attribut se révèle alors délicat. Il en est ainsi de l'attribut variable « altitude » de l'exemple ci-après :

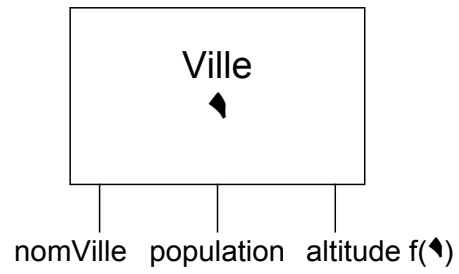


Figure 70

L'altitude est calculée sur tout le domaine spatial du type d'objet « Ville ».La décision d'exécuter cette transformation sera laissée à l'utilisateur.

Chapitre 5 : Implémentation

Chapitre 5 : Implémentation

Le succès d'un modèle conceptuel, indépendamment des ses qualités intrinsèques, dépend bien souvent de la disponibilité d'outils de développement et de production qui l'implémentent.

Des outils CASE sont développés en vue de travailler avec MADS. Nous pouvons en citer deux : **MADS Translator** et **MADS Editor** (tous deux développés dans le cadre du projet MurMur). Mads Editor permet d'éditer des schémas conceptuels suivant le modèle MADS. Les schémas peuvent être ensuite sauvés sous forme binaire ou au format XML. Après l'étude théorique des transformations, la partie pratique du présent travail consistera à ajouter les fonctionnalités de transformation à MADS Editor.

5.1 Présentation de MADS Editor

L'espace de travail de MADS Editor est principalement composé de trois fenêtres :

- La fenêtre « Schema », qui permet de dessiner le schéma,
- La fenêtre « Data Dictionary » qui présente la structure du schéma sous forme d'arbre, permettant ainsi une exploration aisée du contenu du schéma.
- Et la fenêtre « Overview » qui permet d'avoir une vue d'ensemble du schéma. Elle s'avère très utile lorsqu'on a affaire à un schéma de grande taille.

La figure 71 suivante illustre ces trois fenêtres.

