



HAL
open science

Un Modèle de Vue Spatiale pour une Représentation Flexible de Données Géographiques

Christophe Claramunt

► **To cite this version:**

Christophe Claramunt. Un Modèle de Vue Spatiale pour une Représentation Flexible de Données Géographiques. Informatique [cs]. Université de Bourgogne, 1998. Français. NNT : . tel-01275819

HAL Id: tel-01275819

<https://hal.science/tel-01275819v1>

Submitted on 22 Feb 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THESE DE DOCTORAT
EN INFORMATIQUE
DE L'UNIVERSITE
DE BOURGOGNE

Un Modèle de Vue Spatiale pour
une Représentation Flexible
de Données Géographiques

Présentée par

Christophe Claramunt

soutenue le 27 avril 1998 devant le jury composé de:

Rapporteur	D. Laurent	Professeur	Université de Tours
Co-directeur	M. Mainguenaud	Docteur Habilité	Institut
National des Télécommunications			
Rapporteur	S. Spaccapietra	Professeur	Ecole
Polytechnique Fédérale de Lausanne			
Rapporteur	M. Thériault	Professeur	Université Laval
Directeur	K. Yetongnon	Professeur	Université de Bourgogne

Résumé: *L'objectif de cette étude est la définition d'un modèle de vue spatiale dynamique adapté aux Systèmes d'Information Géographiques. La notion de vue spatiale proposée permet une relative indépendance dans l'interprétation d'un schéma de base de données spatiales à savoir la représentation de données géographiques selon différents points de vues et en fonction d'objectifs distincts de ceux du schéma d'origine. Les mécanismes classiques de vue, dans le domaine des bases de données, n'intégrant pas la composante spatiale, cette recherche propose la définition d'un formalisme de vue adapté aux bases de données spatiales.*

Le modèle proposé est défini comme une extension de la vue classique telle qu'elle est identifiée par les bases de données. Le modèle de vue spatiale intègre la représentation de données spatiales et non spatiales. Une vue spatiale est un ensemble ordonné d'atomes de vue spatiale. Chaque atome de vue spatiale est construit à partir de relations et d'opérateurs spatiaux et non spatiaux. Un ensemble d'opérateurs de manipulation de vues spatiales est défini. Ces opérateurs permettent la manipulation et la dérivation de nouvelles vues spatiales.

La vue spatiale autorise la représentation de modèles spatiaux structurés et cognitifs. La proposition définit un modèle et les opérations qui permettent la décomposition et la représentation d'un processus de déplacement. Le modèle de vue spatiale permet de situer la représentation d'un déplacement dans son contexte géographique à partir de niveaux d'abstraction complémentaires qui intègrent des connaissances partielles. Il associe la description d'un processus de déplacement dans un contexte d'espaces multi-dimensionnels. La continuité de la représentation d'un déplacement est assurée par l'application de constructeurs de graphe appliqués au cadre de la vue spatiale à travers des concepts de collages et de connexions de vues spatiales. L'application d'opérateurs de graphe permet des changements de niveau d'abstraction dans la représentation des processus navigationnels.

La vue spatiale apporte une flexibilité aux utilisateurs dans la représentation de données géographiques. Du point de vue de la modélisation, la vue spatiale permet la représentation de différentes interprétations utilisateurs d'une base de données spatiales. Par extension, la vue spatiale facilite l'évolution du schéma des bases de données spatiales. Elle constitue une forme originale de manipulation et de consultation d'applications géographiques.

Mots-Cles: *Systèmes d'Information Géographiques, Bases de Données Spatiales, Vue Spatiale*

Abstract: *The objective of this study is to provide a dynamic spatial view model to Geographical Information Systems. The notion of spatial view allows independent external interpretations of a database schema i.e. the representation of geographical data according to different point of views and in function of objectives which may be distinct from those of the database schema. As no view definition has been offered for spatial data, this research proposes the definition of a view formalism for spatial databases.*

The model we develop is defined as an extension of the classic database view. Spatial views model both alphanumeric and spatial data. A spatial view is defined an ordered set of spatial view atoms. Each atom is a set of database relations, a set of conventional spatial and non spatial database operators. Manipulations of views allow the derivation of new views according to specific needs. They are realized with a set of identified operators. The spatial view model integrates structured and cognitive spatial representations. The proposal defines a model and operations that allow the decomposition and representation of navigation knowledge. The spatial view allows a displacement action to be situated within its geographical context through complementary abstraction levels that accept partial knowledge. It associates the description of a route with related multidimensional spaces including significant symbolic landmarks and textual descriptions. Continuity of route representation is ensured by the graph concept applied to spatial views and spatial collage spatially materialized by connections. A route trace in each spatial view spaces is described by a section. Graph operators allow abstraction level changes within the route representation.

The spatial view concept provides a relative flexibility to compose a land representation adapted to user needs. From the external modeling point of view, the spatial view concept enables representation of different, numerous database schema interpretations which are inevitable in GIS applications. By extension, it facilitates schema evolution by allowing external representations which do not directly affect the original database schema. For geographic applications, spatial views gives an original solution for database consultation and manipulation tasks.

Key words: *Geographical Information System, Spatial Databases, Spatial View*

Sommaire

1.	Introduction	<i>p. 1</i>
1.1.	<i>Représentation de données géographiques</i>	<i>p. 2</i>
1.1.1.	<i>Modèles structurés</i>	<i>p. 3</i>
1.1.2.	<i>Modèles cognitifs</i>	<i>p. 4</i>
1.2.	<i>Rôle de la cartographie</i>	<i>p. 6</i>
1.3.	<i>Processus de visualisation</i>	<i>p. 9</i>
1.4.	<i>Objectifs de la thèse</i>	<i>p. 12</i>
2.	Modélisation de données spatiales	<i>p. 15</i>
2.1.	<i>Structuration cartographique</i>	<i>p. 16</i>
2.2.	<i>Représentations continues et discrètes</i>	<i>p. 17</i>
2.3.	<i>Géométries et topologie de l'espace</i>	<i>p. 18</i>
2.4.	<i>Organisation et partition de l'espace</i>	<i>p. 20</i>
2.5.	<i>Définition et sémantique des entités</i>	<i>p. 22</i>
2.6.	<i>Définition d'une base de données exemple</i>	<i>p. 24</i>
2.7.	<i>Mécanismes d'abstraction</i>	<i>p. 27</i>
2.8.	<i>Représentation multiple</i>	<i>p. 28</i>
	<i>Un exemple de modèle</i>	<i>p. 31</i>
2.9.	<i>Proposition d'un modèle spatial étendu</i>	<i>p. 34</i>
3.	Traitement de données spatiales	<i>p. 41</i>
3.1.	<i>Identification d'un ensemble d'opérateurs</i>	<i>p. 43</i>
3.1.1.	<i>Opérateurs relationnels</i>	<i>p. 45</i>
3.1.2.	<i>Opérateurs thématiques</i>	<i>p. 46</i>
3.1.3.	<i>Opérateurs réseaux</i>	<i>p. 48</i>
3.1.4.	<i>Opérateurs de manipulation</i>	<i>p. 50</i>
3.1.5.	<i>Sémantique des opérateurs spatiaux</i>	<i>p. 52</i>
3.2.	<i>Définition d'un ensemble de requêtes exemples</i>	<i>p. 53</i>
3.3.	<i>Langages d'interrogation de données spatiales</i>	<i>p. 59</i>
3.3.1.	<i>Langages d'interrogation pour architectures étendues</i>	<i>p. 60</i>
	<i>SAND</i>	<i>p. 61</i>
	<i>GEOQL</i>	<i>p. 63</i>
	<i>ARC/INFO</i>	<i>p. 67</i>
	<i>VISION*</i>	<i>p. 74</i>
	<i>Discussion</i>	<i>p. 78</i>
3.1.2.	<i>Langages d'interrogation pour architectures intégrées</i>	<i>p. 79</i>
	<i>PROBE</i>	<i>p. 80</i>
	<i>GRAL</i>	<i>p. 83</i>
	<i>STARBURST</i>	<i>p. 89</i>
	<i>GéoSabrina</i>	<i>p. 94</i>
	<i>GéO2</i>	<i>p. 96</i>
3.1.3.	<i>Discussion</i>	<i>p. 99</i>
4.		

Modèle externe	p. 102
4.1. Apport des vues	p. 102
4.1.1. <i>Dans les bases de données relationnelles</i>	p. 103
4.1.2. <i>Dans les bases de données orientées objets</i>	p. 105
4.1.3. <i>Propriétés et sémantique des vues</i>	p. 108
4.2. <i>Motivations de l'extension de la vue au spatial</i>	p. 109
4.3. <i>Analyse des propositions actuelles</i>	p. 110
4.3.1. <i>Modèle dual d'interrogation et de présentation</i>	p. 112
4.3.2. <i>TIOGA</i>	p. 114
4.3.3. <i>Modèle de manipulation d'interface</i>	p. 119
4.3.4. <i>EDSS</i>	p. 121
4.3.5. <i>Solutions logicielles dans le domaine des SIG</i>	p. 123
5. Extension du modèle externe au spatial	p. 127
5.1. <i>Premiers principes</i>	p. 127
5.1.1. <i>Modèle et langage d'interrogation</i>	p. 128
5.1.2. <i>Gestion des résultats d'interrogation spatiale</i>	p. 130
5.1.3. <i>Présentation des résultats</i>	p. 131
5.1.4. <i>Première synthèse</i>	p. 132
5.2. <i>Définition de la vue spatiale</i>	p. 134
5.3. <i>Principes</i>	p. 134
5.4. <i>Modèle de vue spatiale</i>	p. 138
5.5. <i>Exemples de vue spatiale</i>	p. 143
5.6. <i>Présentation des résultats</i>	p. 145
5.7. <i>Langage de définition de la vue spatiale</i>	p. 148
5.8. <i>Langage de manipulation de la vue spatiale</i>	p. 152
6. Application de la vue spatiale à un processus de navigation	p. 162
6.1. <i>Modèle du processus de navigation</i>	p. 163
6.2. <i>Manipulation du processus de navigation</i>	p. 168
6.2.1. <i>Langage de définition</i>	p. 168
6.2.2. <i>Langage de manipulation</i>	p. 168
6.2.3. <i>Langage de manipulation - niveau interne</i>	p. 170
6.3. <i>Exemple d'application</i>	p. 176
6.4. <i>Discussion</i>	p. 179
7. Conclusion	p. 180
Références	p. 184

1. INTRODUCTION

Les Systèmes de Gestion de Bases de Données (SGBD) sont devenus des outils indispensables dans notre société où les flux d'informations augmentent exponentiellement avec la mondialisation de l'économie et des échanges. La première étape de ce développement s'est essentiellement réalisée auprès des secteurs d'activités de notre société traditionnellement grands gestionnaires et consommateurs d'information tels que les administrations, l'industrie, les commerces et les services. Le développement rapide d'applications dans ces domaines a amené la réalisation de SGBD technologiquement performants avec une large gamme de services couvrant des fonctions d'intégration, de mise à jour et de traitement de données. La maturité technologique des systèmes informatiques permet aujourd'hui une diffusion des SGBD auprès d'un public de plus en plus nombreux tout comme une multiplication des applications de bases de données. L'évolution générale de l'informatique vers une plus grande convivialité permet en effet de démocratiser l'accès aux techniques et aux fonctions de gestion des bases de données. Parallèlement, les utilisations voient leur champ d'activité concerner des domaines de plus en plus larges comme les sciences environnementales et géographiques, la médecine, la biologie ou les activités socio-économiques. Ces nouvelles possibilités sont facilitées par les progrès considérables et continus des capacités de calcul des systèmes informatiques qui permettent le développement d'applications graphiques laborieuses, performantes et conviviales. De nouvelles applications des bases de données apparaissent donc à la convergence des bases de données et des applications graphiques, c'est le cas du domaine des Systèmes d'Information Géographiques (SIG).

De nombreuses définitions de SIG sont actuellement proposées [Thériault 1996] avec quelques variations dans les sigles utilisés, elles regroupent dans un ensemble et selon les cas l'outil logiciel, l'application, les utilisateurs et l'organisation concernée [Burrough 1986, Aronoff 1989, Coppock 1991, Laurini 1992]. Nous retenons pour la suite de cette étude, et dans un double souci de simplicité et de facilité de lecture, la définition qui donne une orientation logicielle au terme de SIG [Worboys 1995] : *Les SIG sont définies comme des logiciels qui permettent l'intégration, la gestion, l'interrogation, l'analyse et la restitution de données géographiques.*

Les premières applications de SIG se sont développées initialement dans les domaines de la cartographie et de thématiques environnementales à partir des années soixante [Coppock 1991]. Cette première phase de développement des SIG a été accompagnée et permise par les progrès sensibles des domaines de la représentation et du traitement des données spatiales comme la définition de modèles discrets ou continus [Peuquet 1984], de modèles spatiaux formels [Egenhofer 1991a et 1992], de structures physiques de représentation [Samet 1990], d'algorithmes géométriques et de langages d'interrogation [Laurini 1992].

Avec le développement des SIG, les recherches en bases de données spatiales tendent parallèlement à se développer ou à se renforcer pour améliorer les méthodes de représentation et de traitement de l'information spatiale (ex: agendas NCGIA et GISDATA). Nous pouvons citer les travaux portant sur la définition de modèles de données spatiales [Güting 1989, David 1991] et la définition de langages d'interrogations de données spatiales et d'interfaces visuelles [Raper 1991, Medyckyj-Scott 1994, Calcinelli 1994, Egenhofer 1995a].

Dans la mesure où notre recherche aborde plus particulièrement le domaine de la représentation et de la manipulation de données géographiques, cette introduction présente le rôle et les types de modèles utilisés pour la

représentation de données géographiques dans la section 1.1., les méthodes de manipulation cartographiques dans la section 1.2. et les principes de visualisation accompagnant ces manipulations dans la section 1.3. Les objectifs et le plan de cette recherche sont introduits dans la section 1.4.

1.1. Représentation de données géographiques

La nature et les propriétés d'un modèle de données spatiales sont déterminants dans l'appréhension des phénomènes géographiques représentés et étudiés [Haller 1990]. En effet, une analyse de l'organisation spatiale d'un territoire peut entraîner des interprétations différentes même si les concepts spatiaux manipulés par les utilisateurs sont proches [Ady 1990]. Une bonne connaissance des concepts associés aux informations spatiales représentées entraîne donc une meilleure compréhension et assimilation des possibilités des SIG par les utilisateurs [McCormick 1987, Raper 1991]. Dans tous les cas, la nature des modèles conceptuels utilisés influence toute représentation de l'espace. La réussite d'une application de SIG dépend en partie du degré de coïncidence entre la vision utilisateur d'un espace géographique et sa représentation implicite dans le système [Banting 1989]. Notons cependant qu'il est cependant difficile d'évaluer complètement et à priori la sémantique d'une représentation compte tenu de la différence entre le monde réel et les entités artificiellement constituées et du fait que les utilisateurs voient un même contexte différemment [Laurini 1992]. De plus la traduction d'une représentation conceptuelle en données de la base se fait finalement plus ou moins bien selon la qualité du SGBD utilisé par le SIG.

Les modèles spatiaux constituent donc le support méthodologique et structurel de représentation de données spatiales. Ils constituent l'outil (le langage) qui permet de décrire une application à partir de structures mentales utilisées pour organiser la perception d'un espace. Ces structures mentales définissent un ensemble de concepts spatiaux utilisés par le modèle (le vocabulaire du langage). Un modèle de données spatiales permet de décrire les composantes d'un schéma de données spatiales. Deux types de modèles peuvent être distingués : les modèles de haut niveau, qui sont proches de la façon dont un utilisateur perçoit les données, et les modèles de bas niveau qui détaillent la façon dont les données sont logiquement représentées [Elmasri 1989]. Nous montrerons que les modèles spatiaux actuellement utilisés pour la représentation de données géographiques privilégient les structures de représentations statiques au détriment de représentations plus dynamiques. Dans ce sens nous catégorisons les modèles spatiaux en deux classes essentielles : les **modèles structurés** plus proches d'une représentation cartésienne et informatique de la donnée spatiale)¹ et les **modèles cognitifs** plus proches d'une vision mentale et de structure non complètement définie du point de vue de l'utilisateur.

1.1.1. Modèles structurés

La plupart des modèles spatiaux des SIG permettent actuellement de décrire l'espace sous une forme structurée continue et cartographique [Peuquet 1984, Burrough 1986, Laurini 1992]. Ils traduisent une organisation spatiale

¹ Cette dénomination a essentiellement pour objectif de distinguer deux catégories de modèles, la terminologie utilisée est proposée à titre indicative.

statique par la définition d'un ensemble d'entités spatiales et de relations. La structure de référence, support de représentation d'un SIG est la carte numérique (par extension de la carte traditionnelle).

Les modèles structurés s'attachent à décrire l'organisation spatiale et fonctionnelle d'un territoire. La représentation d'une organisation spatiale décrit des portions de territoire qui possèdent une homogénéité naturelle (ex: des unités de sol, des zones de cultures, des unités de végétation) ou des modèles administratifs composés d'informations de nature anthropique (ex: un découpage en communes et quartiers, une organisation d'un pays en régions militaires) [Laurini 1992]. Les représentations fonctionnelles s'attachent à décrire et à qualifier les rôles des composantes d'un espace, c'est à dire associer des entités interdépendantes au sens d'une ou de fonctions exercées au sein d'un territoire (ex: un réseau de transport, des zones d'influence [Brunet 1990]). Les représentations fonctionnelles exploitent des structures de graphe intégrant des lieux caractéristiques et leurs interrelations logiques, et des paramètres comme des flux ou des temps de parcours. Les représentations cartographiques basées sur les notions d'organisation spatiale et de fonctions possèdent des propriétés spécifiques : un ingénieur récemment formé aura une vision très géométrique d'une centrale nucléaire, un contrleur expérimenté aura une vision plus fonctionnelle où la taille des éléments sera proportionnelle à leur importance sur le plan de la sécurité [Barthet 1993]. Les applications de SIG devraient idéalement être élaborées à partir d'une combinaison de données traduisant l'organisation et les fonctions d'une application dans la mesure où les objectifs de gestion et de planification du territoire imposent d'intégrer à la fois les structures spatiales du territoire concerné comme leur fonction au sein de ce même territoire.

1.1.2. Modèles cognitifs

Si les modèles structurés sont la référence courante de description de données géographiques, d'autres formes de perception de l'espace, plus cognitives, sont également utilisées. Les modèles cognitifs font appel à des formes mentales de description d'un espace, ils sont dépendants des utilisateurs et donc notamment influencés par les modèles culturels utilisés, ils sont donc nombreux et variés. De par leur grand degré de liberté et de subjectivité, les modèles cognitifs donnent lieu à de multiples formes graphiques combinant des formes d'expression informationnelles ou même artistiques. Dans un contexte d'expressions de formes descriptives et informationnelles d'un espace, l'objectif d'un modèle cognitif est de traduire la perception dynamique d'un territoire. Ces perceptions peuvent relever d'une forme d'expérimentation de l'espace qui se traduit par l'expression d'espaces perçus (l'espace environnant) ou d'espaces transperçus (les espaces à petite échelle que l'on ne perçoit pas directement) [Downs 1977, Kuipers 1978]. L'expression d'un modèle basé sur l'expérience peut être réalisée par une approche multimodale qui permet la présentation de l'information selon différentes formes comme le langage naturel, le gestuel et le graphique [Maaß 1993].

L'expression spatiale d'un modèle cognitif correspond à la traduction cartographique d'une représentation mentale, qualifiée de carte cognitive [Kuipers 1978]. Le pouvoir d'expression d'une carte cognitive est plus proche de l'esprit humain que celui d'une cartographie classique. Les cartes cognitives, largement étudiées tant en psychologie qu'en linguistique, héritent des caractéristiques de diversité et de richesse d'une représentation mentale mais également de leurs imperfections [Mark 1991]. L'espace représenté par une carte cognitive possède

quatre dimensions (trois pour l'espace et une pour le temps). La traduction spatiale d'une carte cognitive est discontinue (connaissance partielle) et dynamique (cohabitation de différentes perceptions de l'espace et notion de mouvement), et en conséquence moins facilement identifiable et caractérisable par les modèles spatiaux courants. La topologie perçue est relativement bien conservée et essentiellement linéaire, les notions de distance sont par contre approximatives.

Ces modèles recouvrent en particulier les **processus de navigation** utilisés pour exprimer la perception d'un déplacement. Les modèles cognitifs décrivant des processus de navigation sont basés sur l'expérience. Ils sont définis par l'association d'un modèle de déplacement et de son espace environnant. L'environnement d'un processus de navigation est composé des perceptions visuelles qui accompagnent ce déplacement et qui agissent comme des points de repères. Ces modèles s'expriment à partir d'une base d'expérience sensimotrice : quand nous identifions le souvenir mental d'un déplacement à travers un espace, le modèle cognitif traduit une séquence d'images correspondant à des points particuliers du parcours (une image est ici équivalente à la mémoire d'un événement visuel correspondant à un point temporel d'un parcours) [Kuipers 1978]. Ces perceptions cognitives sont influencées par les modèles mentaux utilisés (i.e. souvent structurés sous la forme de langages). Les propriétés des cartes cognitives traduisent la relativité du pouvoir d'expression de la mémoire. La perception de ces représentations cognitives provient en effet d'un enregistrement mental de nos activités sensorielles. Cet enregistrement permet la constitution de la mémoire cognitive d'une représentation du territoire dont la qualité sera variable selon les utilisateurs et diminuée par le facteur temps.

Dans l'expression d'un modèle cognitif, le concept de **collage cognitif** décrit l'association sémantique de deux niveaux d'abstraction d'un espace représenté (e.g. collage cognitif de la représentation d'un parcours à échelle locale avec une représentation décrivant le prolongement de ce parcours à échelle régionale). Les représentations d'un même espace à plusieurs niveaux d'abstraction constituent un ensemble d'**espaces multidimensionnels**. Le concept de collage permet une représentation plus fonctionnelle de l'espace dans la mesure où chaque niveau d'abstraction correspond à un rôle donné par l'objectif de l'application [Tverski 1993]. Ces espaces multidimensionnels peuvent être décrits par des modes de représentations variables (ex: cartes numériques, images, symboles, photographies). Si ces modèles cognitifs apparaissent comme des formes pertinentes de description de données géographiques, ils restent encore essentiellement définis par les sciences linguistiques et cognitives, leur intégration en tant que composante des modèles spatiaux de bases de données reste encore à aborder et à modéliser.

1.2. Rôle de la cartographie

Les applications classiques de bases de données manipulent des données à travers des langages d'interrogation et des formes relativement simples de présentation de résultats (e.g. tables, nouvelles relations). Dans le contexte des SIG, la cartographie numérique donne un contexte opératoire et des méthodes de présentation de données géographiques beaucoup plus riches sur le plan de l'expression sémantique.

La conception de cartes est devenue une discipline scientifique moderne après la seconde guerre mondiale quand la cartographie est devenue une ressource stratégique incontournable [McMaster 1987, Buttenfield 1991].

La cartographie numérique bénéficie des siècles de connaissances apportées par les techniques et le savoir-faire de la cartographie traditionnelle. Elle allie la dynamique et la souplesse de la cartographie aux potentialités de traitement et de visualisation offertes par les ressources informatiques. La carte peut avoir plusieurs rôles : un objectif de présentation et de communication du point de vue de l'utilisateur, une structure de représentation et de stockage de données spatiales du point de vue de la technique informatique. Nous abordons dans cette étude et dans cette introduction la composante utilisateur de la carte.