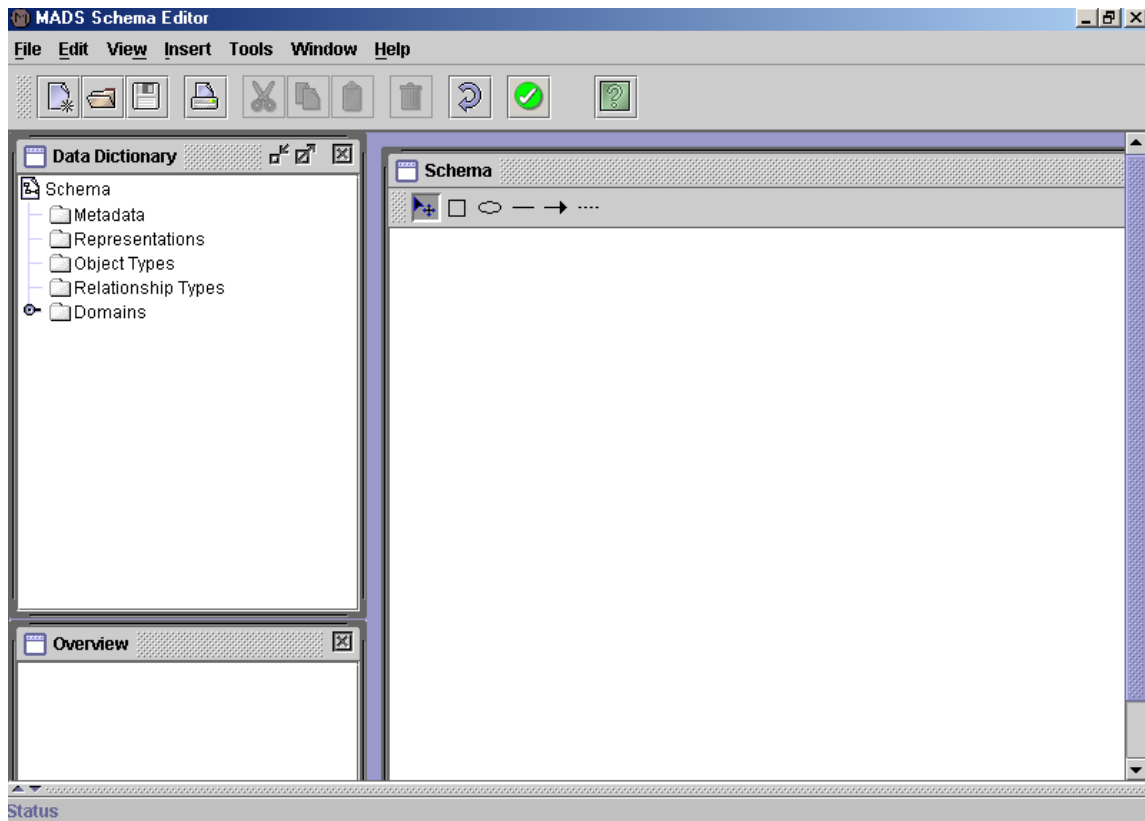


Figure 71

5.2 Implémentation des Transformations

MADS Editor a été développé en langage Java. Notre travail a donc consisté à insérer, créer et adapter le code source afin d'implémenter les nouvelles fonctionnalités. MADS Editor étant un logiciel propriétaire, nous ne pouvons en donner le code source dans ce document. Pour des raisons d'efficacité et d'harmonie, nous avons essayé de réutiliser au maximum le code existant (méthodes, interfaces,...)

Dans les qui suivent, nous exposerons les résultats ainsi que le fonctionnement d'une opération de transformation.

5.3 Réalisation d'une transformation dans MADS Editor

Comme illustration, nous allons à décrire la transformation d'attribut temporel en type

d'objet.

Soit le schéma initial suivant :

Employé	
nom	1:1
prénom	1:1
projets	0:n

Figure 72

Nous proposons de transformer l'attribut « projets » en Représentation par instance.

Etape. 1

Dans le dictionnaire de données, on clique-droit sur l'attribut pour ouvrir le menu contextuel. Sur le menu contextuel, on clique sur « Transform »

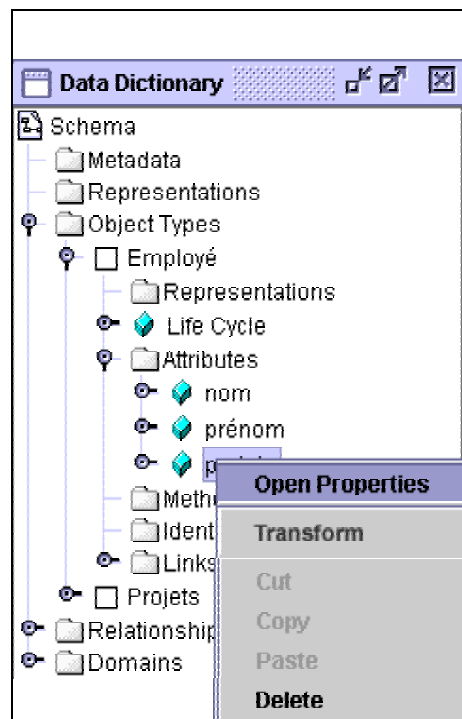


Figure 73

Etape 2

Une boîte de dialogue s'affiche, nous invitant à choisir le type de représentation. Nous choisissons « Instance Representation » et cliquons ensuite sur « continue ».

Le bouton « Cancel » annule l'opération.

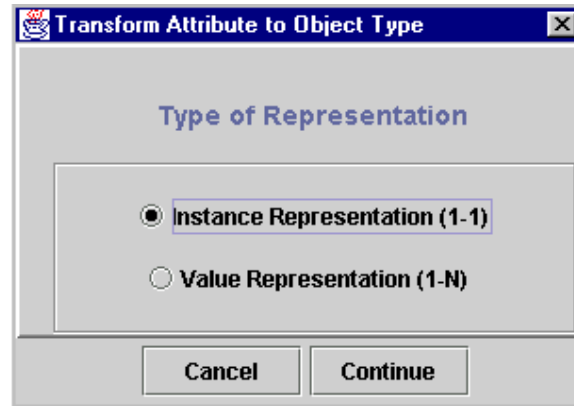


Figure 74

Etape 3

Ensuite, il faut faire le choix sur l'affectation des propriétés spatio-temporelles. Comme vu dans l'étude théorique, les propriétés spatiales ne peuvent être affectées que sur le nouveau type d'objet, aussi la boîte de dialogue est verrouillée sur le type d'objet spatial. Quant aux propriétés temporelles, nous avons le choix de les affecter soit, sur le type d'objet, soit sur le type d'association, soit sur l'attribut.

Nous choisissons d'affecter les propriétés temporelles sur le type d'association.