Les géographes ont traditionnellement développé des règles de construction cartographique de l'information géographique qui restent encore difficiles à formaliser; ces règles font appel à l'expertise du cartographe qui applique toute une série de manipulations variables selon les objectifs de présentation et son degré d'expertise [Imhoff 1982, Bertin 1983, Egenhofer 1990b, Frank 1982]. L'absence de règles cartographiques formelles n'est pas la conséquence d'un manque de connaissance, ou d'un manque d'intérêt des utilisateurs de cartes; c'est simplement le reflet de la difficulté de déduire un ensemble de règles précises et universelles pour la représentation de processus heuristiques complexes. Le facteur de complexité dans la description et la compréhension de l'information contenue dans un document cartographique est a fortiori dépendante du degré d'expertise utilisé lors du processus de création de ce document [Keates 1982]. Cette dernière difficulté complique toute démarche complète d'intégration des processus de composition cartographiques au sein d'un système de gestion et de traitement de bases de données spatiales.

Le support cartographique est l'un des plus communément utilisé pour représenter le monde à distance [Forer 1990] même si la tradition orale, notamment, joue encore un rôle important dans certaines civilisations dans la description d'environnements géographiques [Curnow 1990]. L'un des atouts importants des cartes traditionnelles et numériques est la faculté de pouvoir représenter des informations spatiales définies à des petites échelles avec des modèles de grandes échelles [Mark 1986]. Construire une carte c'est un processus de déconstitution et de reconstitution de la réalité [Kirkpatrick 1989]. Selon le modèle réductionniste, le cartographe déconstitue la réalité en cartes, et l'utilisateur reconstitue un modèle à partir de cette même carte. Le processus de création de carte est relativement artificiel et sélectif, les nuances sont nombreuses. Le processus de reconstitution peut être amélioré si l'on utilise les informations potentiellement décrites par une image. La lecture d'une carte est définie comme la compréhension d'une organisation spatiale et l'identification des éléments de la carte par l'utilisateur selon sa connaissance et son expérience [Scott 1987]. Une carte traduit une relation, entre le cartographe et le lecteur, autour de la perception de l'environnement étudié [Wood 1968 et 1972] (Figure 1.1.).

Le problème de reconstitution d'une carte numérique, et notamment celui des temps de calcul pour de larges volumes de données, est l'une des problématiques informatiques rencontrées par les organismes producteurs et gestionnaires de cartes numériques. La réutilisation de cartes numériques existantes est un besoin fondamental qui impose de gérer les mises à jour de ces documents [Kirkpatrick 1989, Abel 1992]. Si les premiers logiciels de SIG étaient essentiellement des outils de cartographie numérique [Coppock 1991], le rôle de système d'information s'impose donc progressivement avec le renforcement des besoins de gestion et de traitement de ces données numériques.

La carte a forcément un aspect réducteur, le cartographe crée un modèle de la réalité en filtrant certains éléments du monde réel. Une carte est constituée de classes d'objets cartographiques que l'on peut regrouper selon des règles cartographiques et sémantiques. Ces classes d'objets sont élaborées à partir de primitives cartographiques. Un langage cartographique est défini comme un ensemble de symboles et de signes (l'alphabet) lié à un ensemble de règles (la grammaire) qui permettent à un cartographe de communiquer sa conception de la réalité à travers un média cartographique : la carte [Ramirez 1993]. Une représentation graphique est la transcription dans un système graphique de signes d'une information à travers un système de règles intermédiaires [Bertin 1983]. Les techniques de composition graphique s'appuient sur un nombre fini de variables visuelles identifiées par les cartographes [Imhoff 1982, Bertin 1983]. Nous montrerons à travers notre recherche que ces paramètres stables peuvent être modifiés et intégrés dans un modèle de manipulation de données spatiales.

Figure 1.1. - Perception et communication cartographique

1.3. Processus de visualisation

L'étape de visualisation constitue l'un des derniers éléments d'un processus de communication entre une base de données spatiales et un utilisateur final. Une visualisation est un processus de représentation synoptique de l'information dont l'objectif est d'identifier, communiquer et interpréter des formes et des structures (spatiales dans notre contexte applicatif) [Buttenfield 1991]. La réalisation d'une visualisation couvre des aspects informatiques, cognitifs et mécaniques dans la génération, l'organisation, la manipulation et la compréhension des données visualisées. Un processus de visualisation traduit un processus cognitif (Figure 1.2.) ; c'est notre capacité à développer des représentations mentales qui nous permet d'identifier des phénomènes et de créer ou d'imposer un ordre [McEachren 1992]. Ces visualisations peuvent être symboliques, graphiques ou iconiques et sont le plus souvent différentes des formes classiques d'expression (textuelles, verbales et formulatoires) dans leurs caractéristiques, leur format et leur qualité.

Une visualisation numérique permet d'examiner et d'assimiler un plus grand nombre de données avec un gain de temps, permettant ainsi une grande interaction avec l'utilisateur [Ganter 1988]. Ce progrès améliore la perspicacité, l'innovation et plus généralement l'étude et la compréhension d'une information ou d'un phénomène. Un lecteur assimile une représentation par des processus mentaux qui font appel à des mécanismes d'analogie, d'induction et d'imagination. La visualisation d'un ensemble d'informations est un processus basé sur des étapes successives: accéder à des informations pertinentes, communiquer un (des) message(s) par une présentation sous une forme plus ou moins élaborée des données et des traitements de ces données.

Figure 1.2. - Mécanismes associés aux processus de visualisation

Les traditions culturelles, linguistiques, sociales et géographiques peuvent entraîner des structurations différentes de l'espace (ex: la notion de distance varie selon la nature montagneuse ou plane d'un territoire) [Mark 1988]. On peut distinguer les cartes à voir des cartes à lire [Bertin 1983]. La qualification de l'utilisateur sera également un facteur déterminant dans le choix d'une méthode de construction d'une d'interface (ex: novice/expert, degré de familiarité, fréquence d'utilisation). Les utilisateurs d'une base de données peuvent être classifiés en fonction des objectifs de l'application, de leur degré d'expérience avec la tâche à accomplir, et de leur degré de connaissance technique et de la fréquence de renouvellement de cette opération [Card 1983]. Un système de visualisation efficace doit permettre une interaction forte permettant l'apprentissage [Staes 1990]. La réalisation d'une visualisation est une fonction du domaine d'application, de l'utilisateur, du contexte et de l'historique de l'utilisation. Une visualisation doit permettre de comprendre plus ou moins intuitivement la nature des processus représentés et leurs interdépendances [Fedra 1991].

Les processus de visualisation correspondent partiellement à la recherche d'un ordre élégant de représentation. Il est important de préciser les types d'attributs de données (ordinal, cardinal, nominal) tout comme la qualité des données visualisées (ex: position, consistance) [Radermacher 1991, Vidale 1993], voire la propagation des erreurs dans les traitements [Lagacherie 1992]. La somme d'informations contenue dans un message est notamment inversement proportionnelle aux incertitudes résultantes [Shannon 1949]. Les besoins en visualisation des SIG demandent des solutions conceptuelles, technologiques et d'évaluation particulières que l'on peut classifier en trois catégories fonctionnelles: interrogation, analyse et aide à la décision [Buttenfield 1990].

Dans le cas de processus d'interrogation de données spatiales, les qualités de réalisme et de précision requises varient en fonction des domaines et besoins applicatifs. Pour des applications légales de gestion du territoire, la précision et le réalisme sont des critères importants dans la réalisation d'une visualisation cartographique ; pour des opérations de consultation interactives, l'importance des graphismes et des modes de présentation utilisés sont secondaires.

Dans le cas de l'analyse, les techniques de visualisation utilisées couvrent la représentation de processus spatiaux et temporels à travers une compréhension des structures et interrelations des données analysées. Dans ce contexte, les hypothèses de génération des données spatiales doivent être notamment intégrées dans des mécanismes visuels. L'objectif de la visualisation pour de l'analyse est de contrôler et de synthétiser un processus d'analyse des données, les progrès du traitement et finalement les résultats.

Les processus de visualisation jouent également des rôles déterminants dans les processus décisionnels [Buttenfield 1990]. L'objectif principal d'une visualisation est de résumer et d'interpréter un ensemble d'information et de communiquer un certain nombre de solutions ou de décisions potentielles en matière d'aménagement du territoire ou de transport par exemple. Notons que des réflexions restent à mener autour de la manière dont la visualisation de données spatiales affecte la façon dont les utilisateurs vont prendre une décision, et sur la façon dont les processus visuels peuvent s'incorporer dans les étapes de processus de prise de décision.

Ces composantes visuelles apportent un niveau intermédiaire conceptuellement élaboré entre un SIG et un utilisateur final. Nous analyserons à travers cette recherche l'impact et le rôle des processus de visualisation dans la définition d'un outil de manipulation de données spatiales comme les insuffisances actuelles des solutions

proposés. Nous montrerons notamment que leur prise en compte dans la définition d'un outil de traitement flexible améliore les potentialités d'une solution orientée utilisateur. Dans notre contexte de recherche, il s'agira de répondre à des objectifs de présentation et d'interrogation de données spatiales soit le premier niveau des objectifs de visualisation présentés dans cette section.

1.4. Objectifs de la thèse

La définition d'un outil flexible de manipulation de données spatiales impose la caractérisation d'un modèle spatial de référence qui intègre à la fois les composantes structurées et cognitives. Nous montrerons à travers notre étude que les outils de représentation et de manipulation actuellement proposés au sein des SIG ne remplissent pas cette condition. Par extension nous motiverons et illustrerons l'intérêt de notre approche à travers des exemples applicatifs portant sur des contextes qui couvrent les modèles spatiaux structurés et cognitifs.

Nous développerons cette recherche à partir de la construction d'un modèle spatial de référence élaboré à partir d'une analyse des caractéristiques cartographiques des données spatiales continues et discrètes qui constituent les deux formes principales de description de l'information géographique. Les propriétés topologiques principales de ces structures spatiales continues et discrètes sont décrites. Les mécanismes d'abstraction des données spatiales et le phénomène de représentation multiple, au sens spatial du terme, sont analysés en matière d'influence et de contrainte sur le modèle spatial. Cette analyse des modèles spatiaux est complétée par une description de la sémantique des données spatiales et non spatiales représentées, d'un point de vue spatial et alphanumérique. Ces propriétés sont illustrées par une base de données spatiales exemple qui est utilisée le long de ce document comme une référence applicative.

Le modèle spatial de référence constitue la partie statique de la démarche de construction de la vue spatiale. Les opérateurs, à travers le langage d'interrogation, apportent la constituante dynamique. Nous identifions un ensemble d'opérateurs non spatiaux et spatiaux à partir respectivement des opérateurs relationnels pour le domaine alphanumérique, thématiques et réseaux pour le domaine spatial. La sémantique spatiale et non spatiale de cet ensemble d'opérateurs sont définies. Cet ensemble d'opérateurs de base est confronté aux solutions actuellement définies dans le domaine des langages d'interrogation de données spatiales. Cette étude analyse un ensemble représentatif de langages d'interrogation de données spatiales, elle montrera notamment qu'aucune des solutions actuellement proposées par la recherche ou par l'industrie des SIG ne sont satisfaisantes sur le plan de la complétude et des pouvoirs d'expressions disponibles au regard des ensembles d'opérateurs analysés (interrogation d'opérateurs thématiques et réseaux notamment).

La dernière composante du modèle spatial est le niveau de schéma externe tel qu'il est défini par les SGBD. Une analyse des apports actuels des modèles externes (i.e. les vues) est réalisée, elle intègre les différentes propositions des solutions relationnelles et orientées-objets. Les propriétés et avantages de ces mécanismes de vue sont mis en relief dans le contexte des applications spatiales. Nous identifions parallèlement un cadre d'extensions nécessaires dans le contexte du spatial. Si aucune proposition précise ou formelle de vue spatiale n'est, à notre connaissance, proposée dans le domaine des bases de données spatiales, nous réalisons une analyse des solutions actuelles, essentiellement dans le cadre d'interfaces de bases de données spatiales, qui se rapprochent de cet

objectif. Un état de l'art de ces propositions est développé. Il fait notamment apparaître que ces solutions ne répondent pas individuellement à l'ensemble des propriétés que devrait posséder un concept de vue spatiale.

La vue spatiale est construite à partir d'un langage de définition de référence qui est présenté, elle est utilisable et exploitable par un langage de manipulation classique, il n'est pas ici nécessaire de redéfinir un nouveau niveau de langage. Un ensemble d'opérateurs de manipulation spécifique est cependant défini à partir du niveau et des propriétés nouvelles apportées par les notions d'ordre sur les atomes de vue spatiale et d'échelle de manipulation de la vue spatiale. Des opérateurs d'intersection, d'intégration, de fusion et de différence sur les vues spatiales sont proposés et spécifiés.

Le modèle de vue spatiale est illustré dans un contexte de représentation cartographique classique et dans celui d'une représentation navigationnelle proche d'un modèle spatial cognitif. L'application de la vue spatiale dans un contexte de représentation d'un processus de navigation amène notamment la définition de nouvelles propriétés. Des concepts de collages de vues spatiales, représentant des espaces multi-dimensionnels et de connexions entre entités de ces espaces respectifs, sont identifiés et modélisés. L'application d'opérateurs de graphe permet des changements de niveau d'abstraction dans la représentation de processus de navigation. L'ensemble fournit un outil de manipulation de données spatiales qui intègre à la fois les composantes statiques (i.e. modèles structurés) et dynamiques (i.e. modèles cognitifs et processus de navigation) des données spatiales. L'outil de manipulation proposé constitue un cadre flexible de manipulation externe qui modélise les interrogations et présentations de données spatiales. L'outil propose également de nouvelles perspectives dans l'utilisation de bases de données spatiales. Les applications potentielles concernent les domaines qui modélisent à la fois des données cartographiques et des processus spatiaux de navigation.

La suite de cette thèse est organisée de la façon suivante. La section 2 aborde les principes de modélisation de données spatiales et la section 3 décrit les langages de traitement de données spatiales. La section 4 présente les principes et les propositions actuelles des modèles externes. La section 5 développe notre proposition d'extension du modèle externe au contexte spatial et la section 6 présente une application de cette dernière extension à un processus de navigation. La section 7 conclut le document.

2. MODELISATION DE DONNEES SPATIALES

Ce chapitre décrit les caractéristiques des données géographiques qui vont servir de base à la définition d'un modèle spatial de référence. Les données spatiales possèdent en effet des propriétés spécifiques et originales qui les distinguent des informations généralement utilisées par les bases de données traditionnelles. Plusieurs auteurs ont successivement identifié les caractéristiques de l'information spatiale [Peuquet 1984, Marble 1984, Burrough 1986, Aronoff 1989, Samet 1990, Tomlin 1990 et Laurini 1992]. Les données alphanumériques généralement traitées dans les applications traditionnelles de bases de données sont le plus souvent définies par des types atomiques. La représentation de données spatiales pose des problèmes plus complexes de définition des entités dans un espace géographique aux nombreuses propriétés et contraintes de structuration.

La définition des entités d'une application géographique demande notamment la gestion préalable des structures cartographiques de représentation de l'espace (ex: la saisie de données spatiales impose de suivre une procédure préalable de structuration cartographique, de choix d'un système de projection, et d'identification d'un système de coordonnées). Les types représentant les données spatiales sont variés et de nature complexe (ex. hiérarchies de points, lignes, et polygones ; images satellitaires) et sont généralement organisés à travers des modèles discrets ou continus de l'espace porteurs de nombreuses propriétés mais également de contraintes géométriques. Les dépendances sémantiques entre les données spatiales sont importantes, elles couvrent notamment les propriétés topologiques de l'espace (ex: réseaux logiques, notions de voisinage) et les mécanismes d'abstraction utilisés dans les applications géographiques (ex: généralisation et agrégation cartographique). La définition d'une entité, qui varie en fonction des utilisateurs dans les applications classiques, est de plus dans un contexte géographique également dépendante de l'échelle de représentation.

Ces exemples illustrent quelques particularités importantes des données spatiales, les sections suivantes présentent les structures spatiales et les principes méthodologiques utilisés dans les démarches de modélisation de données spatiales.

Sur le plan des structures spatiales utilisées dans une démarche de modélisation de données spatiales, la section 2.1. présente les principes de représentation cartographique indispensables pour géoréférencer des données spatiales ; la section 2.2. décrit les structures de représentation, la section 2.3. les éléments de géométrie utilisés dans la modélisation de données spatiales et la section 2.4. les méthodes de partition de l'espace géographique. Concernant les principes de modélisation utilisés dans les applications spatiales, la section 2.5. décrit les principes de définition des entités, la section 2.6. définit une base de données exemple, la section 2.7. présente les mécanismes d'abstraction et la section 2.8. les propriétés de la représentation multiple. A partir de cette analyse, la section 2.9 propose un modèle spatial de synthèse qui prend en compte ces caractéristiques spatiales et l'illustre à travers la base de données exemple.

2.1. Structuration cartographique

Les méthodes de structuration cartographique utilisées dans une démarche de modélisation de données spatiales sont un élément important d'appréciation de la sémantique spatiale des données représentées. Les propriétés des projections cartographiques indiquent notamment la transformation géométrique effectuée entre les données

géographiques terrestres et les données cartographiques planaires r_sultantes. Plus pr_cis_ ment une projection cartographique permet de passer de coordonn_es terrestres globales (i.e. longitude et latitude) à des coordonnées locales et projetées. Les projections cartographiques sont caractérisées par les propriétés suivantes :

- La projection est conforme quand elle préserve les directions locales et ne déforme pas les angles;
- La projection est équivalente lorsque le rapport entre les superficies des zones est préservé;
- La projection est équidistante lorsqu'elle conserve le rapport d'échelle linéaire le long de certaines lignes choisies.

Le choix d'un type de projection est une fonction de la situation et de la taille du territoire _tudi_ mais _galement des donn_es disponibles et des objectifs de l_ application. Pour la plupart des applications, il est bien souvent pratique et efficace de se conformer au système en vigueur dans la zone d'étude. Le système cartésien à coordonnées absolues est le plus utilisé dans les applications géographiques même si quelques domaines spécialisés utilisent des coordonnées polaires (ex: navigation aérienne), des coordonnées cartésiennes relatives (ex: navigation navale), ou des techniques de géocodage (ex: gestion d'adresses).

2.2. Représentations continues et discrètes

Une structuration cartographique permet de disposer de données spatiales planaires qu'il est nécessaire d'organiser selon un modèle géométrique, Deux catégories de structures géométriques sont utilisées pour exprimer les données spatiales bi-dimensionnelles dans un plan: les repr_sentations **discrètes** (i.e. vectoriel) et les représentations **continues** (i.e. image ou raster) [Peuquet 1984].

Les modèles discrets de l_espace utilis_s dans les SIG sont essentiellement _ deux dimensions (comme la cartographie traditionnelle) et bas_s sur des **primitives cartographiques** de type point, noeud, ligne, polyligne, polygone et géocomposé regroupées à travers des hiérarchies de composition avec quelques variantes dans les primitives utilisées selon les modèles géographiques employés. Une ligne sera par exemple constituée par une liste ordonn_e d_un nombre variable de points entre deux noeuds, elle sera en m_me temps l_un des éléments de définition d'une polyligne qui pourra à son tour servir de contour d'un ou plusieurs polygones².

Les modèles continus de l'espace sont le plus souvent décrits par des grilles matricielles régulières. La primitive de base est le pixel (des structures spatiales comme le quadtree sont des optimisations d'une représentation par pixel). La valeur d'une variable est d_finie sur un maillage r_gulier de l_espace (un seul attribut, g_n_ ralement num_rique). Les repr_sentations irr_guli_res sont r_servées à des applications spécialisées comme les modèles numériques de terrain.

Les principes de définition et les propriétés respectives des représentations discrètes et continues dans le cadre des applications de SIG ont largement été définis et discutés [Peuquet 1984, Abel 1992]. L'intérêt d'une utilisation

² D'autres applications spatiales utilisent des spécifications de primitives graphiques particulières [Hildebrandt 1993] : les applications de CAO tri-dimensionnelles utilisent par exemple des _l_ments comme l_enveloppe graphique, la face et des op_rateurs d_intersection g_om_rique pour exprimer un objet solide.

des deux méthodes de représentation est aujourd'hui admis pour couvrir l'ensemble des fonctions de gestion et d'analyse des applications de SIG. Du point de vue de l'utilisation, la combinaison de données continues et discrètes enrichit sensiblement les applications (ex: fonds de plan raster de situation, utilisation de photos aériennes numérisées pour la mise à jour de planches cadastrales).

Si le besoin d'intégration des modèles continus et discrets de l'espace est bien identifié, il entraîne un challenge important pour la recherche ; la définition d'un modèle spatial intégré supportant ces deux structures de perception de l'espace n'a en effet pas rencontré de solution méthodologique opérationnelle à ce jour. Les modèles spatiaux restent donc hétérogènes et difficilement intégrables dans la mise en œuvre d'applications de SIG. Les opérations de traitement de données spatiales traduisent donc cette spécialisation des modèles continus et discrets ; ces outils ne sont, de plus, pas transparents pour l'utilisateur compte-tenu du niveau d'abstraction offert par les opérateurs de gestion de données spatiales (i.e. une vision cartographique de l'espace). Un modèle spatial de référence doit donc intégrer ces deux modes de structurations de l'espace géographique.

2.3. Géométrie et topologie de l'espace

La définition de modèles de représentation de l'espace permet d'identifier un ensemble de propriétés géométriques et topologiques de l'espace. La représentation de données spatiales discrètes et continues est le plus souvent décrite à partir d'une géométrie euclidienne. Cette géométrie, intuitive dans nos cultures occidentales, permet de décrire la structure et les propriétés de l'information spatiale et les transformations possibles [Worboys 1995]. La structuration d'une information spatiale permet de définir les différents types de l'information spatiale selon les principes de cette géométrie. Les transformations possibles sont celles applicables dans l'espace euclidien représenté, elles modifient différemment les propriétés des entités manipulées (ex: une translation conservera la surface d'une entité, un changement d'échelle modifiera la surface de cette même entité).

La structuration d'un espace euclidien permet l'application des propriétés des espaces topologiques. Les propriétés topologiques, issues des sciences mathématiques, décrivent les relations spatiales invariantes par déformation géométrique continue. Une structuration topologique de données spatiales favorise une meilleure cohérence géométrique de l'information par la vérification de certaines contraintes spatiales (ex: partage de frontières entre entités voisines). Les propriétés topologiques de l'espace permettent également de constituer des réseaux logiques utiles pour la représentation physique de graphes. Différents modèles géométriques de références sont proposés pour la présentation de données spatiales discrètes, ils tendent souvent à discrétiser l'espace en unités géométriques élémentaires [Laurini 1992, Worboys 1995]³. Certaines approches plus mathématiques tentent de construire la représentation d'un espace à partir de primitives géométriques élémentaires [Frank 1986] ou d'un maillage de référence de l'espace [Güting 1993].

Un exemple de modèle spatial construit avec un ensemble minimal de primitives discrètes est le modèle basé sur le concept de **Erreur ! Source du renvoi introuvable.** et de **Erreur ! Source du renvoi introuvable.** [Frank 1986] (Figure 2.1.). Un *d-simplex* représente un objet de dimension *d* (ex: un *0-simplex* représente un point, un

³ Dans le cas des données continues, les propriétés géométriques sont implicitement décrites par la structure de maillage de l'espace utilisée.

1 -simplex représente un segment de ligne). Tout d -simplex est composé de $d+1$ **Erreur ! Source du renvoi introuvable.** de dimension $d-1$ (ex: un segment de ligne de dimension 1 est composé de 2 points de dimension 0). Les *simplex* permettent de constituer des *faces* dans le cas de la représentation d'objets bi-dimensionnels (i.e. 2-simplex). Un *simplicial complex* est un ensemble fini de *simplices* et de leurs *faces*.

Figure 2.1. - Simplex et simplicial complex

Un deuxième exemple est le modèle Gral, basé sur le concept de d segments de ligne finis sur une grille de point avec les propriétés suivantes : (1) chaque point ou noeud d'un segment de ligne est un point de la grille (2) chaque noeud final d'un segment de ligne est aussi un point du *Realm* (3) aucun point du *Realm* n'est membre d'un segment de ligne (4) deux segments du *Realm* ne peuvent s'intersecter qu'à leurs noeuds finaux.

Figure 2.2. - Exemple de Realm

Si ces modèles possèdent l'avantage de représenter l'espace sous une forme plus performante sur le plan des propriétés géométriques et des performances de traitement induites, elles sont cependant reductrices sur le plan des possibilités offertes dans la représentation des entités du monde réel. De nombreuses propositions de modèles spatiaux discrets sont également définies et notamment des tentatives d'extensions qui proposent des modèles intégrant la troisième dimension spatiale [Jones 1989] ou la dimension temporelle [Worboys 1994, Galton 1995, Peuquet 1994 et 1995, Claramunt 1995b et 1996a]. Dans le contexte de notre recherche, nous n'aborderons pas la définition de propriétés géométriques ou topologiques, nous traiterons plus directement ces caractéristiques comme une propriété de l'information spatiale.

2.4. Organisation et partition de l'espace

Les besoins de gestion de larges volumes de données spatiales imposent d'organiser et de regrouper des ensembles de données homogènes en cartes numériques thématiques ou couches (ex: une carte pédologique, une planche cadastrale). Ces cartes numériques constituent des partitions thématiques de l'espace. Ces partitions permettent à la fois de décrire des données spatiales thématiques (i.e. définies par des représentations spatiales ponctuelles, linéaires, polygonales et géo-composées) et/ou représentant des réseaux de communication ; ces partitions sont explicites dans le cas de représentations continues de l'espace. Les principes d'organisation et de partition des données utilisés par les SIG sont définis à partir de différents modèles logiques. Les solutions proposées relèvent souvent d'une vision cartographique et sont contraintes dans la mise en oeuvre par des difficultés techniques de gestion de partition de l'espace.

La première génération de SIG est essentiellement basée sur un principe de partage de frontières communes d'entités polygonales : une polygone décrit dans sa structure logique son polygone à droite et son polygone à gauche. Ce modèle autorise un seul polygone à droite et un seul polygone à gauche d'où le caractère exclusif de la partition, ce qui est pénalisant sur le plan de la qualité de la sémantique représentée mais plus facilement implémentable pour les concepteurs de logiciels. Ces solutions sont communément désignées comme des modèles

droite/gauche. Si ces modèles facilitent l'organisation et la gestion des données spatiales en thèmes ; ils n'autorisent pas par contre les partages multiples de primitives graphiques et entraînent donc des problèmes de redondance d'information et de consistance. Des entités spatiales parcelles et bâtis peuvent, par exemple, posséder des limites spatiales communes, la duplication de cette limite impose la définition de contraintes pour éviter les inconsistances lors d'opération de mise à jour. L'organisation de cartes numériques en partitions de l'espace exclusive est cependant largement utilisée par les SIG actuels (ex: ArcInfo d'ESRI, Genamap de Genasys, MGE d'Intergraph).

Une deuxième famille de SIG, plus proche des principes définis par les recherches dans le domaine de la définition de modèles spatiaux, utilise un principe de séparation des données géométriques (la définition des primitives spatiales) de la description logique des entités spatiales [Poiker 1975]. Ces modèles permettent le partage de primitives géométriques (ex: une polygone limite à la fois d'une entité bâtie et d'une entité parcelle). Ils sont fondamentalement utiles pour certaines applications (ex: des cartes de servitudes ou d'impact d'inondations avec recouvrement de polygones, applications cadastrales avec partages fréquents de primitives spatiales). Ces modèles sont utilisés par des logiciels de numérisation comme Cartier de l'Université Laval, Roots de l'Université Harvard, et par le SIG System9 (Prime et ComputerVision).

En tout état de cause la définition de ces contraintes spatiales thématiques relève d'un processus de structuration d'un ensemble de données spatiales. Elle doit être réalisée à un bas niveau du modèle de données spatiales. Dans la mesure où notre démarche s'attache essentiellement à caractériser les propriétés sémantiques des données spatiales et leur rôle dans les opérations de manipulation, nous assumerons ces propriétés dans notre modèle sans nous attacher à les décrire d'un point de vue formel. Nous ne représenterons donc pas ces contraintes structurelles à travers notre modèle de référence (si ce n'est par l'appartenance potentielle d'un ensemble de données d'une carte numérique de multiples classifications).

2.5. Définition et sémantique des entités

La représentation d'un espace peut être perçue comme un lieu d'expression de relations entre un ensemble d'entités de cet espace [Gatrell 1991]. Cette définition relève d'une perception discrète et structurée de l'espace, elle facilite la modélisation d'un système géographique. Elle est utilisée comme le principe de référence dans le développement de ce modèle de données spatiales.

Une **entité** est la représentation d'un élément d'information du monde réel. Une entité a un caractère spatial si elle possède des propriétés pouvant être rattachées à une localisation dans l'espace [Smith 1987]. Les entités sont les éléments fondamentaux de détermination et d'organisation des propriétés d'une application. Une **classe** d'entités définit une collection d'entités ayant des caractéristiques semblables (ex: des parcelles, des bâtis, des rivières). La définition des attributs d'une classe d'entités est définie par la notion de **type**. Le type décrit les caractéristiques non spatiales (i.e. alphanumériques) et les caractéristiques spatiales d'une classe d'entité.

Les caractéristiques spatiales d'une entité sont définies par un attribut abstrait que nous désignons par le terme générique de **représentation spatiale**. Dans le contexte de cette étude, la représentation spatiale d'une entité est décrite par un type abstrait de données (TAD). Un TAD permet d'étendre les types classiques de données et de

représenter les objets complexes [Stemple 1986, Adiba 1987]. Cette technique est appropriée à la définition d'une représentation spatiale dont la structure est par nature complexe [David 1991]. Un TAD permet de définir la spatialité d'une entité sans en préciser les détails de sa structure; seules les propriétés et les opérations disponibles sont accessibles à l'utilisateur. Les caractéristiques alphanumériques d'une entité sont précisées par des attributs définis sur un domaine de valeur (ex: string, integer). Ces domaines traduisent des représentations ordinales, nominales ou cardinales d'une information [Stevens 1946]. La sémantique de l'information alphanumérique d'une entité à caractère spatial est classifiée en trois groupes orthogonaux [Mainguenaud 1994] (Figure 2.3.):

- Le **Granule** soit la validité d'un attribut sur la totalité ou sur une partie spatiale de la représentation spatiale d'une entité. Cette validité est définie respectivement par les valeurs de *Granule (integral et sub-set)*. Les valeurs de *Granule* sont contraintes pour les attributs identifiant, clés primaires ou clés candidates (valeur *Granule: subset*) (la valeur d'un attribut identifiant reste valide pour une partie spatiale de la représentation spatiale d'une entité, son rôle d'identifiant n'est par contre pas conservé)
- Le **Recouvrement Spatial** (*Set-relationship*) soit la possibilité pour des instances d'une même classe d'entités de se superposer spatialement ou de ne pas se superposer spatialement. Cette possibilité est définie respectivement par les valeurs de (*Set-relationship: overlap*) et (*Set-relationship: non overlap*).
- La **Validité Topologique** (*Topology*) soit la validité d'un attribut sur l'intérieur/frontière/totalité de la représentation spatiale d'une entité, la validité topologique est définie respectivement par les valeurs (*Topology: interior, boundary, global*).

L'identification de cette sémantique permet une meilleure appréciation de la portée des attributs des entités. La spécification de ces facteurs sémantiques est fondamentale pour améliorer le contrôle de la qualité et de la validité des traitements des bases de données spatiales. Dans le cas contraire, l'utilisateur reste libre d'interpréter les données, ce qui entraîne inévitablement des risques d'erreurs lors d'opérations de manipulation et la nécessité de définir la validité des informations alphanumériques associées au résultat d'une opération spatiale.

Figure 2.3. - Sémantique alphanumérique d'une entité

2.6. Construction d'une base de données exemple

L'exemple suivant illustre l'intégration de la représentation spatiale dans un schéma logique d'une base de données (Table 2.4.). La description logique de la base est formulée à partir d'un formalisme relationnel étendu par la notion de type abstrait de données (TAD). Nous utilisons ce formalisme comme support de présentation et d'illustration des notions proposées, le modèle relationnel ayant l'avantage d'être connu et bien standardisé. Les concepts présentés sont cependant indépendants du modèle relationnel et peuvent être exprimés à partir d'autres formalismes avec quelques légères adaptations.

Bâti	(<u>Id</u> ,Affectation,Surface_Utile,Taux_Bruit, Représentation_Spatiale)
Parcelle	(<u>Id</u> , Utilisation, Représentation_Spatiale)

Route_e (Id, Origine, Destination, Catégorie, Représentation_Spatiale)

Route_n (Id, Niveau, Représentation_Spatiale)

Rivière (Id, Largeur, Représentation_Spatiale)

Zone_Inondable (Id, Rivière.Id, Année, Représentation_Spatiale)

Table 2.4. - Schéma logique de la base de données exemple

La base exemple retenue décrit le site d'un campus universitaire. Les relations de cette base sont des relations à caractère spatial. Un graphe routier est représenté par deux relations décrivant des arcs et des noeuds (Route_e et Route_n). La Table 2.5. présente les données alphanumériques des entités de la base exemple. Le graphe de la route est décrit par deux relations *Route_n* et *Route_e* qui modélisent respectivement les noeuds et les arcs de ce graphe. L'attribut Niveau de la relation *Route_e* précise si le noeud d'une route est un carrefour (valeur 1), un noeud de passage (valeur 2), un noeud de fin de route (valeur 3) ou un noeud de passage vers un autre graphe de route (valeur 5).

Bâti	<u>Id</u>	Affectation	Surface_Utile	Taux_Bruit
	1	Administration	600	30
	2	Informatique	1 000	30
	3	Informatique	1 000	60
	4	Maths	550	60
	5	Maths	600	60

Parcelle	<u>Id</u>	Utilisation
	101	Gestion
	102	Enseignement
	103	Enseignement
	104	Mixte

Route_n	<u>Id</u>	Niveau
	n1	1
	n2	2
	n3	2
	n4	3
	n5	1
	n6	1
	n7	5
	n8	5

Route_e	<u>Id</u>	Origine	Destination	Catégorie
	201	n7	n5	Principale
	202	n5	n6	Principale
	203	n6	n8	Principale
	204	n5	n1	Principale
	205	n1	n3	Principale
	206	n2	n6	Principale
	301	n3	n2	Secondaire
	302	n4	n5	Secondaire

Rivière	<u>Id</u>	Largeur
	501	6

Zone_Inondable	<u>Id</u>	Rivière.Id	Année
	301	501	1965
	302	501	1985

Table 2.5. - Données alphanumériques des entités de la base exemple

La sémantique du Granule des entités de la base de données exemple est précisée par la Table 2.6. (une notation pointée est utilisée pour décrire les attributs de relation). Les valeurs d'attributs *Bâti.Surf_Utile*, *Bâti.Taux_Bruit* et *Rivière.Largeur* ne sont pas valides pour des parties de représentation spatiales de toute entité représentée (valeur *Granule: integral*). Les valeurs d'attributs identifiants sont valides pour des parties de représentation spatiales de toute entité représentée (valeur *Granule: subset*), leur rôle d'identifiant n'est par contre plus valide dans la mesure où cette valeur peut être partagée entre plusieurs parties spatiales de cette entité. Les autres attributs sont valides pour des parties de représentation spatiale (valeur *Granule: sub-set*).

Granule	integral	sub-set
Bâti. <u>Id</u>		x
Bâti.Affectation		x
Bâti.Surf_Utile	x	
Bâti.Taux_Bruit	x	
Parcelle. <u>Id</u>		x
Parcelle.Utilisation		x
Route. <u>Id</u>		x
Route.Catégorie		x
Rivière. <u>Id</u>		x
Rivière.Largeur	x	
Zone_Inondable. <u>Id</u>		x
Zone_Inondable.Rivière.Id		x
Zone_Inondable.Année		x

Table 2.6. - Sémantique alphanumérique des entités: Granule

La valeur *overlap* du *Set-relationship* pour la relation *Zone_Inondable* traduit le fait que deux entités de cette classe peuvent se superposer spatialement dans la mesure où elles résultent de représentations spatiales définies à des années différentes. Les autres entités de la base exemple ne sont pas superposables spatialement (valeur *Set-Relationship: non overlap*) (Table 2.7.).

Set-relationship	overlap	non overlap
Bâti		x
Parcelle		x
Rivière		x
Route		x
Zone_Inondable	x	

Table 2.7. - Sémantique alphanumérique des entités: Recouvrement Spatial

La Table 2.8. présente les valeurs de la Validité Topologique (*Topology*) pour les attributs de l'entité *Bâti*. L'attribut *Bâti.Taux_Bruit* présente la particularité d'être valide sur le contour spatial d'une entité (valeur *Topology: boundary*).

Topology	interior	boundary	global
<i>Bâti.Id</i>			x
<i>Bâti.Affectation</i>	x		
<i>Bâti.Surf_Utile</i>	x		
<i>Bâti.Taux_Bruit</i>		x	

**Table 2.8. - Sémantique alphanumérique des entités
de la base exemple: Validité Topologique**

2.7. Mécanismes d'abstraction

Un mécanisme d'abstraction désigne une opération qui permet de modifier la sémantique d'entités ou de types d'entités. Ces mécanismes permettent la définition de hiérarchies de classes au sein du schéma d'une base de données et donc de règles d'héritages de propriétés. Ce sont à la fois des outils de modélisation et des mécanismes utiles dans la gestion d'une base de données.

Plusieurs auteurs proposent une analyse des mécanismes d'abstraction nécessaires aux données spatiales [Brodie 1984, Schrefl 1984, Nyerges 1991, Lagrange 1994, Rigaux 1994]. Nous distinguons quatre classes parmi les mécanismes d'abstraction de données spatiales :

- La **généralisation sémantique** désigne par une même classe des classes d'entités distinctes (ex: généraliser différentes classes d'édifices en bâti). La généralisation sémantique est une opération équivalente à la création d'une super-classe. L'application de la généralisation sémantique ne pose pas de problème particulier sinon le respect de règles de cohérence sémantique.
- La **généralisation cartographique** transforme les primitives cartographiques d'entités (ex: fonctions de lissage, d'exagération) [Brassel 1987, McMaster 1991]. Une opération de généralisation cartographique est réalisée à partir d'algorithmes de transformation cartographique permettant de passer d'une représentation spatiale d'un niveau de représentation vers une échelle plus petite [Brassel 1987, Lagrange 1994]. La part d'expertise de ces transformations reste cependant importante, de nombreuses recherches sont toujours développées pour améliorer l'expertise liée à la généralisation cartographique [Müller 1995].

- L'**agrégation sémantique** regroupe dans une classe composée des entités de types distincts selon un critère d'association sémantique (ex: des instances de parcelle et de b_t_i composent ensemble des propri_t_s fonci_res);

- L'**agrégation cartographique** regroupe spatialement les primitives cartographiques d'entités d'une classe en nouvelles entités d'une même classe (ex: des instances d'entités bâtis qui s'agrègent spatialement en blocs d'entités bâtis lors d'un changement d'échelle).

Les mécanismes d'abstraction permettent la définition dynamique de nouvelles entités et l'héritage de propriétés pour les relations de généralisations. Ils donnent une plus grande richesse s_mantique lors de la r_alisation d_un sch_ma de bases de donn_es et une meilleure efficacit__ la gestion de cette m_me base en minimisant notamment les redondances d_information.

2.8. Représentation multiple

La représentation multiple caractérise l'existence, au sein d'un même système, de plusieurs représentations pour une même entité. Une entité à représentation multiple traduit des appréciations différentes lors de la modélisation (i.e. différents utilisateurs) ou des échelles de perception variables (ex: un bâti défini par un tracé précis à une échelle cadastrale et par une simplification de ce contour à une plus petite échelle). Une représentation multiple liée à différents points de vues utilisateurs entre dans le cadre des probl_mes classiques des bases de donn_es. Le ph_nom_ne de repr_sentation multiple, cons_quence des changements structurels g_om_triques et topologiques qui peuvent se produire lorsque l__chelle de d_finition d_une entit_est modifi_e, est par contre une originalit_des donn_es spatiales [NCGIA 1993].

L'intégration de la complexité du concept de représentation multiple dans les modèles de données spatiales reste encore un problème non résolu par les SIG et la recherche dans ce domaine. Les SIG actuels sont en effet essentiellement basés sur des modèles d_entit_s_ un seul niveau d__chelle et avec cohabitation sans relations de diff_rents points de vues et repr_sentations. Les repr_sentations multiples apparaissent uniquement et indirectement _travers plusieurs représentations spatiales de mêmes instances d'entités sans liens explicites. Ces représentations ne conservent aucun lien sémantique ou logique et entra"nent donc des redondances d'informations [Beard 1991]. Un des challenges de la recherche en SIG est la génération d'une base de données multi-échelle. Ce dernier objectif s'accompagne de plusieurs enjeux de recherche:

- La définition de modèles de données spatiales intégrant les représentations multiples [NCGIA 1993]. Les recherches sont toujours actives autour de cette probl_matique, citons notamment - sans les développer - les approches formelles et mathématiques [Frank 1994a, Puppo 1995].
- L'identification de liens sémantiques entre les différents niveaux de représentations multiples et la gestion associée des consistances [Buttenfield 1990, Beard 1991 et Devogele 1996]. Deux modes de généralisation peuvent être utilisés: la généralisation sémantique et la généralisation cartographique. C'est la combinaison de ces deux paramètres qui permet de d_crire toute la s_mantique des op_rations de changement d__chelle

[Devogele 1996]. Les différentes représentations spatiales d'une entité peuvent partager des valeurs d'attributs (ex: le nom du propriétaire d'un bâtiment), certaines informations topologiques (ex: les parcelles voisines d'une parcelle) voire certaines informations métriques (ex: la longueur d'une rivière). Beard propose une description d'opérateurs qui peuvent produire ou se connecter avec des représentations à différents niveaux de détail [Beard 1991]. La description des entités à chaque niveau de perception spatiale et celle des connexions sémantiques entre ces niveaux d'abstraction peuvent être formalisées sous la forme de tables de correspondances ou de règles [Devogele 1996]. Ces règles de changement d'échelle doivent contribuer à identifier des règles de propagation des mises à jour entre les différents niveaux de perception d'une représentation multiple.

- La coordination des opérations de manipulation d'une base de données multi-échelle [Devogele 1996]. Il s'agit de spécifier les conditions d'application des processus d'interrogation, d'analyse et de visualisation de données spatiales dans un contexte de représentation multiple.
- Le traitement cartographique de la représentation multiple [Brassel 1987, Müller 1995]. Müller aborde le problème de la formalisation des événements (au sens d'un fait de généralisation) par des règles qui pourraient être interprétées et intégrées dans un système de connaissance [Müller 1990]. Le processus cognitif doit être tout d'abord conceptualisé et ensuite représenté dans une forme logique. Un processus de généralisation demande un dialogue entre le système et l'utilisateur. Le processus est ordonné et séquentiel, une des importantes difficultés est l'interpolation des différentes règles de généralisation (relative similitude avec les problèmes rencontrés par les transactions multiples). Il est possible de distinguer les faits concrets (pas de généralisation pour des échelles plus grandes que 1: 5 000 par exemple, ou règles de généralisation pour un passage au 1: 25 000) des faits généraux (quand l'échelle d'croît, les éléments linéaires sont surévalués par rapport aux éléments surfaciques, ou encore le nombre de classes représentées diminue, les remplissages de la carte augmentent). Ces propriétés particulières demandent une identification et une maîtrise de la sémantique associée à une opération de généralisation selon l'échelle de perception et les objectifs recherchés. Les opérations de généralisation les plus couramment utilisées sont les simplifications, les exagérations, les rehaussements qui affectent une entité; les regroupements ou éclatements d'entités qui affectent plusieurs entités [McMaster 1987, Timpf 1995].

La définition de bases de données spatiales multi-échelles est une demande importante des producteurs de cartes (ex: instituts cartographiques comme l'IGN en France ou l'Office Fédéral de Cartographie en Suisse) et des gestionnaires de données géographiques (ex: concessionnaires de réseaux électriques). Les bénéfices attendus sont économiques par les gains de saisie qui pourraient être induits, ils favorisent une meilleure gestion grâce à une meilleure cohérence entre les différentes échelles de travail.

2.8.1. Un exemple de modèle : Geographic Data File

Le modèle Geographic Data File présente un exemple de solution d'implantation apportée au phénomène de représentation multiple dans un contexte de base de données routières. C'est en effet dans le domaine des réseaux

que les besoins d'application de la représentation multiple sont les plus sensibles que ce soit pour des applications routières ou de réseaux de distribution publics. Le projet Geographic Data File (GDF) est un exemple pré-opérationnel de format de données orienté réseau. Il préfigure les modèles qui seront utilisés par les futurs systèmes européens de navigation (i.e. guidage des automobiles) [Heres 1993]. Le modèle de données spécifie les entités avec leurs références géométriques et topologiques, les attributs de et les relations décrivant ces entités. Un modèle de généralisation cartographique est défini par le modèle GDF comme l'association de règles d'agrégation de l'information spatiale avec des processus de représentation de cette même information. Il est défini à partir de primitives cartographiques pour trois différents niveaux de représentation et de règles de représentation pour chaque classe d'entités. Les trois niveaux de représentation, en partie emboîtés, sont des structures différentes d'une même démarche de représentation (Figure 2.9).

**Figure 2.9. - Modèle GDF de généralisation cartographique
d'après [Heres 1993]**

Chaque niveau de représentation est associé à un ensemble de primitives cartographiques. Une schématisation précise les relations des primitives cartographiques à travers les différentes formes de visualisation. Le projet GDF contribue à mettre en évidence le fait que le modèle doit intégrer les différents niveaux de représentation de l'information spatiale. C'est donc dans ce cas l'association du modèle de généralisation cartographique avec le modèle décrivant les relations des primitives cartographiques à travers ces niveaux de représentation qui a pour objectif d'élargir la qualité de la représentation d'une entité.

L'Office de Cartographie Britannique (Ordnance Survey) développe une base de données routières avec notamment l'intégration de différents points de vue selon les recommandations de ce format GDF [Wood 1993]. Les besoins d'intégration d'applications et de maintien de la consistance des bases de données imposent à l'Office Britannique de Cartographie cette approche par vue utilisateur. Cette proposition a pour objectif de favoriser l'émergence de passerelles de communication entre différentes bases de données, d'introduire des mécanismes de mises à jour efficaces et d'améliorer le partage des données et de réduire finalement les coûts de maintenance de l'ensemble. Les exigences de ces vues utilisateurs se définissent en terme de détail de définition du réseau, de

définition sémantique des données, et de règles de connectivité et d'intégrité. Pour chaque type d'application, et donc de vues utilisateurs, sont définies des classes d'applications qui affinent le niveau de détail du réseau, sa structure, et ses conditions de maintenance. Ces classes existent sur un continuum du réseau et constituent des vues du réseau réel. Elles relèvent d'un même niveau d'interprétation mais peuvent se recouper en partie. La construction de vues utilisateurs (au sens graphique) se fait par la dérivation de caractéristiques du réseau à partir du réseau de base, et à partir de règles topologiques (ces règles sont gérées par des programmes d'application). La construction de vues détaillées ou simplifiées est possible à partir de cette architecture.

2.8.2. Discussion

L'intégration de la représentation multiple dans la définition des entités impose une certaine rigueur dans les procédures de mise à jour de bases de données spatiales. Une opération de mise à jour peut ne pas être uniformément applicable [Buttenfield 1990] (ex: le changement spatial d'un tracé d'un bâti à grande échelle peut ne pas affecter ce même tracé à petite échelle). Certaines opérations de mise à jour de données descriptives doivent par contre répercuter leurs effets pour toutes les représentations spatiales (ex: le changement de nom d'affectation d'un bâti est valable pour toutes les représentations spatiales). Selon Haller, les changements d'échelle de représentation en fonction du contexte de la carte _ exprimer, transforment notre perception de cet espace représenté [Haller 1990]. Les relations vont évoluer avec cette dynamique : une relation de type "dans" pourra devenir une relation de type "à côté" (ex: une zone inondable qui s'intersecte avec un réseau routier défini à petite _chelle, la m_me zone inondable qui ne s_intersecte pas avec une g_n_ralisation de ce m_me r_seau routier _ une plus petite _chelle).