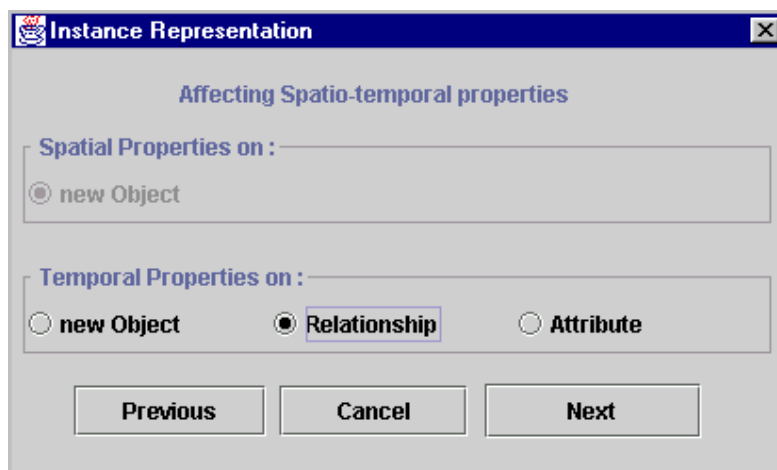
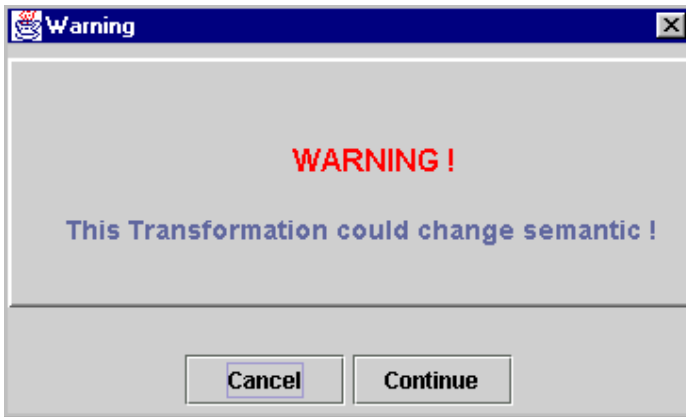


Figure 75



Si la transformation, avec les paramètres choisis, pourrait impliquer un changement de sémantique, un message d'avertissement tel que celui de la Figure 76 est affiché pour en avertir l'utilisateur.

Figure 76

Etape 4.

Nous nommons (« Projets ») le nouveau type d'objet, et cliquons sur « OK »

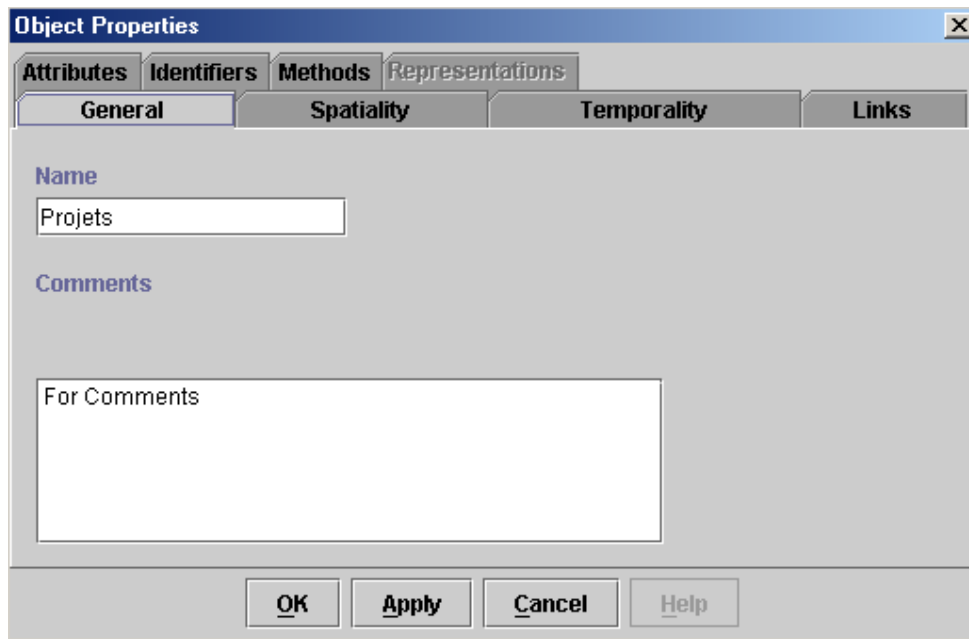


Figure 77

Etape 5.