Becker propose une application originale : la représentation multiple peut permettre d'apprécier à différents niveaux de pr_cision (et donc avec diff_rents temps de r_ponse) une m_me requ_te g_ographique [Becker 1990]. Cette m_thode est int_ressante pour appr_cier en premi_re approximation une requ_te g_ographique complexe et co_teuse en temps de calcul. Les mod_les spatiaux intégrant la représentation multiple permettent par extension la prise en compte de différents points de vue utilisateurs.

En l'absence de connaissances de solutions facilement identifiables et spécifiables, les SIG actuels ne proposent pas de solutions solides au probl_me de repr_sentation multiple. Certains _diteurs de SIG proposent cependant quelques premi_res avanc_es dans le domaine. Citons la notion de r_pliqua propos_e par le SIG Vision de GeoVision qui permet d_associer _ des entit_s une double représentation cartographique partageant des attributs alphanumériques, soit une représentation schématique et géométrique pour des applications de gestion de réseaux de distribution [VISION 1995].

Les applications de la représentation multiple sont nombreuses et couvrent les domaines de la cartographie traditionnelle comme des applications de gestion d_itin_raire et notamment la description de processus de navigation. Dans ce dernier cas, la d_finition d_un parcours va utiliser plusieurs r_f_rences cartographiques successives, dont les échelles vont varier de la plus générale (définition de la direction générale du parcours) à la plus précise (définition du départ, des points d'arrêts et de l'arrivée). Le modèle doit ici tenter de répondre à un double objectif soit (1) intégrer plusieurs niveaux d'abstraction et également (2) les relier sémantiquement pour permettre à la description du processus de navigation de **Erreur ! Source du renvoi introuvable.** ces différents niveaux d'abstraction.

2.9. Proposition d'un modèle spatial étendu

L'étude des caractéristiques des données spatiales nous donne des éléments de référence pour la construction d'un modèle support de notre recherche. Dans ce sens nous retiendrons les propriétés spatiales qui ont un rôle actif dans les opérations d'interrogation et de présentation de données spatiales. Nous considérons notamment que les

composantes cartographiques de structuration, et géométriques de description des entités spatiales constituent des informations de bas niveau qui sont implicitement prises en compte par le mod_le. La g_om_trie des entit_s spatiales est notamment assum_e par notre mod_le comme une r_f_rence de type abstrait (au sens des types abstrait de donn_es utilis_s en bases de donn_es).

Par contre, le concept de multi-représentation est int_gr_ dans notre mod_le de r_f_rence dans la mesure o_ il implique une variation potentielle de repr_sentation cartographique qui a des cons_quences importantes sur les op_rations de manipulation et de pr_sentation. Nous d_taillerons cette composante et préciserons ses propriétés dans la suite de cette section. Le modèle proposé s'attache également à décrire les informations stables utilisées dans les processus de présentation et de visualisation de données spatiales, et notamment les variables de visualisation (i.e variables de Bertin) ainsi que certaines informations contextuelles qui accompagnent la sémantique des entités présentées.

Nous précisons la sémantique spatiale d'une entité à partir du type abstrait de la représentation spatiale. Les attributs qui décrivent la représentation spatiale sont les suivants :

Représentation_Cartographique

Une représentation cartographique est définie par un type abstrait TAD (*Représentation_type*) qui permet la prise en compte extensible de types géométriques bi-dimensionnels (ex: point, ligne, polygone, image) [selon Larue 1993].

Référentiel_Observation

Le référentiel d'observation (*Référentiel_Observation*) permet de définir le niveau d'abstraction d'une représentation spatiale. Dans le cadre des applications géographiques, les valeurs possibles que nous proposons, à titre d'exemple, sont *global*, *régional* et *local*. Ces valeurs qualifient des niveaux de perception complémentaires souvent utilisés dans les applications de SIG, d'autres domaines de valeurs sont définissables par les utilisateurs. Cette information s_mantique qualitative r_f_rence le niveau de perception du ph_nom_ne_tudi_. Elle permet de regrouper des donn_es d_finies _des niveaux d_abstraction semblables et de favoriser ainsi des traitements sur des donn_es relativement homogènes sur le plan de leur perception et de leur définition.

Echelle_Validité

L'échelle de validité précise l'intervalle de manipulation possible d'une représentation spatiale, qu'elle soit définie à partir du référentiel d'observation d_origine ou d_riv_e si l_entit_d_crite est issue d_une op_ration spatiale.

L'échelle de validité est définie comme un type complexe qui précise l'échelle de saisie ou de définition de la représentation spatiale (*Echelle_Capture*), et l'intervalle de limite de validité de la représentation spatiale pour les opérations de traitement et de visualisation (*Echelle_Minimum* et *Echelle_Maximum*). Ces deux dernières caractéristiques permettent de donner l'intervalle d'échelle dans lequel il est possible de manipuler ou de visualiser cette repr_sentation spatiale. Cette qualification est importante pour des op_rations d_analyse spatiale et de cartographie, elle permet de d_terminer ou d_appr_cier la faisabilit_d_un traitement sur ces donn_es (ex: g_n_ralisation, croisements de cartes numériques) et l'intervalle d'échelle utilisable dans les processus de

visualisation. L'échelle de validité d'une représentation cartographique peut être caractérisée par une fonction d'appartenance définie sur l'intervalle d'échelle correspondant selon une fonction lin_aire ou floue (i.e. une fonction d_appartenance qui tend vers une valeur d_appartenance_gale_1 pour la valeur de l__chelle de saisie et vers 0 sur les limites de l_intervalle de d_finition de l__chelle de validit_). Ces fonctions d'appartenances donnent des éléments d'appréciations de la validité des résultats d'une opération spatiale. La représentation spatiale résultant d'une opération spatiale peut être en effet qualifiée par une échelle de validité qui est l'intersection des échelles de validité des représentations spatiales argument de l'opération spatiale et par une fonction d'appartenance qui est l'intersection des fonctions d'appartenance des représentations spatiales argument de l'opération spatiale.

Mode

La représentation cartographique d'une entité peut être décrite à partir de deux catégories de primitives : réelles et virtuelles [Ooi 1989]. Les représentations cartographiques à caractère réel (*réel*) correspondent à une réalité spatiale perçue et possèdent une spatialit_ qui a une valeur significative (la limite spatiale correspond _ une limite r_elle observ_e) alors qu'une repr_sentation cartographique de type virtuel (*virtuel*) correspond à une représentation cartographique schématique sans portée spatiale autre que le message transmis par le graphisme utilisé (la limite spatiale ne correspond pas à une limite de l'entité dans l'espace). Les opérations spatiales sont par conséquence applicables sur des repr_sentations spatiales de *Mode* réel. Inversement, la qualit_des r_sultats n_est pas assur_e dans le cas d_utilisation de repr_sentations virtuelles. Ces repr_sentations virtuelles peuvent _tre cependant utilis_es pour des approximations d_op_rations spatiales dans le cas de larges et complexes bases de donn_es spatiales, ces opérations donnent dans ce cas une première évaluation du résultat.

Visualisation

Les formes utilisées dans les processus de visualisation sont dépendantes des représentations cartographiques utilisées. Dans le cas de la cartographie, une visualisation est construite _partir d_un syst_me de symbologie. Les primitives employ_es (i.e. forme, trame, orientation, couleur, taille, contraste) sont d_crites par les variables de Bertin [Bertin 1983]. Ces variables sont repr_sent_es dans ce mod_le par un type complexe (*Bertin_type*). Un label qualifie textuellement le thème du processus de visualisation décrit (*Label*).

La spatialité d'une entité est représentée par un type complexe (*Représentation_Spatiale_type*). La structure de la représentation spatiale est compos_e d_une agr_gation d_attributs spatiaux et non spatiaux. La notation des objets complexes permet de d_finir des valeurs complexes non atomiques, les constructeurs disponibles sont l_ensemble { }, la liste (), l_agr_gation [] et _num_r_ <>. La représentation spatiale d'une entité est précisée à partir de cette notation.

Représentation_Spatiale_type:

[Représentation_Cartographique:

Référentiel_Observation:

Echelle_Validité:

Représentation_type ,

<'global', 'régional', 'local'> ,

[Echelle_Minimum: Real ,

Echelle_Capture: Real ,

Echelle_Maximum: Real] ,

```

Mode: < 'réel', 'virtuel'> ,
Visualisation: [Fonction_Visualisation: Bertin_type ,
Label: string ]

```

L'exemple suivant présente la valeur de l'attribut *Représentation_Spatiale* de l'instance de *Bâti* d'identifiant *Id* = 5 (des valeurs complexes *polygone* et *hash* sont utilisées pour décrire respectivement les attributs *Représentation_cartographique* et *Bertin_type*). Afin de faciliter la lecture de l'exemple, les valeurs complexes sont d_crites par une notation en majuscule sur le type correspondant (ex: POLYGONE et HASH).

```

Représentation_Spatiale=
  [Représentation_Cartographique= POLYGONE ,
  Référéntiel_Observation= 'local' ,
  Echelle_Validité= [Echelle_Minimum: 1/ 10 000 ,
  Echelle_Capture: 1/5 000 ,
  Echelle_Maximum: 1/2 500 ] ,
  Mode= 'réel' ,
  Visualisation= [Bertin_type: HASH,
  Label: 'fonds_de_plan' ] ]

```

La Figure 2.10. illustre les représentations spatiales des entités de cette base exemple à travers deux référentiels d'observation (*Référentiel_Observation*) *local* et *régional*. Les entités des relations *Bâti*, *Parcelle* et *Route* sont définies par un référentiel d'observation de valeur *local*, les entités des relations *Rivière* et *Zone_Inondable* sont définies par un référentiel d'observation de valeur *régional*.

Figure 2.10. - Représentations spatiales des instances de la base exemple

Les identifiants des représentations spatiales des instances du graphe de route représenté par les relations *Route_n* et *Route_e* sont illustrées par la Figure 2.11.

Figure 2.11. - Illustration du réseau routier de la base exemple

Les représentations spatiales de la base exemple présentent les caractéristiques principales suivantes :

- Les deux valeurs de référentiels d'observation (*Référentiel_Observation*) utilisées (*local* et *regional*) traduisent des niveaux d'abstraction complémentaires.
- Les primitives cartographiques des instances présentées sont de (*Mode: réel*), à l'exception de la représentation spatiale des instances de route qui traduisent une vision schématique du réseau routier traduit par des représentations cartographiques de (*Mode: virtuel*).
- L'attribut de visualisation (*Visualisation*) des entités *Bâti* utilise des variables de Bertin (*Bertin_Variable*) de type hachure (*hash*) ; la valeur de *Label* utilisée par les instances exemple de la représentation spatiale de la relation *Bâti* traduit une symbologie utilisable pour un objectif de fonds de plan (valeur *Label: fonds_de_plan*).

L'intégration de la représentation multiple est réalisée en étendant la définition du type *représentation spatiale* (*Représentation_Spatiale*) à un type ensemble. Le modèle permet la définition de plusieurs types de représentations spatiales pour une même échelle (i.e. différents points de vues pour une même échelle).

Représentation_Spatiale_type:

```
{ [Représentation_Cartographique:      Représentation_type ,
Référentiel_Observation:              <'global','régional','local'> ,
Echelle_Validité:                     [Echelle_Minimum: Real ,
Echelle_Capture:      Real ,
Echelle_Maximum:      Real ] ,
Mode:                                  <'réel','virtuel'> ,
Visualisation:                         [Fonction_Visualisation: Bertin_type ,
Label: string          ]          ] }
```

L'exemple suivant présente un exemple d'instance de *Bâti* (identifiée *Id=5*) possédant une représentation multiple :

```
Représentation_Spatiale =
{ [Représentation_Cartographique=      POLYGONE ,
Référentiel_Observation=              local ,
Echelle_Validité=                     [Echelle_Minimum=  1/ 10 000 ,
Echelle_Capture=                       1/5 000 ,
Echelle_Maximum=      1/2 500 ] ,
Mode=                                  real ,
Visualisation=                         [Fonction_Visualisation: HASH ,
Label: fonds_de_plan          ]          ] ,
[Représentation_Cartographique=      POLYGONE ,
Référentiel_Observation=              regional ,
Echelle_Validité=                     [Echelle_Minimum:  1/ 20 000 ,
Echelle_Capture:                       1/15 000 ,
```

```
Mode=                               Echelle Maximum: 1/10 000 ],
Visualisation=                       virtual ,
                                      [Fonction_Visualisation: HASH ,
                                      Label: plan_aménagement ] ] }
```

La Figure 2.12. illustre le phénomène de représentation multiple avec deux cartes définies à des niveaux d'observations (*Référentiel_Observation*) respectivement *local* et *regional*. Les représentations cartographiques des entités *Bâti* au niveau d'observation *regional* sont le résultat d'une généralisation des représentations cartographiques du niveau d'observation *local*. Les représentations cartographiques des entités des relations *Route_n* et *Route_n* valides au niveau d'observation *regional* sont les instances de valeur de *Catégorie Principale*.

Figure 2.12. - Représentation multiple

3. TRAITEMENT DE DONNEES SPATIALES

Les fonctionnalités de traitement des SIG fournissent de nombreuses possibilités d'utilisation de données cartographiques numériques. La variété et la qualité de ces outils permettent de tirer partie de toute la sémantique des modèles de données spatiales et de fournir ainsi des informations utiles aux utilisateurs de SIG. Les SIG présentent une large panoplie de traitements qui couvrent les fonctions d'interrogation et d'analyse de données géographiques [Burrough 1986, Smith 1987, Rhind 1990, Thériault 1996]. Les opérations d'interrogation sont orientées vers l'identification de propriétés de données existantes à partir d'un langage de manipulation. Les opérations d'analyses étudient la structure, la signification et la sémantique des données utilisées pour générer de nouvelles informations. Cette distinction entre les fonctions d'interrogation (orientées vers la gestion du territoire) et les fonctions d'analyse (orientées vers l'étude des phénomènes) de données spatiales se répercute dans les différentes catégories de SIG qui recouvrent généralement l'une ou l'autre de ces familles de fonctionnalités. Notre étude est orientée vers la définition d'un outil de manipulation flexible et ouvert de données spatiales. Le domaine couvert vise un potentiel large d'utilisateurs finaux. Les processus d'interrogations en feront naturellement partie. Les opérations d'analyses, qui font appel à des démarches exploratoires plus spécialisées, ne seront par contre pas prises en compte. Nous classifions les opérations d'interrogation de données spatiales par critères dominants:

- **interrogation relationnelle** : Ces interrogations sont définies comme des procédures de recherche de données réalisées à partir de critères alphanumériques classiques (ex: retrouver l'ensemble des parcelles d'un propriétaire).
- **interrogation spatiale** : Ces interrogations sont des procédures de recherche de données spatiales définies à partir de propriétés spatiales des entités (ex: retrouver les parcelles voisines d'une parcelle), ou d'opérateurs logiques de graphe (ex: identifier les chemins possibles entre deux nœuds identifiés d'un réseau).
- **interrogation temporelle** : Ces interrogations sont définies comme des procédures de recherche de données à partir de critères relationnels tendus par des opérateurs temporels (ex: retrouver les bûches qui ont changé d'affectation en 1995).

Nous orienterons nos opérations de traitement vers les interrogations relationnelles et spatiales dans la mesure où le domaine temporel reste encore un sujet non complètement stabilisé (opérateurs utiles dans un contexte de SIG notamment).

Cette étude est orientée vers la définition d'un niveau externe (i.e. d'un point de vue base de données) de traitement de données spatiales, la définition des opérateurs du modèle est centrée sur une approche intégrée des composantes relationnelles et spatiales des fonctions d'interrogation des bases de données. Les opérateurs étudiés sont classifiés en opérateurs de base (ceux du relationnels) et en opérateurs spatiaux (ceux nécessaires au spatial). Les opérateurs spatiaux sont catégorisés en **opérateurs thématiques** s'appliquant sur les représentations spatiales d'entités (i.e. entités d'un espace bi-dimensionnel) et en **opérateurs réseaux** s'appliquant sur des graphes

modélisant des réseaux logiques [Mainguenaud 1994 et 1997]⁴. Ce chapitre propose d'identifier un ensemble d'outils d'interrogations de données spatiales. La section 3.1. définit un ensemble d'opérateurs relationnels, spatiaux et de manipulation de référence, et évalue leur sémantique. La section 3.2. décrit un ensemble de requêtes construites à partir de cet ensemble d'opérateurs sur la base de données exemple. La section 3.3. présente une étude des caractéristiques des langages d'interrogation de données spatiales et la section 3.4. propose une discussion autour des potentialités et des limites de ces langages.

⁴ Dans les classifications généralement proposées, les opérateurs identifiés n'intègrent pas les opérateurs réseaux [Burrough 1986, Laurini 1992].

3.1. Définition des opérateurs

Les opérateurs identifiés dans cette partie ont pour objectif de constituer un noyau utile au traitement de données spatiales et de permettre de développer les outils de manipulation du modèle. En effet, et contrairement aux bases de données classiques, les propositions actuellement formulées dans le domaine du spatial ne permettent pas de dégager une référence commune définissant un ensemble minimal d'opérateurs spatiaux. Les opérateurs spatiaux à analyser sont de haut-niveau, ils ne sont pas définis à un niveau géométrique ou topologique (i.e. la définition formelle de relations topologiques entre différents types spatiaux dans un espace bi-dimensionnel); ce sont les opérateurs manipulés par un utilisateur dans l'expression de ses requêtes. Les opérateurs à identifier ont pour objectif de constituer un ensemble significatif à défaut de permettre la complétude. Les opérateurs s'appliquent sur le domaine des relations et des réseaux, ces derniers représentent comme des relations pour donner un caractère homogène au modèle, qui sont précisés par les définitions et les propriétés suivantes :

Une relation (Rel) est définie par un nom (R) et un ensemble d'attributs notés Att_i

L'attribut modélisant la représentation spatiale est nommé Représentation_Spatiale

Un graphe G est un couple (N,E) où N est un ensemble de noeuds et E un ensemble d'arcs identifiés, sous-ensemble du produit cartésien N x N.

Les arcs sont représentés par une relation Rel_e et les noeuds par une relation Rel_n.

Les relations représentant des noeuds (Rel_n) et les relations représentant des arcs (Rel_e) possèdent des attributs à sémantique particulière :

Rel_n : (Identifiant, Représentation_Spatiale, Att₁, ..., Att_p)

Rel_e : (Identifiant, Origine, Destination, Représentation_Spatiale, Att₁, ..., Att_q)

Les types de base utilisés dans la définition des opérateurs sont les suivants :

Rel (relation)	Rel _n (relation noeud)
Rel _e (relation arc)	Att (attribut)
bool (booléen)	num (numérique)
str (chaîne de caractères)	⊥ (valeur indéfinie)
Comp _{stan} (comparateur standard)	Comp _{stan} : = , > , < , ≠ , ≥ , ≤
Critère (critère de sélection)	

Table 3.1. - Types de base

Les sections suivantes catégorisent et décrivent les opérateurs utilisés dans notre modèle sur la base de ces notations. La section 3.1.1. décrit les opérateurs relationnels, la section 3.1.2. les opérateurs thématiques, la section 3.1.3. les opérateurs r_seaux et la section 3.1.4. les opérateurs de manipulation. La section 3.1.5. aborde les aspects sémantiques de ces opérateurs.

3.1.1. Opérateurs relationnels

Les opérateurs relationnels et leurs propriétés ont l'avantage d'être standardisés. Les opérateurs relationnels standards sont le produit cartésien (\times), la projection (Π), la différence ($-$), la sélection (σ) et l'union relationnelle (\cup).

$$X \in \{\text{Rel} \mid \text{Rel}_e \mid \text{Rel}_n\}, Y \in \{\text{Rel} \mid \text{Rel}_e \mid \text{Rel}_n\}, Z \in \{\text{Rel} \mid \text{Rel}_e \mid \text{Rel}_n\}$$

$$A = \{\text{Att}_1, \dots, \text{Att}_p\}, p \leq n$$

opérateurs relationnels

\cup	$X \times X$	-->	X
$-$	$X \times X$	-->	X
\times	$X \times Y$	-->	X^5
σ	$X \times \text{Critère}$	-->	X
Π	$\text{Rel} \times \text{Att}_1 \times \dots \times \text{Att}_p$	-->	Rel
	$\text{Rel}_e \times \text{Att}_1 \times \dots \times \text{Att}_p$	-->	Rel_e si Identifiant, Origine, Destination $\in A$ Rel sinon
	$\text{Rel}_n \times \text{Att}_1 \times \dots \times \text{Att}_p$	-->	Rel_n si Identifiant $\in A$ Rel sinon

Les opérateurs relationnels dérivés principaux sont l'intersection⁶ (\cap), la division⁷ (\div), et la jointure relationnelle⁸ (\bowtie).

⁵ Le produit cartésien devient non commutatif contrairement à l'algèbre relationnelle standard.

⁶ L'opérateur relationnel d'intersection \cap est un dérivé de l'opérateur relationnel de différence.

⁷ L'opérateur relationnel de division \div est obtenu à partir des opérateurs relationnels de différence, de produit cartésien et de projection [Codd 1970].

⁸ L'opérateur relationnel de jointure (\bowtie) est obtenu par le produit cartésien suivi d'une sélection relationnelle [Codd 1970].

3.1.2. Opérateurs thématiques

Un opérateur thématique est un opérateur spatial dont le résultat est une relation (Rel). Les opérateurs thématiques sont fermés sur l'ensemble des relations. Le résultat spatial de l'application d'un opérateur spatial thématique est représenté par l'attribut *Représentation Spatiale* de la relation résultante. Les propriétés géométriques des opérateurs thématiques relèvent de la spécification des opérateurs dans l'implémentation des TAD, elles ne sont pas détaillées dans cette étude. Les opérateurs thématiques identifiés constituent un ensemble de base qui nous permet d'illustrer les concepts définis dans cette étude. Les symboles utilisés pour désigner les opérateurs thématiques sont présentés par la Table 3.2 :

Adjacence:	$\supset\subset$	Buffer:	¥
Union spatiale:	\cup^S	Intersection spatiale:	\cap^S
Inclusion spatiale:	\subseteq^S	Vol d'oiseau:	•--•
Début chemin:	•-->	Fin chemin:	-->•
Comp_spa (comparateur_spatial)	$\subseteq^S, \supset\subset, \cap^S, \cup^S$		

Table 3.2. - Opérateurs thématiques

La sémantique des opérateurs thématiques est définie par les caractéristiques suivantes:

Adjacence ($\supset\subset$) Intersection spatiale des frontières de deux représentations cartographiques.

Buffer (¥) Extension spatiale d'une représentation cartographique exprimée à partir d'un critère de distance (Critère).

Union spatiale (\cup^S) Union spatiale de deux représentations cartographiques.

Intersection spatiale (\cap^S) Intersection spatiale de deux représentations cartographiques.

Inclusion spatiale (\subseteq^S) Représentation cartographique (premier opérande de l'opérateur) si celle-ci est incluse dans une représentation cartographique (deuxième opérande de l'opérateur).

Vol d'oiseau (•--•) : Segment de droite de plus courte distance qui relie deux représentations cartographiques.

Début d'un chemin (•-->) Représentation cartographique de l'origine d'un chemin.

Fin d'un chemin (-->•) Représentation cartographique de la destination d'un chemin.

Ces opérateurs thématiques constituent un ensemble de base manipulable par un utilisateur de SIG dans la formulation de ces requêtes. Ils constituent des outils d'interrogation de haut-niveau qui s'appliquent sur les différents types de représentations cartographiques. La définition de cet ensemble de base est d'autant plus nécessaire que ces opérateurs sont souvent manipulés dans des processus cognitifs de description d'entités dans un espace ou de caractérisation de relations spatiales. Les signatures des opérateurs thématiques sont les suivantes :

$\supset\subset$	Rel x Rel	-->	Rel
¥	Rel x Critère	-->	Rel

$\cup S$	Rel x Rel	-->	Rel
$\supset S$	Rel x Rel	-->	Rel
$\sqcup S$	Rel x Rel	-->	Rel
$\bullet \dashv \bullet$	Rel x Rel	-->	Rel
$\bullet \dashv \rightarrow$	Rel _n x Rel _e	-->	Rel
$\dashv \rightarrow \bullet$	Rel _n x Rel _e	-->	Rel

Les propriétés des opérateurs thématiques sont précisées par la Table 3.3.

Opérateurs them.	Unaire	Binaire	Commutatif
$\supset \subset$		X	X
\nexists	X		
$\cup S$		X	X
$\supset S$		X	X
$\sqcup S$		X	
$\bullet \dashv \bullet$		X	X
$\bullet \dashv \rightarrow$	X		
$\dashv \rightarrow \bullet$	X		

Table 3.3. - Propriétés des opérateurs thématiques

3.1.3. Opérateurs réseaux

Un opérateur réseau est un opérateur spatial, défini à partir de graphes représentés par des relations représentant des noeuds (Rel_n) et des arcs (Rel_e), dont le résultat est un réseau défini par une relation représentant des noeuds (Rel_n) et par une relation représentant des arcs (Rel_e). Les opérateurs réseaux sont fermés sur l'ensemble des graphes.

Les opérateurs réseaux sont basés sur la manipulation de graphes, de chemins, et de leurs composants. L'opérateur de chemin délivre l'ensemble des chemins possibles entre une origine et une destination à partir de l'identification de l'ensemble des chemins possibles à partir de cette origine. Cet opérateur peut être affiné par la définition de restrictions dans l'expression de la requête [Mainguenaud 1995]. Cette notion d'ensemble de chemins traduit l'expression d'une recherche de parcours dans un contexte orienté utilisateur⁹.

Les symboles utilisés pour désigner les opérateurs réseaux sont présentés par la Table 3.4.

Chemins:	$-->$		
Inclusion noeuds:	\subseteq^n	Intersection noeuds:	\cap^n
Intersection chemins:	$\cap-->$	Inclusion chemins:	$\subseteq-->$

Table 3.4. - Opérateurs réseaux

Les opérateurs réseaux sont des opérateurs logiques. La sémantique du résultat est donnée par la formulation logique de l'opérateur, la spatialité du résultat d'un opérateur réseau est une propriété du résultat au même titre que les propriétés alphanumériques.

La sémantique logique associée aux opérateurs réseaux est définie par les caractéristiques suivantes :

Chemins ($-->$) Ensemble de chemins composé à partir d'origines et de destinations sur un graphe, d'un critère réseau définissant une sélection sur les noeuds et sur les arcs de ce graphe (Critère_réseau), et d'une contrainte à vérifier lors de l'exécution de cet opérateur sur ce graphe (Contrainte_réseau).

Inclusion noeuds (\subseteq^n) Ensemble de noeuds s'ils sont inclus dans un ensemble de chemins.

Intersection noeuds (\cap^n) : Ensemble de noeuds communs à deux ensembles de chemins.

Inclusion de chemins ($\subseteq-->$) Ensemble de chemins s'ils sont contenus dans un deuxième ensemble de chemins.

⁹ La notion de plus court chemin proposée par la plupart des SIG exprime a priori un critère de choix de parcours sans fournir l'utilisateur d'arbitrage sur le résultat de la requête.

Intersection Sous-ensemble de chemins communs à deux ensembles de chemins ($\cap^{-->}$) chemins.

Les signatures des opérateurs réseaux sont les suivantes :

$-->$	$Rel_n \times Rel_n \times Rel_n \times Rel_e$	
	Critère_réseau x Contrainte_réseau	$--> Rel_n \times Rel_e$
\sqcup^n	$Rel_n \times Rel_e \times Rel_n \times Rel_e$	$--> Rel_n \times Rel_e$
\cap^n	$Rel_n \times Rel_e \times Rel_n \times Rel_e$	$--> Rel_n \times Rel_e$
$\sqcup^{-->}$	$Rel_n \times Rel_e \times Rel_n \times Rel_e$	$--> Rel_n \times Rel_e$
$\cap^{-->}$	$Rel_n \times Rel_e \times Rel_n \times Rel_e$	$--> Rel_n \times Rel_e$

Les propriétés de ces opérateurs réseaux sont précisées par la Table 3.5. (un opérateur est binaire s'il possède deux graphes en argument).

Opérateurs réseaux	Binaire	Commutatif
\rightarrow	X	
\cup^n	X	
\subset^n	X	X
$\bigcup \rightarrow$	X	
$\bigcup \rightarrow$	X	X

Table 3.5. - Propriétés des opérateurs spatiaux réseaux

3.1.4. Opérateurs de manipulation

Les opérateurs de manipulation permettent de réaliser des opérations métriques sur des relations, des opérations arithmétiques et booléennes, des opérations logiques, des fonctions, des opérations d'identification de composants de graphe et des opérateurs d_riv_s. Les opérations d'identification permettent notamment de traduire des composants de graphe sous une forme de relation de base et d'exprimer ainsi des opérateurs thématiques à partir d'éléments de graphe. La sémantique de ces opérateurs est la suivante :

$$X \in \{\text{Rel} \mid \text{Rel}_e \mid \text{Rel}_n\}$$

- métriques

area	Attr	-->	num \cup \perp ¹⁰
perimeter	Attr	-->	num \cup \perp
length	Attr	-->	num \cup \perp

- arithmétiques

+	num \cup \perp x num \cup \perp	-->	num \cup \perp
-	num \cup \perp x num \cup \perp	-->	num \cup \perp
*	num \cup \perp x num \cup \perp	-->	num \cup \perp
/	num \cup \perp x num \cup \perp	-->	num \cup \perp

- booléens

<	num \cup \perp x num \cup \perp	-->	bool
=	num \cup \perp x num \cup \perp str x str bool x bool Rel x Rel Rel _n x Rel _e x Rel _n x Rel _e	-->	bool

- logiques

¹⁰ La valeur indéterminée \perp permet de gérer les résultat de fonctions qui ne seraient pas définies. Une représentation spatiale étant définie par un type abstrait de données, l'existence de la valeur du résultat de l'application d'une fonction sur un TAD représentation spatiale n'est en effet pas systématique (ex: le résultat du calcul d'une surface sur une représentation spatiale de type ponctuel est indéterminé). L'absence d'attribut Représentation_Spatiale dans une relation entraîne obligatoirement la valeur du résultat indéterminé \perp .

and	bool x bool	-->	bool
not	bool	-->	bool
- fonctions			
count	X	-->	num
sum	X x Attr	-->	num \cup \perp
avg	X x Attr	-->	num \cup \perp
min	X x Attr	-->	num \cup \perp
max	X x Attr	-->	num \cup \perp
- identificateurs			
noeuds_réseau	Rel _n x Rel _e	-->	Rel
arcs_réseau	Rel _n x Rel _e	-->	Rel
noeuds&arcs_réseau	Rel _n x Rel _e	-->	Rel
nbre_arcs	Rel _n x Rel _e	-->	Rel

Les opérateurs d'identification ont une sémantique particulière : L'opérateur *noeuds_réseau* délivre une relation de base représentant les noeuds d'un réseau, l'opérateur *arcs_réseau* délivre une relation de base représentant les arcs d'un réseau, l'opérateur *noeuds&arcs_réseau* délivre une relation de base représentant les noeuds et les arcs d'un réseau, et l'opérateur *nbre_arcs* délivre le nombre d'arcs d'un réseau. Ces opérateurs assurent la conversion de relations représentant des graphes en relations thématiques ; ils sont transparents pour l'utilisateur (l'ensemble des opérateurs définis ne constitue pas une interface utilisateur mais un langage d'interrogation du modèle de données). Les opérateurs de base permettent l'expression d'opérateurs dérivés booléens et logiques (\neq , \geq , \leq , $>$, or).

3.1.5. Sémantique des opérateurs spatiaux

Les opérateurs spatiaux thématiques et réseaux identifiés délivrent des relations constituées de nouvelles caractéristiques alphanumériques et spatiales. Ces opérateurs spatiaux présentent en effet la propriété particulière¹¹ de pouvoir créer une nouvelle sémantique spatiale et alphanumérique dérivée des relations op_randes et des op_rateurs utilis_s.

La sémantique spatiale d'un opérateur spatial varie selon la nature des représentations spatiales délivrées [Mainguenaud 1994] :

- L'opérateur crée une relation à représentation spatiale fictive, ces opérateurs sont les opérateurs spatiaux thématiques de vol d'oiseau (•--•) et de buffer (¥);

¹¹ Le modèle relationnel ne permet pas de créer de nouvelle sémantique dans l'application d'opérations, les résultats d'opérations sont basés sur l'utilisation des schémas et des valeurs des relations opérands.

- L'opérateur délivre une relation ou un graphe existant (représenté par des relations) dont les représentations spatiales sont définies. Ce sont les opérateurs thématiques d'inclusion spatiale (\subseteq^S) et les opérateurs réseaux d'inclusion de noeuds ($\subseteq^{N \rightarrow}$), d'inclusion de chemins ($\subseteq^{-->}$), d'intersection de noeuds (\cap^N) et d'intersection de chemins ($\cap^{-->}$);

- L'opérateur délivre une nouvelle relation ou un nouveau graphe (représenté par des relations) à nouvelles représentations spatiales, ce sont les opérateurs thématiques d'intersection spatiale (\cap^S), d'union spatiale (\cup^S), d'adjacence ($\supset\subset$) et les opérateurs réseaux de chemin ($-->$), de début d'un chemin ($\bullet-->$) et de fin d'un chemin ($-->\bullet$).

La création d'une nouvelle relation pose un problème de définition de la sémantique et des valeurs des attributs résultants. Cette sémantique doit être appréciée en fonction des qualifications sémantiques des attributs des relations d'origine (i.e. granule) et de celle des opérateurs utilisés, elle est difficilement généralisable et relève d'une expertise à effectuer au cas par cas. Nous assumons dans la suite de cette étude l'existence d'une procédure de détermination de cette sémantique.

Ces propriétés ont des conséquences sur l'application d'opérateurs classiques au contexte du spatial. La jointure du modèle relationnel n'est, par exemple, pas applicable avec un comparateur spatial dans la mesure où la validité de l'ensemble des attributs résultants n'est pas garantie (la définition de la jointure impose de retrouver l'ensemble des attributs des deux relations opérantes dans la relation résultat). La création d'une nouvelle relation entraîne de façon générale un double problème de définition de la sémantique de cette nouvelle relation et plus globalement de gestion de l'évolution du schéma de la base de données (i.e. la gestion de ces nouvelles relations et leur prise en compte dans le schéma de la base de données).

3.2. Définition d'un ensemble de requêtes exemples

Si les besoins des utilisateurs de SIG en terme d'interrogations spatiales et non spatiales applicables sur une base de données spatiales sont bien identifiés, les langages classiques d'interrogation de bases de données demandent à être étendus par de nouveaux opérateurs et des méthodes de composition d'opérateurs pour prendre en compte la spécificité du spatial (e.g. extensions de SQL) [Boursier 1992]. Ces extensions doivent permettre la manipulation de données spatiales comme des réseaux ou des données thématiques, et permettre la composition d'opérateurs dans la réalisation d'une interrogation spatiale (e.g. sélectionner les parties de réseau qui sont à moins de 200 mètres de bâtis s'intersectant avec une zone inondable c'est à dire la composition d'un opérateur d'extension spatiale avec un opérateur d'intersection spatiale). D'un point de vue global, les requêtes d'un langage d'interrogation sont classifiables en :

- **requêtes élémentaires** constituées à partir d'opérateurs non composés. Ces requêtes se distinguent en (1) interrogations alphanumériques (i.e. relationnelles) traditionnelles (2) interrogations spatiales (3) interrogations réseaux;

- **requêtes composées** construites à partir de requêtes élémentaires. Ces requêtes regroupent des (4) interrogations spatiales thématiques et (5) interrogations supportant la composition d'opérateurs.

Cette section présente un ensemble d'interrogations de référence qui couvrent ces catégories de requêtes et qui sont orientées, à des fins d'illustration, vers l'utilisation de la base de données exemple. Ces requêtes sont décrites par leur forme textuelle et par leur formulation réaliste à partir des opérateurs précédemment identifiés :

(1) Interrogations alphanumériques:

A1: Quels sont les bâtiments affectés à l'enseignement et qui ont une surface utile de plus de 750 m² ?

$$A1 = \sigma (\text{Bâti}, \text{Critère}_{A1})$$

$$\text{Critère}_{A1} = (\text{Affectation} = \text{'Enseignement'} \text{ and } \text{Surface_Utile} \geq 750)$$

(2) Interrogations thématiques:

T1: Quelles sont les parties spatiales de bâti qui s'intersectent avec les zones inondables ?

$$T1 = \cap^S (\text{Bâti}, \text{Zone_Inondable})$$

T2: Quelles sont les zones qui sont à moins de 200 mètres de bâtiments ?

$$T2 = \neq (\text{Bâti}, \text{Critère}_{T2})$$

$$\text{Critère}_{T2} = (\text{Longueur} = 200)$$

(3) Interrogation réseaux:

R1: Quels sont les chemins qui permettent d'aller du nœud routier identifié n1 jusqu'au nœud routier identifié n2 ?

$$R1 = \rightarrow (\text{Route}_n, 1\sigma_{R1}, 2\sigma_{R1}, \text{Route}_e, \text{null}, \text{null})$$

$$1\sigma_{R1} = \sigma (\text{Route}_n, \text{Critère}_{1R1})$$

$$\text{Critère}_{1R1} = (\text{Id} = \text{'n1'})$$

$$2\sigma_{R1} = \sigma (\text{Route}_e, \text{Critère}_{2R1})$$

$$\text{Critère}_{2R1} = (\text{Id} = \text{'n2'})$$

Le résultat de cette requête est un graphe représentant un chemin que nous définissons par deux relations Chemin_R1_n et Chemin_R1_e

R2: Quels sont les chemins d'une longueur totale inférieure à 1000 mètres qui permettent d'aller du nœud routier identifié n1 jusqu'au nœud routier identifié n2 en passant exclusivement par des tronçons de route de catégorie principales et non situés en zone inondable ?

$$R2 = \rightarrow (\text{Route}_n, 1\sigma_{R2}, 2\sigma_{R2}, \text{Route}_e, \text{Critère}_{1R2}, \text{Contrainte_Réseau}_{1R2})$$

$1\sigma_{R2} = \sigma(\text{Route}_n, \text{Critère}_{2R2})$ $\text{Critère}_{2R2} = (\underline{\text{Id}} = 'n7')$

$2\sigma_{R2} = \sigma(\text{Route}_n, \text{Critère}_{3R2})$ $\text{Critère}_{3R2} = (\underline{\text{Id}} = 'n2')$

$\text{Critère}_{1R2} = \sigma(\text{Route}_e, \text{Critère}_{4R2})$

$\text{Critère}_{4R2} = (\text{Categorie} = \text{'Principale'} \text{ and not in } \sigma(\text{Route}_e, \text{Critère}_{5R2}))$

$\text{Critère}_{5R2} = (\cap^S(\text{Route}_e, \text{Zone_Inondable}))$

$\text{Contrainte_Réseau}_{1R2} = (\text{Longueur} < 1000)$

Le résultat de cette requête est un graphe représentant un chemin que nous définissons par deux relations Chemin_R2_n et Chemin_R2_e

R3: Quels sont les noeuds communs aux chemins respectivement définis par les relations $(\text{Chemin_R1}_n, \text{Chemin_R1}_e)$ et $(\text{Chemin_R2}_n, \text{Chemin_R2}_e)$?

$R3 = \cap^{n \rightarrow}(\text{Chemin_R1}_n, \text{Chemin_R1}_e, \text{Chemin_R2}_n, \text{Chemin_R2}_e)$

Le résultat de cette requête est un ensemble de noeuds.

(4) Interrogation composée:

La composition d'opérateurs est une propriété essentielle pour l'application de requêtes combinant les critères de sélection. La formulation syntaxique de requêtes composées impose de coordonner la composition des résultats intermédiaires [Mainguenaud 1994]. Ces compositions d'opérateurs permettent de réaliser les exemples de requêtes suivantes :

C1: Quelles sont les parties spatiales de bâtis affectés à l'enseignement, avec une surface utilisée de plus de 750 m² et qui s'intersectent avec la zone inondable de l'année 1965 ?

$$C1 = \cap^S (\sigma (\text{Bâti}, \text{Critère}_{A1}), \sigma (\text{Zone_Inondable}, \text{Crit_re}_{C1}))$$

$$\text{Critère}_{A1} = (\text{Affectation} = \text{'Enseignement'} \text{ and } \text{Surface_Utile} > 750)$$

$$\text{Critère}_{C1} = (\text{Année} = 1965)$$

La formulation de cette requête compose des opérateurs de sélections (σ) avec un opérateur d'intersection spatiale (\cap^S), le premier argument de l'opérateur d'intersection est la requête alphanumérique A1. Le résultat est une relation.

C2: Quels sont les noeuds de route qui sont à moins de 200 mètres du bâti identifié Id=2 ?

$$C2 = \cap^S (\forall (\text{noeuds_réseau}(\text{Route}_n, \text{Route}_e), \text{Critère}_{1C2}), \text{I}\sigma_{C2})$$

$$\text{Critère}_{1C2} = (\text{Longueur} = 200)$$

$$\text{I}\sigma_{C2} = \sigma (\text{Bâti}, \text{Critère}_{2C2})$$

$$\text{Critère}_{2C2} = (\text{Id} = 2)$$

La formulation de cette requête compose successivement un opérateur d'identification de noeuds d'un réseau (noeuds_réseau), un opérateur d'extension spatiale (\forall) et un opérateur d'intersection spatiale (\cap^S). Le résultat est une relation.

C3: Quels sont les tronçons de route de catégorie principale ?

$$C2 = \sigma (\text{arcs_réseau}(\text{Route}_n, \text{Route}_e), \text{Critère}_{C2})$$

$$\text{Critère}_{C2} = (\text{Catégorie} = \text{'Principale'})$$

C4: Quelles sont les tronçons de route qui sont à moins de 200 mètres de bâtis s'intersectant avec une zone inondable ?

$$\cap^S (\forall (\text{arcs_réseau}(\text{Route}_n, \text{Route}_e), \text{Critère}_{C4}), \cap^S (\text{Bâti}, \text{Zone_Inondable}))$$

$$\text{Critère}_{C4} = (\text{Longueur}=200)$$

La formulation de cette requête C4 compose successivement un opérateur d'identification d'arcs d'un réseau (arcs_réseau) avec un opérateur d'extension spatiale (\bowtie) et un opérateur d'intersection spatiale correspondant à la requête exemple T2 (\cap^S). Le critère Critère_{RT1} représente l'extension spatiale autour des b_tis. Le r_sultat est une relation.

C5: Quel est le nombre d'arcs des chemins qui permettent d'aller du noeud routier identifié 'n1' jusqu'au noeud routier identifié 'n2' ?

$C5 = \text{nbre_arcs} \xrightarrow{-->} (Route_n, 1\sigma_{C5}, 2\sigma_{C5}, Route_e, \text{nul}, \text{nul})$

$1\sigma_{C5} = \sigma (Route_n, \text{Critère}_{1C5})$

$\text{Critère}_{1C5} = (\underline{Id} = 'n1')$

$2\sigma_{C5} = \sigma (Route_n, \text{Critère}_{2C5})$

$\text{Critère}_{2C5} = (\underline{Id} = 'n2')$

C6: Quels sont les noeuds du réseau des routes qui sont en zone inondable ?

$C6 = \cap^S (\text{noeuds_réseau}(Route_n, Route_e), \text{Zone_Inondable})$

La formulation de cette requête fait intervenir un opérateur spatial (\cap^S) qui s'applique sur un réseau dont un opérateur extrait les noeuds (noeuds_réseau). Le résultat est une relation.

(5) Interrogation réseau/thématique:

RT1: Quels sont les tronçons en zone inondable parmi les chemins qui permettent d'aller du noeud routier identifié 'n1' jusqu'au noeud routier identifié 'n2' en passant exclusivement par des routes de catégorie principale ?

$$RT1 = \cap^S(\text{arcs_réseau}(\rightarrow (Route_n, 1\sigma_{RT1}, 2\sigma_{RT1}, Route_e, \\ \text{Critère_r_seau}_{RT1}, nul)),$$

Zone_Inondable)

$$1\sigma_{RT1} = \sigma (Route_n, \text{Critère}_{1RT1})$$

$$\text{Critère}_{1RT1} = (\underline{Id} = 'n1')$$

$$2\sigma_{RT1} = \sigma (Route_n, \text{Critère}_{2RT1})$$

$$\text{Critère}_{2RT1} = (\underline{Id} = 'n2')$$

$$\text{Critère_réseau}_{RT1} = (\text{Catégorie} = 'Principale')$$

La formulation de cette requête fait intervenir un opérateur réseau (\rightarrow) et un opérateur d'intersection spatiale (\cap^S). L'opérateur de réseau fait intervenir des critères de sélection ($1\sigma_{RT1}$, $2\sigma_{RT1}$) afin de déterminer l'origine et la destination du graphe. Le critère de sélection appliqué sur le réseau routier ($\text{Crit_re_r_seau}_{RT1}$) représente l'obligation de passer par des routes de catégorie principale. Le résultat de l'opérateur de chemin est un graphe et se traduit donc par une paire (Rel_n , Rel_n). L'application de l'opérateur d'intersection spatiale impose d'appliquer un opérateur de conversion des chemins obtenus en leur équivalent thématique (arcs_réseau). Le résultat est une relation.

3.3. Langages d'interrogation de bases de données spatiales

Un langage d'interrogation de SGBD est caractérisé par un ensemble d'opérateurs manipulables à travers un langage formel ; il constitue le principal outil d'interaction avec une base de données. De nombreuses propositions sont identifiées pour la définition de langages d'interrogation de données spatiales discrètes¹² [Egenhofer 1987, Orenstein 1988, Haas 1991, Lorie 1991, Larue 1993, Kolovson 1993].

Le pouvoir d'expression des langages d'interrogation de données spatiales est influencé par les propriétés des architectures utilisées. En effet, c'est la sémantique du modèle et sa prise en compte au niveau des architectures qui déterminent le potentiel d'expression d'un langage d'interrogation. Les propositions actuelles d'architectures de bases de données spatiales peuvent se classer en deux catégories (Figure 3.6.) :

- Les architectures étendues qui représentent les données spatiales à partir d'une sur-couche logique bâtie sur un SGBD existant et de fichiers indépendants;
- Les architectures intégrées qui représentent les données spatiales au sein du noyau du SGBD.

Figure 3.6. - Architectures de bases de données spatiales

Cette section présente un aperçu des langages d'interrogation de données spatiales actuellement proposés par la recherche et par les solutions logicielles, et ce, à partir des différentes catégories d'architectures de données spatiales. La section 3.3.1. décrit des exemples de langages utilisés pour des architectures étendues et la section 3.3.2. pour des architectures intégrées. La section 3.3.3. aborde la définition de propriétés minimales dans la définition de langages d'interrogation.

3.3.1. Langages d'interrogation pour architectures étendues

Le principe de base des architectures étendues est l'association de données descriptives représentées dans des tables relationnelles des données spatiales structurées dans des fichiers plus ou moins indépendants selon les SIG. Ces solutions développent des fonctionnalités en sur-couche du noyau d'une architecture de type relationnelle, elles proposent un ensemble de constructions logiques et de procédures donnant des outils de manipulation et de traitement des données spatiales. Ces architectures exploitent partiellement les fonctionnalités des SGBD pour des données à caractère spatial. Elles présentent l'avantage, pour les concepteurs de SIG, d'utiliser une technologie mature et diffusée (i.e. les SGBD relationnels) et de réduire par conséquent les coûts de développement [Ooi 1989].