Nous nommons « TravailleSur » le nouveau type d'association., et cliquons sur « OK »

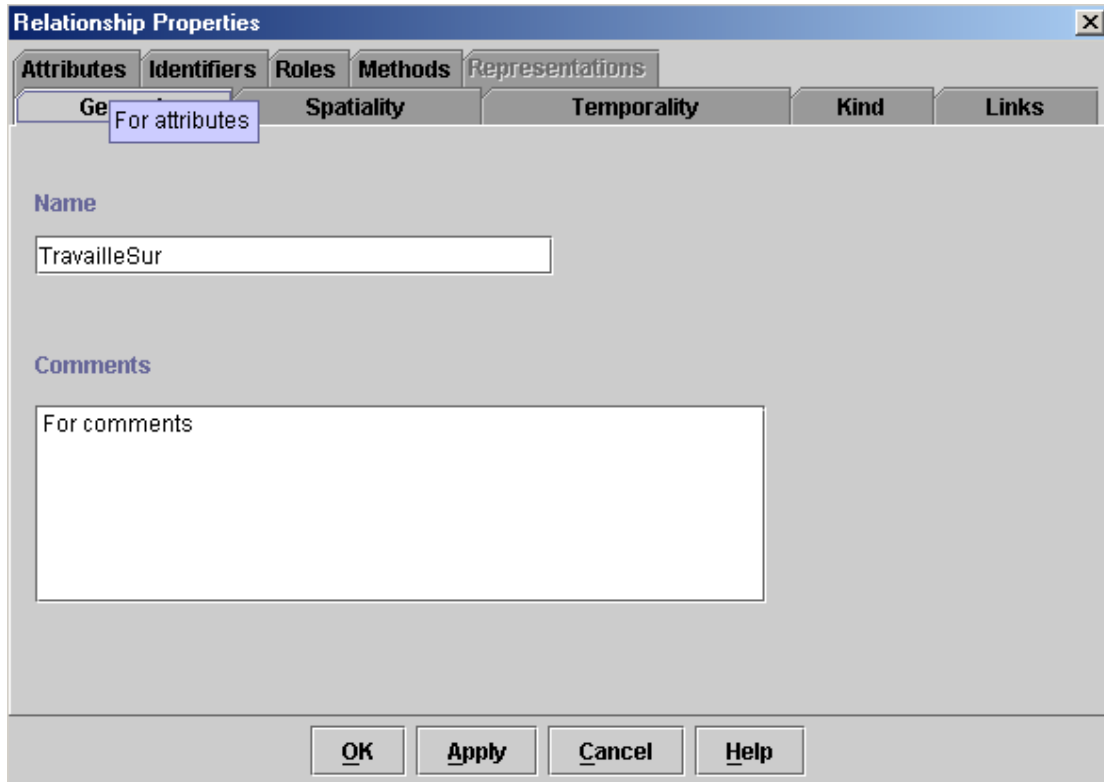


Figure 78

Le schéma transformé remplace le schéma initial (Fin de l'opération !)

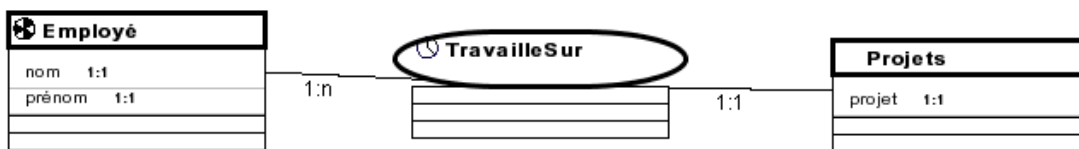


Figure 79

Chapitre 6 : Conclusion

Chapitre 6 : Conclusion

L'étude que nous venons de mener, nous a permis de mettre en évidence la particularité de la transformation de schémas spatio-temporels. Alors que dans le cas classique nous disposons des méthodes clairement définies, le cas spatio-temporel exige plus d'implication (et de responsabilité) de la part de l'utilisateur. Cette différence provient du fait que la spatio-temporalité élargissant considérablement la sémantique, plusieurs interprétations du schéma sont possibles.