Les architectures étendues sont qualifiées de propriétaires dans la mesure où les solutions retenues sont souvent ad-hoc. C'est le cas des prototypes SAND [Aref 1991], du langage GEOQL [Ooi 1989] et de la majorité des SIG

¹² Dans le domaine des données spatiales continues (i.e. images), citons les langages PSQL [Roussopoulos 1984], PICQUERY [Joseph 1988], ISQL [Assmann 1986] et IIDS [Chang 1988].

actuels comme Arc/Info d'Esri [Morehouse 1989, ESRI 1994], MGE d'Intergraph, System 9 et Argis d'Unisys, Genamap de Genasys et Vision* de SystemHouse [VISION 1995].

Nous illustrons les architectures étendues à travers les exemples des prototypes de recherche SAND et GEOQL, et des SIG Arc/Info et Vision*. Nous analysons, pour chaque cas, la sémantique des opérations proposées par le langage d'interrogation. Nous présentons la sémantique des prédicats et des opérateurs à travers des requêtes significatives de la base de données exemple. Les fonctionnalités sont analysées lorsque le modèle et le langage d'interrogation décrits permettent la manipulation de graphes.

Le prototype SAND

L'architecture du prototype SAND (Spatial And Non-spatial Data), développé par l'université du Maryland, est basée sur une séparation de la représentation des données spatiales et des données alphanumériques (avec spécialisation des index physiques associés) (Aref 1991). Dans l'architecture logique proposée, les attributs spatiaux et non spatiaux sont liés par des chaînes bi-directionnelles. Afin de contourner le problème de séparation des données spatiales et alphanumériques, des identifiants uniques sont définis pour l'ensemble des entités. Le modèle spatial supporte les types spatiaux suivants : point, segment de ligne et région (i.e. une région désigne un polygone). Les opérations spatiales disponibles sont des prédicats spatiaux de sélection et des fonctions spatiales (surface, périmètre et centroïde d'une région). Les opérations spatiales proposées sont intégrées dans une grammaire proche d'un langage de type SQL. La sémantique et les signatures des prédicats spatiaux disponibles sont les suivantes :

- p représente un type point
- l représente un type ligne
- r représente un type région
- s représente un type point, ligne ou région
- c représente un cercle
- w représente une fenêtre spatiale
- a et n représentent des valeurs numériques

<u>prédicat</u>	<u>signature</u>	<u>sémantique</u>
in_window	s x w --> bool	vrai si s est à l'intérieur de la fenêtre spatiale w
in_circle	s x c --> bool	vrai si s est à l'intérieur du cercle c
nearest_to	s x p --> bool	vrai si s est l'objet spatial le plus proche du point p
objects_at	s x p --> bool	vrai si s est localisée au point p
area	s x a --> bool	vrai si la surface de s est plus grande que la valeur a
pass-through	l x s --> bool r x s	vrai si l (resp r) croise s

adjacent_to	r x r	--> bool	vrai si r est adjacent à r
contained	s x s	--> bool	vrai si s est contenu dans s
within n	s x s x n	--> bool	vrai si s est a moins d'une distance n de s
intersect	s x s	--> bool	vrai si s croise s

Les requêtes exemples basées sur des opérateurs spatiaux, ne sont pas réalisables avec l'algèbre de SAND. L'exécution de la requête exemple T1 sous une forme de prédicat donne la formulation suivante (les attributs *Bâti.Location* et *Zone_Inondable.Location* désignent respectivement les représentations cartographiques des relations *Bâti* et *Zone_Inondable*). Le résultat délivre les identifiant des bâtis qui s'intersectent avec des zone inondables.

```
T1:      T1= T/F $\cap$ S (Bâti, Zone_Inondable)
        T1= Select  Bâti.Id
          From    Bâti, Zone_Inondable
          Where   intersect(Bâti.Location, Zone_Inondable.Location)
```

La définition de ces prédicats spatiaux manque d'homogénéité dans les arguments supportés: Les signatures sont définies de façon variable sur des types spatiaux points, régions ou lignes. Cette hétérogénéité dans la définition des signatures entraîne des problèmes de consistence (ex: le prédicat de contenance *contained* n'est pas valable pour un argument contenant de type ponctuel alors que la spécification le permet). Le modèle ne supportant pas la représentation de graphes, la réalisation d'interrogations complexes intégrant des opérateurs spatiaux ou la composition d'opérateurs n'est pas réalisable.

Le principe de spécialisation des composantes spatiales et alphanumériques affaiblit la qualité de la représentation et des opérations possibles. Ce principe impose une limite structurelle aux potentialités de traitement de SAND même si des outils de gestion et d'interrogation spécialisés sont développés pour contourner le problème de fragmentation (ex: la destruction d'une entité impose de coordonner la destruction des données spatiales et non spatiales de cette entité). Notons que la portée limitée de la sémantique de ces opérations, de par leur définition sous une forme de prédicat, facilite l'exécution des opérations et notamment le calcul des relations topologiques non explicitement représentées dans le modèle.

L'architecture de SAND est adaptée des outils d'interrogation spatiale de première valuation sous une forme de prédicat. En terme d'application, SAND couvre potentiellement le domaine des échelles d'aménagement et de planification. Les traitements réalisés sur ces échelles régionales combinent en effet le plus souvent des données de sources multiples mais sans besoin indispensable de précision géométrique dans les résultats délivrés (i.e. compatible avec une approche des interrogations spatiales par les prédicats).

Le prototype GEOQL

Le langage GEOQL (Geographic Language) est un deuxième exemple de langage utilisant une architecture étendue. L'architecture proposée spécialise les données alphanumériques représentées dans des tables relationnelles et les données spatiales correspondantes dans des fichiers externes [Ooi 1989]. Les types spatiaux

support_s sont le point, la ligne (i.e. ensemble connect_ de segments de droite) et la r_gion (i.e. liste ordonn_e de polygones).

Chaque polygone formant une région est caractérisé par un signe positif (i.e. le polygone forme un ensemble de points _ inclure dans la r_gion) ou par un signe n_gatif (i.e. le polygone forme un ensemble de points _ exclure de la r_gion). Les contraintes spatiales suivantes sont définies pour toute liste ordonnée de polygones formant une région :

- Les polygones d'une succession de polygones de signes positifs doivent être disjoints;
- Tout polygone de signe négatif doit être inclus dans le premier polygone précédent de signe positif.

Nous illustrons ces contraintes à travers les exemples suivants :

- Une propriété parcellaire est décrite par une liste ordonnée de polygones disjoints de signes positifs;
- Une parcelle dans laquelle est incluse un lac est décrite par un polygone de signe positif représentant l'enveloppe externe de cette parcelle suivi d'un polygone de signe négatif représentant l'emprise spatiale du lac.

Le modèle décrit des objets géographiques réels (*geo_obj*) et des objets géographiques réels ou virtuels (*geo_term*). Les objets géographiques virtuels sont le r_sultat d'un op_rateur spatial. Un identifiant unique est utilis_ pour chaque entit_ g_ographique et permet de r_f_rencer les donn_es spatiales _ partir du tuple de la relation. Les relations d_crivant des entit_s spatiales peuvent _tre directement utilisés dans l'expression de requêtes spatiales ce qui permet la réalisation conjointe de requêtes spatiales et alphanumériques.

Le langage de GEOQL est basé sur l'extension d'un langage de type SQL à la définition de prédicats spatiaux et d'opérateurs spatiaux manipulant les primitives spatiales. Deux op_rateurs spatiaux permettent de cr_er des objets virtuels : L_op_rateur *line joining* (i.e. délivre le segment de ligne virtuel de plus courte distance reliant deux objets géographiques de types ponctuels soit un opérateur de vol d'oiseau •--• restreint aux types spatiaux ponctuels) et l'opérateur *bounded by* (i.e. délivre un objet géographique virtuel encadrant deux objets géographiques).

Les opérateurs et prédicats spatiaux définis utilisent les types et valeurs suivants :

geo_obj:	objet géographique (i.e. objet géographique réel)
geo_term:	objet géographique réel ou virtuel (i.e. résultat d'un opérateur <i>line joining</i> ou <i>bounded by</i>)
window_term:	fenêtre spatiale (définie interactivement par l'utilisateur à l'écran ou par la saisie d'une double paire de coordonnées)
n	valeur numérique représentant une distance

Les signatures et les sémantiques des opérateurs et prédicats définis sont les suivantes :

<u>opérateur</u>	<u>signature</u>	
line joining	geo_obj x geo_obj	--> geo_term (d_livre un objet g_ographique virtuel, soit un segment de ligne, reliant deux objets g_ographiques, implicitement de types ponctuels).
bounded by	geo_obj x geo_obj	--> geo_term (délivre l'objet géographique virtuel, soit une région, encadrant deux objets géographiques)
<u>prédicat</u>	<u>signature</u>	
intersects:	window_term x geo_term	--> bool (vrai si la fenêtre spatiale intersecte l'objet géographique réel ou virtuel)
contains:	window_term x geo_term	--> bool (vrai si la fenêtre spatiale contient l'objet géographique réel ou virtuel)
closest:	window_term x geo_term	--> bool (vrai si l'objet géographique réel ou virtuel est le plus proche de la fenêtre spatiale)
furthest:	window_term x geo_term	--> bool (vrai si l'objet géographique réel ou virtuel est le plus éloigné de la fenêtre spatiale)
adjacent:	geo_obj x geo_term	--> bool (vrai si l'objet géographique réel est adjacent à l'objet géographique réel ou virtuel)
joins:	geo_obj x geo_term	--> bool (vrai si l'objet géographique réel et l'objet géographique virtuel ou réel s'intersectent en un point extrémité de l'un des deux objets, les deux op_randes sont implicitement des types lignes)
ends_at:	geo_obj x geo_term	--> bool (vrai si l'objet géographique réel s'intersecte en un point extrémité de l'objet géographique réel ou virtuel implicitement de type ligne)
situated_at:	geo_obj x geo_term	--> bool (vrai si l'objet géographique réel, implicitement de type ponctuel, est situé à l'extrémité de l'objet géographique virtuel ou réel, implicitement de type ligne)
within:	geo_obj x geo_term x n	--> bool (vrai si l'objet géographique réel est à une distance inférieure à n d'un objet géographique virtuel)

Le langage supporte essentiellement des opérations spatiales sous la forme de prédicats, les requêtes exemple basées sur des opérateurs ne sont donc pas formulables. Certains prédicats, qui évaluent des propriétés géométriques, sont de bas-niveau (*joins*, *ends_at*, *situated_at*) et ne sont valides que pour des types spatiaux bien spécifiques. Le langage dispose des prédicats génériques d'interrogation spatiale ($T/F \cap S$, $T/F \subseteq S$ et $T/F \supseteq S$), le prédicat d'extension spatiale n'est pas disponible ($T/F \nsubseteq S$). L'expression de la requête exemple T1 sous une forme de prédicat donne la formulation suivante (à la différence de SAND où le nom de l'attribut spatial est explicité dans la clause Where, le nom seul de la relation est formulée dans la clause Where).

```
T1:      T1= T/F ∩ S (Bâti, Zone_Inondable)
        Select Bâti.Id
        From  Bâti, Zone_Inondable
```

Where Bâti intersects Zone_Inondable

L'opérateur délivrant un objet virtuel permet d'appliquer une requête composée formulée selon l'expression suivante (utilisation de l'opérateur *line joining*):

C2: Quels sont les noeuds de route qui sont à moins de 200 mètres du bâti identifié Id=2 ?)

```
Select    Route_n.Id
From      Route_n, Bâti B1
Where     Route_e within 200 of B1
          B1.Id = 2
```

Le modèle de GEOQL ne supporte pas explicitement la représentation de graphes logiques, les requêtes réseaux ne sont donc pas réalisables par le langage d'interrogation qui ne dispose pas des opérateurs nécessaires. GEOQL intègre un langage d'exécution et des structures spatiales de représentation. Un processeur spatial est développé pour exécuter les composantes spatiales des requêtes. GEOQL propose une stratégie de coordination de l'exécution des opérations non spatiales et spatiales basée sur une décomposition hiérarchique des prédicats à partir de critères spatiaux (i.e. intersection et contenance).

Le SIG Arc/Info

Le SIG Arc/Info est composé d'un ensemble de fonctions et de modules spécialisés, nous distinguons dans cette analyse les fonctions thématiques et les fonctions réseaux.

Fonctions thématiques :

Le modèle spatial du SIG Arc/Info est construit à partir d'un référentiel de base, communément nommé couverture (ou layer), qui définit une géométrie sur une partition de l'espace [Morehouse 1989]. Les données géométriques et topologiques (i.e. basées sur un modèle topologique droite/gauche) sont décrites par des fichiers propriétaires non accessibles aux utilisateurs. Les attributs alphanumériques sont gérés à partir de tables relationnelles pour des primitives cartographiques de types spatiaux point, segment de ligne, polygone et région (i.e. polygone composé)¹³. Des liens bi-directionnels assurent comme pour les architectures de SAND et de GEOQL la mise en relation des données cartographiques et des données alphanumériques (sans identifiants globaux cependant).

Cette séparation des données spatiales et descriptives entraîne une spécialisation des opérations spatiales et alphanumériques :

¹³ Un polygone est décrit par un contour et un centroïde représenté par un point, ce qui exclut la cohabitation de polygones et de points dans une même couverture de données spatiales.

- Les interrogations alphanumériques sont réalisées à partir d'un langage d'interrogation propre¹⁴ qui intègre des opérateurs relationnels, booléens (=, <, >, <=, >=, cn¹⁵, in), logiques (and, not, or) et des opérateurs arithmétiques (*, /, +, -, exp) ;

- Les opérations spatiales font l'objet de traitements indépendants par des programmes spécialisés qui manipulent des couvertures de données pour créer de nouvelles couvertures de données (i.e. fonctions valides pour des ensembles d_entit_s). Les traitements disponibles sont essentiellement bas_s sur des fonctions d_intersection spatiale avec toute une s_rie de variantes possibles utilisant des crit_res bool_ens.

La réalisation d'une requête combinant des critères alphanumériques et spatiaux impose la coordination de deux processus spécialisés. L'exemple de la requête T1 se traduit par une formulation en deux étapes :

T1: T1= \cap^S (Bâti, Zone_Inondable)
 Identity Bâti Zone_Inondable_1965 Bâti_Inondable_1965
 Reselect B,ti_Inondable Poly

La fonction spatiale *Identity* combine deux couvertures d'informations spatiales *Bâti* et *Zone_Inondable_1965* pour créer une nouvelle couverture d'information *Bâti_Inondable_1965* (le modèle de partition exclusif d'Arc/Info impose de définir une couverture d'information pour les zones inondables de 1965 et une deuxième couverture pour les zones inondables de 1985). La sélection alphanumérique est ensuite réalisée sur la nouvelle couverture d'information *Bâti_Inondable_1965* à partir d'un op_rateur de s_lection non standardisé *Reselect*. Nous pouvons remarquer que cette sélection alphanumérique spécifie les types spatiaux recherchés (*Poly* dans cet exemple) alors que la définition de ces types ne relève pas d'un langage d'interrogation orienté-utilisateur. Cette spécification est révélatrice de la philosophie logicielle d'Arc/Info qui est basée sur la manipulation de primitives cartographiques tant au niveau du mod_le que du langage d_interrogation. Un deuxi_me exemple de requ_te C4 illustre cet encha_nement de processus indépendants nécessaire à l'expression d'une interrogation :

C4: C4= \cap^S (\forall (arcs_réseau(Route_n,Route_e), Critère_C4), \cap^S (Bâti,Zone_Inondable))
 Critère_C4 = (Longueur=200)

Identity	Bâti	Zone_Inondable_1965	Bati_Inondable_1965
Buffer	Route	Buffer_Route	200
Identity	Bati_Inondable	Buffer_Route	Resultat_C4
Reselect	Resultat_C4	Poly	

¹⁴ Le SIG Arc/Info est basé sur un SGBD non relationnel, INFO, qui relève d'une architecture logicielle ancienne et qui ne semble plus en développement actif compte-tenu de la faible diffusion de ce système. Les fonctionnalités SGBD d'Arc/Info sont donc souvent limitées. C'est notamment le cas du langage d'interrogation des données alphanumériques qui est développé à partir d'opérations et d'une grammaire non normalisées.

¹⁵ L'opérateur cn teste l'appartenance d'une cha_ne de caract_res _ une cha_ne de caract_re

La couverture cartographique Route représente à la fois les noeuds et les arcs des relations exemples $Route_n$ et $Route_e$. Deux fonctions spatiales sont parallèlement réalisées (*Buffer* et *Identity*) pour générer respectivement une couverture d'information qui représente les bâtis inondés en 1965 (*Bati_Inondable_1965*) et une deuxième couverture d'information qui décrit les extensions spatiales de 200 mètres autour des tronçons de route (*Buffer_Route*). Ces deux nouvelles couvertures sont combinées par une fonction d'intersection (*Identity*), le résultat donne une nouvelle couverture de données (*Résultat_C4*) qui est interrogée à travers l'opérateur de sélection *Reselect* (i.e. sélection des polygones formant le résultat de la requête C4).

Ce deuxième exemple montre la complexité de formulation d'une requête spatiale courante dans le système Arc/Info, le nombre d'opérations nécessaires (formulées de plus dans des modules distincts) et la multiplication rapide des informations intermédiaires nécessaires à la réalisation de la requête (trois nouvelles couvertures d'information pour cette requête C4).

Cette séparation des opérations d'interrogation alphanumérique et des fonctions spatiales pose des problèmes de fragmentation des opérations. Il est notamment impossible d'inclure des critères ou des opérateurs spatiaux dans la réalisation d'une interrogation. La création systématique de nouvelles couvertures d'informations, pour toute opération spatiale, alourdit sérieusement les volumes de données à manipuler. Ces dernières opérations sont également coûteuses en temps de traitement et rendent difficile la répétition interactive des manipulations et la mise à jour de ces couvertures d'informations dérivées.

Fonctions réseaux :

Le modèle réseau d'Arc/Info est basé sur des couvertures cartographiques représentant des segments de lignes orientés (*arc*) et des noeuds associés (*node*). Certains noeuds ont une sémantique particulière pour les fonctions de calcul : les noeuds de passage (*stop*), les noeuds représentant des centres (*center*). Des tables de connectivité sont utilisées pour définir les liaisons de réseau possibles entre les noeuds et les arcs (*turntable*).

Les fonctions principales et leur sémantique sont :

Path: Plus court chemin entre deux noeuds d'un réseau, avec une contrainte pouvant imposer des noeuds de passage sur ce même réseau (*stop*). Le résultat de cette fonction de plus court chemin est défini par le parcours qui offre l'impédance totale minimale. L'impédance de calcul est définie de façon statique par un attribut du réseau (i.e. attribut des segments de lignes et des noeuds).

Tour: Plus court chemin entre deux noeuds d'un réseau avec une contrainte pouvant définir des noeuds de passage ordonnés sur ce même réseau (*stop*). Les noeuds de passage peuvent être pré-regroupés par une fonction de proximité (*spatialorder*) pour optimiser le parcours autour de ces noeuds de passage.

Allocation: Affectation des tronçons de réseau à un noeud de contrôle du réseau selon une impédance cumulée maximale (*supply*).

Trace: Détermine les tronçons d'un réseau connectés à un noeud de contrôle du réseau.

Le modèle réseau d'Arc/Info gère deux directions de parcours possibles sur les arcs. Les tables de gestion des passages sur les noeuds (*turntable*) permettent de définir les directions de parcours possibles (ex: tout droit, à gauche, à droite).

Nous illustrons ces fonctionnalités réseaux à travers la formulation de l'exemple de requête réseau R1.

R1: R1= -->(Route_n, 1σ_{R1}, 2σ_{R1}, Route_e, nul, nul)
 1σ_{R1} = σ (Route_n, Critère_{1R1})
 Critère_{1R1} = (Id = 'n7')
 2σ_{R1} = σ (Route_n, Critère_{2R1})
 Critère_{2R1} = (Id = 'n2')

La première étape de la réalisation de la requête R1 est la définition d'une structure r_seau spécifique et des attributs qui sont utilisés comme impédance. Les deux premières fonctions créent deux attributs d'impédance orientés (*from_to_imp* et *to_from_imp*) sur la couverture des arcs (*Route.aat*) (deux attributs numériques dont les caractéristiques du type sont spécifiés dans la commande: 4 5 n):

```
additem Route.aat Route.aat from_to_imp 4 5 n
additem Route.aat Route.aat to_from_imp 4 5 n
```

Ces nouveaux attributs permettent de définir une impédance de calcul sur chaque arc (ex: la longueur pondérée par la catégorie pour chaque tronçon routier).

Une table relationnelle (*Route.stp*) précise les données concernant le noeud de départ, les noeuds de passage et le noeud d'arrivée soit notamment leurs identifiants (attribut *Route-id*) représentés comme des entiers (n7 et n2 identifiés respectivement par les entiers 7 et 2), l'ordre de passage (*in_order*), et une impédance de passage (*stop_imp*) (Table 3.7.). L'attribut Chemin-Id donne l'identifiant du chemin recherché (valeur 1 dans cet exemple). Un attribut de transfert (*transfert*), non utilisé dans cette requête, permet de sommer un attribut sur les noeuds de passage pendant le parcours.

Table Route.stp	Route-id	in_order	Chemin-Id	stop_imp
	7	1	1	0

	2	3	1	0

**Table 3.7. - Exemple de table de gestion
de parcours**

La recherche de plus court chemin se fait par la séquence de fonctions suivante:

- (1) netcover Route Chemin1
--> fonction (*netcover*) qui définit le réseau utilisé par le calcul (*Route*) et le nom du chemin résultant (*Chemin1*)
- (2) stops Route.stp in_order Chemin_id stop_imp # # cum_imp
--> fonction (*stops*) qui définit le nom de la table des noeuds de passage (*Route.stp*), le nom de l'attribut qui ordonne les noeuds de passage (*in_order*), le nom de l'attribut identifiant du chemin résultant (*Chemin_id*), le nom de l'attribut d'impédance sur les noeuds de passage (*stop_imp*) et le nom de l'attribut résultant (*cum_imp*) qui donnera l'impédance cumulée pour chaque tronçon du chemin résultant.
- (3) impedance from_to_imp to_from_imp
--> fonction qui définit les attributs des impédances (*from_to_imp* et *to_from_imp*) pour les tronçons du réseau
- (4) path stops
--> fonction qui charge en mémoire le réseau et ses paramètres, et exécute la requête.

La gestion de l'affichage du résultat impose le passage au module de cartographie d'Arc/Info (*arcplot*) où la visualisation du chemin résultant est effectuée par une deuxième séquence de fonctions.

- (5) arcplot
---> initialisation du module de cartographie
- (6) mapextent Route
---> définition de la fenêtre spatiale d'affichage (fonction *mapextent*)
- (7) arcs Route
---> affichage des arcs du réseau routier (fonction *arcs*)
- (8) reselect Route nodes Route.stp Route-id
---> sélection des noeuds de passage (fonction *reselect*)
- (9) nodes Route ids
---> affichage des noeuds de passage (fonction *nodes*) et de leurs identifiants
- (10) routelines route chemin1 13
---> affichage des segments de ligne (fonction *routelines*) du chemin résultat avec une couleur codée (13)

La lecture du processus de calcul et d'affichage de ce chemin montre la complexité de la séquence de fonctions nécessaires comme la lourdeur de leur définition (absence de normalisation notamment des fonctions de sélection, codage de la fonction d'affichage pour le choix de la couleur). Le processus est peu interactif (la seule option relativement interactive offerte est le choix de noeuds de passage sur une représentation cartographique affichée à l'écran). Un seul chemin résultant est délivré comme résultat, la définition de critères de sélection sur les tronçons ou sur les noeuds est complexe et doit être réalisée de façon préalable par l'intermédiaire de tables relationnelles (ex: appliquer la recherche du plus court chemin sur les routes de catégorie principale). La sémantique potentiellement représentable par ces tables reste donc limitée. Très peu de fonctions de graphes sont disponibles dans la mesure où le réseau représenté est uniquement cartographique et non logique (ce qui rend impossible la définition de plusieurs niveaux d'abstraction dans la définition du réseau). Toute requête doit être formulée d'un point de vue cartographique et non logique (e.g. afficher un chemin consiste à sélectionner les segments de ligne du réseau routier selon un critère attributif). La réalisation de la requête composée exemple RT1 est partiellement définissable (application sur le plus court chemin et non pas l'ensemble des chemins possibles) à partir d'une coordination des requêtes C4 et R1 que nous venons de décrire :

$$RT1: \quad RT1 = \cap S(\text{arcs_r_seau} \rightarrow (\text{Route}_n, 1\sigma_{RT1}, 2\sigma_{RT1}, \text{Route}_e, \text{Critère_réseau}_{RT1}, \text{null}), \text{Zone_Inondable})$$

$$1\sigma_{RT1} = \sigma(\text{Route}_n, \text{Critère}_{1RT1}) \quad \text{Critère}_{1RT1} = (\text{Id} = 'n1')$$

$$2\sigma_{RT1} = \sigma(\text{Route}_e, \text{Critère}_{2RT1}) \quad \text{Critère}_{2RT1} = (\text{Id} = 'n2')$$

$$\text{Critère_réseau}_{RT1} = (\text{Catégorie} = 'Principale')$$

Reselcover	Route	Chemin	Chemin_Id = 1
Identity	Bâti	Zone_Inondable_1965	Bati_Inondable_1965
Buffer	Chemin	Buffer_Chemin	200
Identity	Bati_Inondable	Buffer_Chemin	Resultat_C3
Reselect	Resultat_C3	Poly	

La formulation de la requête composée RT1 est proche de celle de la requête C4, la couverture Route de la requête C4 est remplacée par une sélection (*Reselcover*) sur cette couverture Route des arcs membre du plus court chemin (Chemin_Id = 1) résultat de la requête R1.

Les interrogations dans le système ArcInfo se réalisent avec des fonctions spatiales et alphanumériques indépendantes. La coordination de ces fonctions relève de la gestion utilisateur, il n'existe pas de forme de langage permettant d'intégrer des opérateurs alphanumériques et spatiaux.

Le SIG Vision*

Dans le modèle de données de Vision* [VISION 1995], les données spatiales sont décomposées et représentées au sein d'un SGBD classique. Les types spatiaux sont organisés en une hiérarchie de généralisation implicite. Des tables relationnelles dédiées décrivent par exemple les identifiants des lignes composant les polygones, et les identifiants des noeuds et des sommets composant les arcs. Un deuxième niveau de tables et de fichiers, non

clipping option: clip to Area

La région de filtre spatial est identifiée par une sous-requête (*clipping area*) et une option de sélection des entités spatiales dans cette région (*clipping option*). Le résultat affiche les parties spatiales de bâtis (*feature*) correspondant au critère de sélection des bâtis et contenues dans la zone inondable de 1965.

La requête exemple C2 se traduit sous le langage d'interrogation de Vision* par la formulation suivante :

```
C2:      C2 =  $\cap^S$  (  $\forall$  (noeuds_reseau( Routen, Routee), Critère1C2), 1 $\sigma$ C2 )
        Crit_re1C2 = (Longueur = 200) 1 $\sigma$ C2 =  $\sigma$  (Bâti, Critère2C2)
        Critère2C2 = (Id = 2)

        Select      feature
        From        Route
        clipping area: Select  feature From Bâti Where_Id = 2
        buffer:      200
        clipping option:      all in
```

La représentation de la route donne (comme dans Arc/Info) un ensemble d'entités représentant à la fois les noeuds et les arcs (*Route*). Un critère de buffer sur l'objet spatial sélectionné est introduit dans cette dernière requête (*buffer*). Les objets spatiaux bâtis résultants sont ceux entièrement contenus dans la région de sélection (*all in*). Un seul objet spatial est utilisable pour définir une région spatiale de sélection. Des requêtes définissant un filtre spatial avec plusieurs régions spatiales ne sont pas réalisables, c'est notamment le cas de l'exemple de requête C4 où plusieurs objets spatiaux sont utilisés par un opérateur de buffer.

Si ce langage permet la définition de variantes d'opérateurs d'intersection spatiale et d'inclusion spatiale, la définition des requêtes se réalise par des séquences de *proc_dures* complexes. Les opérateurs spatiaux disponibles ne sont pas nombreux (ex: extension spatiale uniquement valide sur un objet spatial, pas d'opérateur d'adjacence). Les seuls outils de gestion fournis sont la gestion temporaire (i.e. durant une session) de ces résultats sous une forme d'affichage géographique et d'une table alphanumérique associée. Si une option de sauvegarde de la partie alphanumérique de la requête est fournie, aucune gestion homogène des requêtes (i.e. composantes alphanumériques et spatiales) et de leurs résultats n'est fournie par ce langage. Cette absence de véritable intégration d'opérateurs spatiaux dans les outils d'interrogation reflète la difficulté de développer un langage homogène à partir d'une architecture spécialisant la représentation des données spatiales et non spatiales.

Sur le plan de la gestion des réseaux, Vision* distingue plusieurs formes de réseau logique:

- réseaux de polygones;
- réseaux linéaires où le graphe logique est défini implicitement défini par les relations topologiques;

- réseaux linéaires où le graphe logique est défini explicitement par l'utilisateur.

Des relations de connectivité logique sont supportées par le modèle (i.e. constitution de réseaux non géométriques pour certaines applications de gestion synoptique de réseaux). Le modèle supporte la gestion des directions sur les arcs du réseau. Les fonctionnalités de traitement disponibles relèvent de programmes d'applications réalisés à partir des outils de développement fournis avec le système. Ces applications permettent de développer des fonctions de plus court chemin ou d'identification de parties de réseaux connectés à un noeud du réseau. Le système en l'état ne dispose pas d'outils de définition et de manipulation de réseaux logiques, même si une option de construction d'un réseau défini non implicitement par les relations topologiques est possible. Le modèle de données spatiales de Vision* intègre des mécanismes de version pour les entités (graphes de versions linéaires sans cycle), la gestion de replicas (i.e. plusieurs représentations cartographiques pour une même entité), et un concept de plan schématique qui permet, par extension du concept de replica, de doubler une couverture géographique par une correspondance schématique (utile pour des applications de réseaux combinant une vision géométrique à une vision schématique sans notion d'échelle mais avec une représentation logique). L'architecture de Vision* utilise essentiellement le SGBD Oracle (Ingres ou Sybase peuvent être utilisés mais n'offrent pas le même niveau de service à cause d'un décalage dans la mise à jour des versions de Vision*).

Discussion

L'étude de ces solutions de langage basées sur l'application des principes des architectures étendues permet d'identifier leurs limitations dans la représentation et le traitement des données spatiales. Les données spatiales ne possèdent pas de représentation homogène de leur sémantique. Les relations à caractère spatial sont décrites implicitement par des relations imbriquées hiérarchiquement, ce qui entraîne une représentation complexe non transparente pour l'utilisateur. La sémantique spatiale est uniquement comprise par les programmes du système (une partie des données spatiales est souvent représentée dans des tables d'autres systèmes difficilement accessibles et compréhensibles pour les utilisateurs). La sémantique spatiale du modèle est partiellement reconnue par le SGBD, ce qui limite le pouvoir d'expression des opérateurs. Les interrogations spatiales possibles se limitent donc dans la plupart des cas à l'application de prédicats spatiaux et de quelques opérateurs spatiaux souvent définis de façon incomplète sur le plan de la gestion des résultats. La plupart des langages proposés ne permettent pas la réalisation d'opérateurs réseaux et de composition d'opérateurs. Du fait de la complexité logique du modèle spatial, le modèle comme les prédicats spatiaux sont enfin difficilement extensibles.

Ces architectures présentent l'inconvénient de séparer la gestion des informations spatiales de celle des informations relationnelles, ce qui entraîne des risques évidents d'incohérence. Ce doublement pose des problèmes de fragmentation dans la représentation des données et par conséquent dans les opérations de contrôle d'intégrité. L'ensemble des services rendus par un SGBD n'est pas utilisable pour les données spatiales dans la mesure où leur représentation et leur stockage est en partie exclue de ce même SGBD (ex: gestion des accès concurrents, protection des données spatiales, accès multi-utilisateurs). La définition d'index spatiaux est également difficile à réaliser à partir de la couverture externe. Les relations n'étant pas directement accessibles par les fonctionnalités du langage d'interrogation, le traitement de requêtes sur des volumes croissants de données entraîne des baisses progressives de performance. Si ces architectures continuent à être utilisées par les SIG de la première génération, elles sont cependant en cours d'abandon progressif en raison des limitations énoncées.

3.3.2. Langages d'interrogation pour architectures intégrées

Les architectures intégrées sont basées sur une extension interne du SGBD aux caractéristiques spatiales. Un modèle homogène décrit les données spatiales et non spatiales. Cette intégration, plus complète au sein du noyau du SGBD, apporte l'ensemble des fonctionnalités classiques des SGBD aux données spatiales dans la mesure où la sémantique de l'information spatiale sera comprise par le langage d'interrogation et par le moteur de résolution des opérations. Les règles d'intégrité et de protection des données spatiales sont dans ce cas également et directement prises en charge par le système.

Plusieurs propositions développent des modèles d'architectures intégrées, Probe [Orenstein 1988], GéO2 [Scholl 1992], GEO++ [van Oosterom 1991, Vijbrief 1992], GEOVIEW [Waugh 1987], Gral et Rose [Güting 1989 et 1993], XSQL2 [Lorie 1991], Geo-SAL [Svensson 1991], Starbust [Haas 1991], GéoTropics et GéoSabrina [David 1991, Larue 1993], Papyrus [Kolovson 1993] (Figure 3.8.).

Figure 3.8. - Architectures intégrées [Güting 1989]

Dans le prolongement de notre analyse des modèles spatiaux, nous détaillons successivement les langages proposés par Probe et GRAL, le projet Starbust qui est un exemple d'intégration de données raster et vecteur au sein du modèle et du langage d'interrogation, GéoSabrina qui propose l'extension d'un SGBD relationnel et un langage d'interrogation incluant des opérateurs spatiaux, et GéO2 qui est un exemple de langage défini sur un modèle orienté-objet.

Le modèle Probe

Le modèle Probe [Orenstein 1988] aborde la définition de données spatiales à partir d'un principe qui identifie une représentation spatiale élémentaire : un constructeur de référence **Erreur ! Source du renvoi introuvable.** est défini. Le constructeur **Erreur ! Source du renvoi introuvable.** type *Point_Set* permet de définir à la fois des données discrètes et continues. Probe définit la *région uniforme* comme une zone de l'espace homogène pour une variable donnée :

Une région uniforme est le sous-espace maximal u dans un espace S telle que pour tout point p dans u , $obj(p, S)$ est le même (i.e. l'objet défini en ce point p de l'espace S).

La notion de région uniforme permet d'introduire la définition du résultat de l'opérateur d'intersection spatiale (*overlay*) :

Une opération d'*Overlay(S)* délivre un espace contenant un objet spatial pour chaque région uniforme de l'espace S .

Les fonctions de base applicables (et extensibles) sur des régions uniformes sont les opérateurs spatiaux d'union spatiale (*point-set_union*), d'intersection spatiale (*point-set_intersect*), et de différence spatiale (*point-set_diff*).

Une hiérarchie de généralisation permet de définir des constructions particulières de ce type de base (ex: un type *Parcelle* sera un super-type du type *Feature* qui sera un super-type du type *Point_Set*). Le type de référence *Point_Set* peut être utilisé récursivement dans la définition d'un même type ; une carte (*Layer*) contiendra l'ensemble des points des entités appartenant à cette carte.

Les fonctions ont le caractère particulier de représenter uniformément des propriétés d'entités, des relations entre entités et des opérations sur ces entités. Chaque type d'entité définit les opérations qui lui sont applicables. Ces fonctions combinent des opérations de calcul où les valeurs résultantes sont calculées par des procédures, avec des fonctions basées sur des interrogations.

Dans l'exemple suivant, des déclarations définissent un type d'entité (*Parcelle*) qui possède une fonction décrivant les propriétés de cette entité (*description*) et une fonction définissant une opération sur cette entité (*area*) et un sous-type (*Propriété*) qui possède une fonction propre décrivant les propriétés de cette entité (*statut*) et une relation avec la relation *Parcelle* (*type Propriété is Parcelle*) qui définit l'héritage des fonctions du premier type (*Parcelle*).

```

type Parcelle is Entity
description(Parcelle) --> string
area(Parcelle) --> polygon

type Propriété is Parcelle
statut (Propriété) --> Propriétaire

```

Quelques exemples de prédicats et d'opérateurs sont donnés parmi la liste disponible sous Probe :

Point_feature est une entité spatiale de type ponctuel

Polygon est une entité spatiale de type polygonal

<u>prédicat</u>	<u>signature</u>	<u>sémantique</u>
Overlaps	Polygon x Polygon --> bool	Vrai si les deux polygones s'intersectent
<u>opérateur</u>	<u>signature</u>	<u>sémantique</u>
Overlay	Layer x Layer --> Layer	Délivre une carte intersection de deux cartes
Select	Rel x Prédicat --> Rel	Délivre une nouvelle relation à partir d'une relation et d'un prédicat sur cette relation
Feature	Feature --> {Entity}	Délivre l'ensemble des entités de type Feature
Attribut	Rel x Att --> {value}	Délivre les valeurs de l'attribut (Att) d'une relation
Location	Point_Feature --> (x, y)	Délivre la latitude et la longitude d'une entité ponctuelle
Components	Type --> {Type}	Délivre l'ensemble des types composants un type

Ces opérateurs forment une algèbre élémentaire dont la combinaison permet l'application de requêtes. Un ensemble de fonctions de manipulation proches de l'algèbre relationnelle sont définies [Manola 1986] . L'expression de la requête C1 est formulée par l'expression suivante:

```
C1 = Overlay ( Select (Bâti,  
                    Affectation = 'Enseignement'  
                    and Surface_Utile > 750),  
              Select (Zone_Inondable,  
                    Année = 1965) )
```

Les autres requêtes exemples ne sont pas formulables dans Probe : Le langage dispose essentiellement d'opérateurs spatiaux d'intersection spatiale, le modèle et le langage ne représentent pas les réseaux.

Le constructeur d'ensemble de points à l'avantage de la généralité, il demande à être précisé par des classes d'entités spécialisées en fonction des besoins applicatifs. Ce modèle spatial impose une perte de précision géométrique dans la représentation des données spatiales et rend difficile la cohabitation avec d'autres modèles spatiaux.

L'architecture de Probe est basée sur deux composantes : Le noyau du SGBD avec les processeurs permettant de manipuler les différents types et une collection de classes d'entités. Ce modèle a été en partie implanté avec la définition d'une algèbre, d'une partie des index spatiaux identifiés, des classes d'entités géographiques génériques et une interface graphique.

Le modèle et langage GRAL

Les signatures de l'algèbre de Gral sont définies sur des domaines multiples (many-sorted algebra). Le domaine des signatures couvre un ensemble S de types de base et un ensemble de types dérivés $\{ \Sigma \text{Erreur ! Source du renvoi introuvable.} / w \in S \text{Erreur ! Source du renvoi introuvable. et } s \in S \}$. Cette algèbre est munie d'un ordre partiel \leq , l'algèbre ordonnée résultante est définie par le triplet (S, Σ, \leq) , elle possède les propriétés suivantes :

- Les types de base et les types dérivés sont décrits par une hiérarchie de généralisation qui traduit l'ordre partiel (Figure 3.9.);
- L'algèbre définit des opérateurs sur l'ensemble des types de base et des types dérivés. La signature d'un opérateur est définie par un produit de ces types. Tout type appartenant à ce produit peut être substitué par un de ces sous-types (i.e. héritage implicite).

Figure 3.9. - Hiérarchie des types selon GRAL

Le type area (Area) désigne des polygones non spatialement superposables (partition de l'espace), le type polygone (Pgon) désigne des polygones spatialement superposables.

Le domaine représenté par cette hiérarchie décrit un ensemble de types de base {Real, Str, Ord, Int, Bool, Num, Pt, Line, Region, Pgon, Area, Ext, Geo, Data} et de types dérivés, par exemple {Parcelle, Culture, Propriété_Bâtie}, qui composent les objets de la base (BaseObject) et des objets dérivés (Object) à partir de ces objets de base. L'ensemble des types de base et dérivés se généralisent en un type générique Any. Un type particulier est défini pour représenter les graphes (Graph), le type chemin (Path) est une spécialisation du type Graph.

Les objets dérivés sont définis par l'application d'un opérateur d_union \oplus (délivre une classe d'objets à partir de l'union ensembliste de classes d'objets de la base) et d'un opérateur de produit cartésien \otimes (opérateur de produit cartésien sur des instances d'objets de la base) :

<Type> dénote une classe, Type dénote une instance de cette classe

Prédicat/Opérateur

Signature

\oplus

<BaseObject> x <BaseObject> --> <Object>

\otimes

BaseObject x BaseObject --> Object

Notons que les contraintes et la sémantique de l'opérateur de produit cartésien \otimes ne sont pas complètement définis par l_alg_bre (i.e. validit_ et s_mantique des attributs r_sultants).

Les opérations introduites s'appliquent sur des ensembles ou sur des valeurs de Type Abstrait de Donnée (TAD). Les opérations spatiales identifiées par l'algèbre de GRAL sont (1) des prédicats spatiaux sur les types spatiaux (2) des opérateurs spatiaux sur les types spatiaux (3) des fonctions de manipulation (4) des opérateurs et des fonctions de graphes. Nous précisons la sémantique de l'ensemble de ces opérations :

<u>(1)prédicat</u>	<u>signature</u>		<u>sémantique</u>
intersects	Ext x Ext	--> bool	vrai si les deux objets spatiaux s'intersectent
inside	Geo x Reg	--> bool	vrai si l'objet géographique est entièrement inclus dans la région
outside	Geo x Reg	--> bool	vrai si l'objet géographique est entièrement extérieur à la région
is_neighbour_of	Area x Area	--> bool	vrai si les deux polygones sont voisins (forme de prédicat d'adjacence)
<u>(2)opérateur</u>	<u>signature</u>		<u>sémantique</u>
intersection	{Line} x {Line}	--> {Point}	délivre une relation qui contient
	{Line} x {Reg}	--> {Line}	autant de tuples
	{Pgon} x {Reg}	--> {Pgon}	que d'intersections
overlay	{Area} x {Area}	--> {Area}	délivre une relation qui contient autant de tuples que d'intersections
vertices	{Ext}	--> {Point}	délivre l'ensemble des points composant la frontière d'un objet étendu
vorono	{Point} x {Reg}	--> {Area}	délivre des polygones de vorono à partir d'un ensemble de points et sur un polygone donné (i.e. fenêtre spatiale)
closest	{Point} x Point	--> Point	délivre le points, parmi un ensemble de points, le plus proche d_un point

L'algèbre étant ordonnée, le prédicat inside de premier argument Geo dans sa signature est également valide, par héritage, pour un premier argument de type Point, Ext, Line ou Region. Parallèlement, l'opérateur name est valide pour les types dérivés Culture et Propri_t__B,tie. Des fonctions du langage permettent de réaliser des calculs de distance, de longueur, de périmètre et de surface sur des types géométriques. Des fonctions de manipulation sont d_finies sur la hiérarchie des types identifiée, nous en donnons des exemples:

<u>(3)Fonction</u>	<u>Signature</u>		
seq	Any	-->	Seq(Any)
function	Any	-->	Any Seq(Any) Bool
map	Seq(Any)	-->	Seq(Any)
count	Seq(Any)	-->	Int

the	Seq(Any)	-->	Any
in	Seq(Any) x Any	-->	Bool

La sémantique de ces fonctions est explicitée par les descriptions suivantes :

<u>Fonction</u>	<u>Sémantique</u>
seq	délivre une séquence de types à partir d'un type de la base ou dérivé.
function	fonction sur un attribut de la base ou dérivé qui délivre un type (ou une s_quence d'un type) de la base ou d_riv_, ou un bool_en.
map	délivre une séquence de types à partir d'une séquence de type, il est utilisé pour extraire des sous-types de types (i.e. une séquence de types Geo extraits à partir d'un objet de la base).
count	donne le nombre d'instances d'une séquence
the	délivre un type parmi une séquence
in	délivre une valeur vrai si un type appartient à une séquence

Le modèle de graphe logique est construit à partir de trois ensembles d'objets de la base ($\langle s \rangle$, $\langle t \rangle$, $\langle u \rangle$) qui définissent respectivement les noeuds, les arcs et les chemins du graphe (un chemin est un graphe composé d'une séquence ordonnée de noeuds et d'arcs sans cycle). Formellement un graphe $\{ (N, E, XP, \varepsilon, \Pi) \}$ est défini par :

- (1) $N \subseteq \langle s \rangle, E \subseteq \langle t \rangle, XP \subseteq \langle u \rangle$
- (2) $\varepsilon : E \rightarrow N \times N$ total et injectif (un seul arc entre deux noeuds)
- (3) $\Pi : XP \rightarrow$ **E****Erreur ! Source du renvoi introuvable**.total, son domaine contient uniquement les routes du graphe générées par ε

Les opérateurs définis sur le modèle de graphe sont les suivants :

<u>(4)Opérateur</u>	<u>Signature</u>		
subgraph	Graph x Seq(Node)	-->	Graph
	Graph x Seq(Edge)	-->	Graph
remove	Graph x Seq(Node)	-->	Graph
	Graph x Seq(Edge)	-->	Graph
nodes	Graph	-->	Seq(Node)
from	Edge	-->	Node
to	Edge	-->	Node
path	Graph	-->	Path
shortest_path	Graph x Node x Node		
	x Edge x Contrainte	-->	Path
circle	Graph x Node x Num		
	x Edge x Contrainte	-->	Graph

Le type Contrainte définit l'attribut d'impédance sur les arcs (Edge) pour les opérateurs shortest_path et circle. La variable Num donne la limite maximale de cette contrainte pour l_op_rateur circle.

Ces opérateurs donnent un ensemble de base de manipulation des graphes, deux de ces opérateurs ont une sémantique particulière : `shortest_path` et `circle`. L'opérateur `shortest_path` donne le plus court chemin entre deux noeuds d'un graphe, la fonction de cet opérateur intègre une contrainte sur les arcs (exemple: le nombre d'arcs, la distance géométrique où le temps de parcours). L'opérateur `circle` génère un graphe représentant les parties d'un graphe accessibles à partir d'un noeud de ce graphe et selon une contrainte sur les arcs (exemple: le graphe défini par les parties d'un graphe à moins d'une distance d'un noeud de ce graphe). L'identification de cet ensemble d'opérateurs permet d'étudier leur application sur la base de nos requêtes exemples :

Route = (Route_e , Route_n , ") le type graphe est directement représenté par le modèle

R1:

Route Route_n ('n1') Route_n ('n2') `shortest_path (Routee , length) {Chemin_R1}`

Recherche du plus court chemin sur un critère de distance cumul_e minimale (length), la requête génère un graphe (Chemin_R1)

R2(en partie):

Route Route_e select [Affectation not eq 'Principale'] remove

Route_n ('n1') Route_n ('n2') `shortest_path (Routee length) {Chemin_R1}`

Application d'une requête proche de R1 avec une sélection préalable des arcs du réseau d'Affectation de valeur Principale (remove)

R3:

Select [(Chemin_R1 nodes) intersects (Chemin_R2 nodes)]

Détermine le sous-ensemble des noeuds (intersects) appartenant aux deux chemins Chemin_R1 et Chemin_R2

C1:

(Bâti Select [Affectation='Enseignement' and Surface_Utile > 750]

Zone_Inondable Select [Année = 1965]

Intersection [Bâti, Zone_Inondable, {Crossing}]

Extract [true; Crossing]

Bâti_Inondable)

L'interrogation spatiale exemple C1 est réalisée en deux étapes : l'intersection spatiale entre la relation *Bâti* et la relation *Zone_Inondable* est d'abord réalisée pour les bâtis et les zones inondables correspondant aux critères de sélection. Dans un deuxième temps, le résultat est extrait (opérateur *extract*) et constitue une nouvelle relation *Bâti_Inondable*.