L'étude de différentes transformations d'attribut nous a révélé que dans le cas spatial, la spatialité devrait être affectée au nouveau type d'objet et que la représentation par instance conservait toujours la sémantique. Par contre pour les attributs temporels nous avons constaté d'une part que la temporalité sur l'association conserve toujours la sémantique et d'autre part, que la représentation par valeur, lorsqu'elle est réalisable, s'avère très économique par rapport à la transformation par instance. Tout au long de l'étude spatio-temporelle, des pistes ont été évoquées, afin de permettre d'éventuelles études sur d'autres aspects de la transformation spatio-temporelle.

En pratique, il est difficile d'établir une procédure de transformation unique. Il faudra souvent faire à appel à l'utilisateur afin d'avoir la certitude de l'expression de la sémantique. Pour une transformation donnée, nous proposerons la solution la plus économique (évitant ou réduisant les redondances) et qui respecte le mieux la sémantique initiale. Cependant, nous offrons à l'utilisateur, après avertissement, la possibilité de modifier la sémantique initiale.

L'implémentation a nécessité une phase(assez longue) d'exploration de la structure et du code source de MADS. Cet exercice avait un double but : permettre une insertion harmonieuse du code nécessaire aux opérations de Transformation et pouvoir réutiliser au maximum les composants existants.

Bibliographie

OUVRAGES:

Fundamentals of Database Systems

R. Elmasri, S. Navathe
3rd ed. Benjamin/Cummings, 1999.

Introduction to Database Design

(The LIDB tutorial Series - version électronique)

Fifth Edition – March 2002

J-L Hainaut

Université de Namur – Institut d’informatique.

Evolution d’Applications de Bases de Données Relationnelles

(Thèse de Doctorat)

Jean-Marc HICK

Université de Namur

Institut d’Informatique

PUBLICATIONS :

MADS ou l’information spatio-temporelle à portée de ses utilisateurs

Christine Parent*– Stefano Spaccapietra**– Esteban Zimányi***

Pier Donini**– Corinne Plazanet**– Christelle Vangenot**

* HEC-INFORGE, Université de Lausanne, CH 1015 Lausanne, christine.parent@hec.unil.ch

** LBD DI, Ecole Polytechnique Fédérale de Lausanne, CH 1015 Lausanne, prénom.nom@epfl.ch

*** Université Libre de Bruxelles, Informatique CP 165, B 1050 Bruxelles, ezimanyi@ulb.ac.be

MADS, modèle conceptuel spatio-temporel

Christine Parent*– Stefano Spaccapietra**– Esteban Zimányi***

Pier Donini**– Corinne Plazanet**– Christelle Vangenot**

Nadia Rognon***–Pierre-André Crausaz***

*

HEC-INFORGE, Uni Lausanne, CH 1015 Lausanne, christine.parent@hec.unil.ch

**

LBD DI, Ecole Polytechnique Fédérale de Lausanne, CH 1015 Lausanne

*SIRS et HYDRAM, DGR, Ecole Polytechnique Fédérale de Lausanne, CH 1015 Lausanne
{Prénom.Nom}@epfl.ch <http://lbdwww.epfl.ch>*

*** Université Libre de Bruxelles, Informatique CP 165, B 1050 Bruxelles, ezimanyi@ulb.ac.be

Designing Spatio-Temporal Databases

Christine Parent

University of Lausanne , Switzerland

Stefano Spaccapietra

EPFL (Swiss Federal Institute of Technology Lausanne), Switzerland

Esteban Zimanyi, Université Libre de Bruxelles, Belgium

Murmur Project - Workpackage 4

Implementation – Deliverable 11, Schema Editor

Reference : MM-WP4-DLA-011

Version : 3 date :21/03/2001

Murmur Project - Workpackage 2

The Murmur Data Modeling Approach Deliverable 5, Data Model Specification

Reference : MM-WP2-DLA-005

Version : 4 date : 18/09/2000

Specification preservation in schema transformations-application to semantics and statistics

Jean-Luc Hainaut

Data & Knowledge Engineering 19 (1996)99-134

NH-Elsevier

LOGICIELS :

MADS Editor

version 1.3.1

Swiss Federal Institute of Technology

Lausanne

DB-MAIN

Demo version 6.0 g – Jan 8 2002

Copyright FUNDP, 1991-2000

<http://www.info.fundp.ac.be/~dbm/>

SITES WEB DE REFERENCE:

Projet Murmur (site public) : <http://lbdwww.epfl.ch/e/MurMur/>

Projet Murmur (site ULB) : <http://polytech.ulb.ac.be/cs/projects/murmur/>