C5:

Route Route_n ('n1') Route_n ('n2') `shortest_path (Routee , length) edge count`

Délivre le nombre d'arcs du plus court chemin délivré par la requête R1 à partir de l'extraction des arcs de ce chemin (edge) et de la somme (count) de leurs occurrences.

C6:

```
select [ ( Route nodes ) intersects (Zone_Inondable) ]
```

Délivre les noeuds du graphe en zone inondable

RT1:

```
Route Route_e select [Affectation not eq 'Principale'] remove
```

```
Route_n ('n1') Route_n ('n2') shortest_path (Route_e length) edge intersects Zone_Inondable
```

Délivre les arcs du chemin délivré par la requête R1 qui s'intersectent avec des zones inondables

Les opérateurs définis sur ce modèle de graphe permettent la réalisation de la plupart des requêtes réseaux exemples. Les restrictions concernent la génération d'un seul chemin résultant pour l'opérateur de plus court chemin et l'absence de prise en compte de crit_res de s_lection sur les arcs ou sur les noeuds pour ce m_me op_rateur.

L'alg_bre ne permet pas en l__tat de r_soudre les requ_tes exemples bas_s sur des op_rateurs d_extension spatiale. Un seul opérateur spatial de base est disponible : l'intersection spatiale (\cap^S , avec les opérateurs *overlay* et *intersection*). Les prédicats disponibles sont l'intersection spatiale ($T/F \cap S$), l'inclusion ($T/F \subset S$) et l'adjacence ($T/F \supset S$).

GRAL présente l'avantage de disposer d'une large gamme de types et d'opérateurs sur ces types, et une extensibilité qui permet d'introduire de nouvelles fonctions. Ces op_rateurs orient_s utilisateurs sont compl_t_s par une alg_bre ex_cutable utilisant des index spatiaux pour les op_rations spatiales. Les opérations de calcul de la géométrie et des relations topologiques sont facilitées par le concept de *Realm*, la géométrie présente par contre l'inconvénient d'être simplifiée. Une partie de ce modèle est implémentée par une algèbre exécutable qui traduit les opérateurs de haut niveau par des opérateurs spécifiques aux différents types spatiaux [Güting 1995]. Les travaux réalisés dans le cadre de GRAL sont prolongés par le projet Rose à travers une algèbre orientée utilisateur [Güting 1993].

Starburst

Le projet Starburst, réalisé par le centre de recherche d'IBM d'Almaden de 1984 à 1992, développe un SGBD Relationnel extensible pouvant répondre aux besoins de nouveaux types d'application (ex: CAO/DAO, SIG, systèmes experts) [Haas 1991]. L'apport principal de Starburst, dont l_extensibilit_ est la caract_ristique principale, est de permettre _l_utilisateur de d_finir de nouveaux types spatiaux et opérateurs. Différents niveaux d'extensibilité complémentaires sont définis par Starburst : le modèle de données, le langage d'interrogation, les techniques d'implantation des opérations et les index spatiaux. Les deux premiers niveaux d'apports sont abordés dans le cadre de cette _tude c_est _dire :

- La réalisation de nouvelles méthodes de définition de tables;
- La définition d'opérations sur les tables et les attributs de ces tables.

Starburst permet de définir des types utilisateurs et des fonctions associées. Ces fonctions sont ensuite utilisables par le langage d'interrogation. Le mod_Le permet l'intégration de différences spatiales par la définition de types utilisateurs pour les données discrètes (ex: segment, point) et pour les données continues (ex: type quadtree) [Haas 1991]. Des opérateurs sont définissables (ex: distance pour des données vectorielles, intersection pour des données quadtree, fonction de conversion vecteur-raster). Cette intégration dans un même modèle de données discrètes et continues (et d'outils de conversion raster-vecteur intégrés au sein du langage d'interrogation comme des fonctions) est une originalité de Starburst¹⁶. Nous décrivons les signatures des opérateurs principaux :

Quadtree_Leaf représente un type quadtree

Line représente un type ligne

<Quadtree> désigne un ensemble de cellules quadtree

<u>Opérateur</u>	<u>Signature</u>
Quad_Leaf_Intersect	Quadtree_Leaf x Quadtree_Leaf --> Quadtree_Leaf
Quad	Line --> <Quadtree>
Quad_Intersect	<Quadtree> x <Quadtree> --> <Quadtree>

Afin de mettre en oeuvre la notion d'objet complexe inspirée des SGBD orientés-objets, plusieurs constructeurs ont été successivement utilisés pendant le projet Starburst et notamment le groupage de tuples [Lorie 1983], la connexion de relations (nested relations) [Dadam 1986], des contraintes d'intégrité référentielles [Lorie 1991], et des vues structurées [Lindsay 1990]. Dans le cas des vues structurées, des objets complexes sont définis intentionnellement par des interrogations. Cette extension est appelée XNF pour eXtended Normal Form. Une interrogation XNF consiste en:

- La définition de tables de références (les noeuds), définis par les interrogations standards;
- La définition de relations entre les tables composantes, définies par des prédicats liant un noeud **Erreur ! Source du renvoi introuvable.** avec 0 ou plus d'éléments et un noeud **Erreur ! Source du renvoi introuvable.** en utilisant la sémantique des jointures externes [Date 1986].

Ces noeuds et relations forment un graphe acyclique avec une racine unique. Ces interrogations peuvent être combinées (pour former de nouvelles tables composites), projetées et restreintes. Ces pseudo-tables sont définies par des expressions dynamiques (i.e. à travers le langage d'interrogation), elles peuvent être combinées avec des fonctions utilisateurs et permettent la réalisation d'interrogations récursives (i.e. la composition d'opérateurs). Une hiérarchie de généralisation permet l'héritage de propriétés concernant les types utilisateurs et les fonctions. Sur le plan des opérateurs, le langage d'interrogation de Starburst combine des opérations sur les structures quadtree et sur les relations. L'expression des requêtes T1 et C1 dans le langage d'interrogation de Starburst est présentée par la Table 3.10.

¹⁶ Ces principes d'intégration sont en partie utilisés par le SIG Spans [Tydac 1993].

<u>T1: en mode quadtree</u>	
Bâti_Quadtree	désigne la structure quadtree de la relation Bâti
Z_I_Quadtree	désigne la structure quadtree de la relation Zone_Inondable pour 1965
Bâti_Leaf	désigne les cellules élémentaires de la structure quadtree de Bâti_Quadtree
Z_I_Leaf	désigne les cellules élémentaires de la structure quadtree de Z_I_Quadtree
Quadtree_leaf_intersect	désigne un opérateur d'overlay sur des structures quadtree
Select	Bâti_Leaf
From	Bâti_Quadtree, Zone_Inondable_Quadtree
where	Quadtree_leaf_intersect (Bâti_Leaf , Zone_Inondable_Leaf)
<u>C1: en mode vectoriel</u>	
Select	Bâti.Id
From	Bâti, Zone_Inondable
	B_Loc as (Select Line_Segments from Location L1
	where L1.type = 'Bâti'
	Z_I_Loc as (Select Line_Segments from Location L2
	where L2.type = 'Zone_Inondable')
	QI as (Quad_Intersect (Bâti_Quadtree as Quad(B_Loc),
	Z_I_Quadtree as Quad(Z_I_Loc))
Where	QI.Overlap > 0
	and Zone_Inondable.Année = 1965
	and Bâti.Affectation = 'Enseignement' and Bâti.Surface_Utile > 750

Table 3.10. - Expression de requête exemple selon Starburst

La requête T1 en mode quadtree retrouve les cellules, parmi les cellules (*Bâti_Leaf*) de la structure quadtree de *Bâti*, qui s'intersectent avec les cellules quadtree de la zone inondable de 1965 (*Zone_Inondable_Leaf*).

La requête C1 en mode vecteur sélectionne dans un premier temps, les segments de ligne (*Line_Segments*) des relations *Bâti* et *Zone_Inondable* pour réaliser ensuite l'intersection (*Quad_Intersect*) en mode quadtree à partir d'une transformation de ces structures vectorielles en mode quadtree (*Quad*). Ces segments de ligne sont décrits dans une relation générique (*Line_Segments*) qui représente les segments de toutes les relations en les distinguant par un attribut identifiant (*Location.Type*). Les tables intermédiaires *B_Loc* et *Z_I* constituent des pseudo-tables au sens de Starburst. La fonction *Quad_Intersect* délivre un résultat numérique supérieur à 0 quand la valeur résultat de l'intersection est non vide (*QI.Overlap* > 0). Le résultat final délivre les identifiants des bâtis qui répondent aux critères de sélection de la requête.

Ce langage d'interrogation a l'avantage de combiner l'exploitation de structures vectorielles et de structures quadtree. Le langage est extensible, ce qui rend possible la formulation des requêtes exemples après définition des opérateurs nécessaires. Les interrogations possibles restent cependant limitées à une variable dans le cas de la représentation quadtree ce qui limite les possibilités de combinaison dans l'expression d'interrogations et impose la création d'autant de structures quadtree que d'attributs dans une relation.

Des programmes externes peuvent être intégrés dans le langage d'interrogation pour notamment réaliser des fonctions de graphe. L'interaction de ces fonctions réseaux avec le corps de la requête n'est cependant pas réalisable : Dans la tentative suivante de réalisation de la requête exemple théorique-réseau RT1, la requête est réalisée en deux phases (Table 3.11.) : Une première étape de calcul du plus court chemin à partir d'une matrice

externe de calcul (i.e. une table *Matrice_arc* décrivant des arcs) et d'une fonction externe de calcul (*Path*) avec création d'une table résultat, une deuxième étape de recherche des parties de chemin en zone inondable.

RT1 (phase 1) =	
Select	SegFromVertex, SegToVertex, SegNum
From	PT as Path(:Begnpath, :Endpath, :Matrice_arc, :Linknum) P1
	T1 (Length, PathNo) as
Select	Route.Length, Route.Id
From	Route
Where	Route.Catégorie = 'Principale'
Where	P1.PathNum = T1.PathNo
-----> création d'une table Chemin de gestion du résultat	
RT1 (phase 2)=	
Select	Chemin.Id
From	Chemin_Loc as (Select Line_Segments from Location L1
	Where L1.Type = 'Chemin'
Z_I_Loc as (Select Line_Segments from Location L2
	Where L2.Type = 'Zone_Inondable')
QI as (Quad_Intersect (Z_I_Quadtree as Quad(Z_I_Loc,	
	Chemin_Quadtree as Quad(Chemin_Loc)))
Where	QI.Overlaps > 0

Table 3.11. - Requête réseau-thématique selon Starburst

La requête RT1 est réalisée à partir de l'application d'un programme externe de calcul des chemins (*Path*) qui se réalise sur les deux noeuds de référence (*Begnpath* et *Endpath*) d'une matrice des arcs (*Matrice_arc*) et des identifiants des arcs (*Liknum*). Le résultat final délivre les arcs du plus court chemin qui s'intersectent avec la zone inondable (valeur de *QI.Overlap* > 0). Starburst est un système générique (i.e. la composante spatiale est intégrée au sein d'un SGBD classique) et extensible qui à l'avantage de présenter des outils de composition d'opérateurs et d'intégration de données raster-vecteur (et des outils de conversion) au sein d'un même langage d'interrogation de données spatiales. Dans l'expression de requêtes, un objet spatial est représenté par l'association de plusieurs tuples et notamment à travers la table de gestion des données vectorielles (*Location*). Ce principe de table générique pour la représentation des données vectorielles pose un problème de fragmentation des données spatiales et alphanumériques, et de performance dans le cadre de larges bases de données spatiales. Une extension également proposée est l'introduction d'un identifiant unique pour chaque entité spatiale représentée par un objet complexe (ce qui permet d'optimiser l'écriture des requêtes) [Lorie 1991]. Des techniques d'optimisation ont *_t_d_finies* pour traduire les requêtes du langage d'interrogation vers des requêtes exécutables par le système.

Le SDBD GéoSabrina

GéoSabrina propose un modèle relationnel étendu par des types spatiaux [Larue 1993]. Il est développé à partir du SGBD relationnel Sabrina [Gardarin 1987]. L'intégration des données spatiales est réalisée par l'introduction d'un type spatial générique *GEOM* défini par un TAD. Ce type supporte les primitives spatiales point, ligne connectée et région (éventuellement avec trou). Un objet géographique sera décrit par un tuple dans une table contenant la fois des attributs alphanumériques et spatiaux (i.e. forme et position). Des contraintes définies sur les types spatiaux. Les types spatiaux et les opérateurs définis par le modèle sont extensibles en fonction des besoins applicatifs (ex: application 2.5D ou 3D). Le modèle de référence est le modèle dit spaghetti. Les relations spatiales sont calculées lors de l'expression de requêtes par le langage d'interrogation (ex: inclusion, intersection, adjacence). GéoSabrina distingue trois classes d'opérations spatiales :

- Des fonctions d'agrégat (i.e. itération d'opérateurs binaires sur tous les membres d'une classe, ex: agrégation d'union géométrique) et des fonctions spatiales (aire, longueur, distance);
- Des prédicats spatiaux (ex: union, intersection, adjacence);
- Des opérateurs spatiaux (intersection géométrique, union géométrique, différence géométrique).

Nous précisons la signature d'exemples significatifs de ces prédicats et opérateurs spatiaux (Att est un attribut spatial, Sum une fonction d'agrégation spatiale):

<u>Fonction</u>	<u>Signature</u>		
Sum	Att	-->	Att
Area	Att	-->	Value
Length	Att	-->	Value
<u>Prédicat</u>	<u>Signature</u>		
Union	Att x Att	-->	Bool
Intersection	Att x Att	-->	Bool
Adjacence	Att x Att	-->	Bool

<u>Opérateur</u>	<u>Signature</u>
Union	Att x Att --> Att
Intersection	Att x Att --> Att
Adjacence	Att x Att --> Att

Le domaine défini par le TAD *GEOM* est fermé sur l'ensemble de ces opérations ce qui donne une relative homogénéité au modèle. Ces opérations spatiales sont définies sur ce TAD *GEOM* qui correspond au type Représentation_Spatiale_type de notre modèle. La sémantique des opérateurs spatiaux est définie sur le TAD *GEOM*. L'expression de la requête composée exemple C1 donne notamment (l'opérateur d'intersection est noté * dans GéoSabrina) :

```

C1:      C1 =  $\cap^S$  ( $\sigma$  (Bâti, CritèreA1),  $\sigma$  (Zone_Inondable, CritèreC1))
          CritèreA1 = (Affectation = 'Enseignement' and Surface_Utile > 750)
          CritèreC1 = (Année = 1965)

          select (Bâti.GEOM * Zone_Inondable.GEOM)
          from   Bâti, Zone_Inondable
          where  Bâti.Affectation = 'Enseignement'
          and    Bâti.Surface_Utile > 750 and Zone_Inondable.Année = 1965

```

L'algèbre de GéoSabrina est extensible (i.e. définition de types spatiaux et opérateurs utilisateurs), ce qui permet de réaliser notamment l'expression de la requête T2 à partir de la définition d'un opérateur d'extension spatiale et de la requête composée C4. Les requêtes exemples réseaux ne sont pas exprimables dans la mesure où le modèle de GéoSabrina ne représente pas les graphes. Une proposition d'extension de l'algèbre relationnelle est formulée pour intégrer ces opérateurs dans le langage d'interrogation (i.e. sélection spatiale, projection spatiale et regroupement spatial). Les opérations de sélection permettent l'évaluation de prédicats spatiaux, la projection, le calcul de fonctions, d'opérateurs spatiaux sur les tuples et le regroupement spatial d'agrégats spatiaux. Ces opérations sont implantées dans le noyau du SGBD, leur sémantique est donc connue par le système. GéoSabrina présente l'originalité de ne pas stocker les relations topologiques au sein de son modèle (discutable en terme d'efficacité dans la gestion des contraintes topologiques). Son algèbre s'accompagne de la définition d'index spatiaux dédiés basés sur l'optimisation des accès et des algorithmes.

Le SGBD orienté-objet GéO2

GéO2 étend le SGBD orienté-objet O2 aux données spatiales [Scholl 1992]. GéO2 est constitué d'une architecture à deux niveaux :

- Le niveau conceptuel définit la sémantique des entités géographiques à travers un modèle Entité-Relation étendu par des mécanismes d'héritage et de propagation. L'héritage regroupe les abstractions de généralisation et de spécialisation. La propagation est un mécanisme qui permet de définir les propriétés d'un type à partir de ces constituants (ex: agrégation d'un attribut à partir d'une fonction portant sur les attributs des composants).

- Le niveau d'implémentation est composée de trois structures de représentations complémentaires (i.e. spaghetti, r_seau planaire et non planaire, topologique).

Le prototype développé [Scholl 1992] est basé sur les principes de modélisation de cartes thématiques [Scholl 1989]. Le modèle spatial distingue les régions élémentaires, définies comme des sous-ensembles d'un espace bi-dimensionnel R^2 , et les cartes thématiques définies comme des ensembles de tuples décrits par des données spatiales et non spatiales. Une région peut être une région élémentaire (i.e. un polygone, un sous-ensemble ouvert de R^2 , une union de parties connectées ou non connectées du plan). Les tuples d'une carte thématique doivent au moins posséder un attribut spatial. Les signatures et la sémantique des opérations définies sur les régions élémentaires sont les suivantes :

Reg est une région élémentaire de R^2

<u>prédicat</u>	<u>signature</u>		<u>sémantique</u>
non-empty	Reg	--> Bool	si Reg n'est pas l'ensemble vide de R^2
<u>opérateur</u>	<u>signature</u>		
Union	Reg x Reg	--> Reg	
Intersection	Reg x Reg	--> Reg	
	{Reg} x Reg	--> {Reg}	
	Reg x {Reg}	--> {Reg}	
	{Reg} x {Reg}	--> {Reg}	
Difference	Reg x Reg	--> Reg	
\oplus	{Reg ₁ , ..., Reg _n }	--> Reg	Union ensembliste de n régions
Set	Reg	--> {Reg}	Délivre un ensemble formé d'une région élémentaire à partir d'une région singleton

La définition des cartes thématiques est formulée à partir de la notation des objets complexes [Abiteboul 1988] étendue par des opérations spatiales. Les opérateurs de base disponibles sur les cartes sont les opérateurs relationnels (projection, sélection, produit cartésien) et une opération de regroupement spatial (*nesting*) de régions ayant une même valeur d'attribut (i.e. équivalent à la notion de partitionnement issue du Group By de SQL). Les signatures des opérations spatiales et leur sémantique sont les suivantes (Map désigne une carte thématique de R^2) :

<u>opérateur</u>	<u>signature</u>		<u>sémantique</u>
Fusion	Map	--> Map	Délivre une carte dont l'attribut spatial de chaque tuple délivré est l'union géométrique de l'ensemble des régions de l'attribut spatial d'origine
$\sigma_{\text{géo}}$	Map x σ	--> Map	Délivre une carte résultant d'une sélection selon un critère spatial
Produit géométrique	Map	--> Map	Délivre une carte dont l'attribut spatial de chaque tuple est l'intersection des deux attributs spatiaux de la carte d'origine (avec une condition d'existence de deux attributs spatiaux sur la carte d'origine)

spatiale

Map x Map --> Map

Délivre une carte résultat d'une jointure de deux cartes, dont les tuples existent si l'intersection des attributs spatiaux des deux cartes d'origine est non vide, et dont la valeur de l'attribut spatial est le résultat de l'opérateur d'intersection spatiale.

Les opérations de superpositions spatiales s'expriment à partir de l'opérateur de jointure spatiale (*spatiale*). Cette algèbre permet l'application des requêtes thématiques et thématiques composées. Le modèle ne supportant pas les graphes, l'application de requêtes réseaux n'est pas réalisable. Nous illustrons ce langage à travers la réalisation de la requête C1 :

$C1 = \cap^S (\sigma(\text{B\^a}ti, \text{Crit\`e}re_{A1}), \sigma(\text{Zone_Inondable}, \text{Crit\`e}re_{C1}))$
Critère_{A1} = (Affectation = 'Enseignement' and Surface_Utile > 750)
Critère_{C1} = (Année = 1965)

spatiale ($\sigma(\text{B\^a}ti, \text{Crit\`e}re_{A1}), \sigma(\text{Zone_Inondable}, \text{Crit\`e}re_{C1})$)
Critère_{A1} = (Affectation = 'Enseignement' and Surface_Utile > 750)
Critère_{C1} = (Année = 1965)

Ce modèle et ces opérateurs servent de base au développement du prototype Géo2. Une carte est définie par une classe (*Map*), une carte est formée par un ensemble d'entités spatiales qui forment une deuxième classe (*GeoTuple*).

```
class Map =  
    [ xmax-map:      integer ;  
      ymax-map:      integer ;  
      ...  
      SetGeoTuple:   {GeoTuple} ]  
  
class Geotuple =  
    [The-description: ListAttributes ;  
     The-geometry:     Geometry ]
```

Le type Geometry est un TAD qui supporte des types spatiaux point, arc et polygone (i.e. représentés par des classes). L'ensemble des types spatiaux élémentaires se généralisent en une super-classe (*Spatial*).

Des méthodes sont associées à ces classes, nous en donnons quelques exemples :

<u>Méthode</u>	<u>Classe</u>	<u>Argument</u>	<u>Résultat</u>
Adjacent	Spatial	Spatial	bool
PointInPolygon	Point	Spatial	bool
Intersection	SetPolygon	SetPolygon	SetPolygon

Ces méthodes sont associées aux classes spatiales correspondantes.

Cette séparation de l'architecture de GÉO2 en un niveau conceptuel et structurel assure une indépendance du modèle (i.e. entre les objectifs de modélisation et d'implantation). GÉO2 bénéficie des avantages des SGBD orientés-objets :

- pas de limitation due à la première forme normale dans la définition des types;
- possibilité de définir des objets géographiques complexes avec des méthodes associées;
- extensibilité du système en terme de définition de classes d'objets géographiques et de méthodes associées.

3.3.3. Discussion

L'analyse de ces démarches de définition de langage d'interrogation de données spatiales permettent d'identifier un ensemble de propriétés minimales et nécessaires, un langage d'interrogation de données spatiales doit intégrer :

- Un modèle support permettant de représenter des données thématiques et réseaux;
- L'intégration au sein du modèle de types spatiaux spécifiques supportant la définition d'objets complexes en incluant les représentations discrètes et continues [Lorie 1991];
- Des prédicats spatiaux et des opérateurs spatiaux sur ces nouveaux types [Güting 1989, Haas 1991];
- La composition d'opérateurs dans l'expression de requêtes;
- Des solutions de gestion des résultats d'interrogation [Mainguenaud 1994].

L'intégration de la sémantique des données spatiales sera plus ou moins complète selon le niveau de définition des objets complexes au sein du noyau du SGBD (i.e. en permettant la définition d'objets complexes) ou d'extension d'un modèle existant [Güting 1989]. Les architectures intégrées facilitent, par la représentation de la sémantique des données spatiales, la définition de fonctions et d'opérations spatiales au sein du langage d'interrogation.

L'ensemble des entités et relations spatiales représentables dans les différents modèles proposés restent le plus souvent limitées à des opérations géométriques qui ne peuvent en l'état représenter toutes les interprétations possibles d'un espace géographique. Le manque de normalisation de ces modèles ne rend pas facile leur utilisation

et leur évolution. La plupart des propositions actuelles sont basées sur des extensions du modèle relationnel qui n'est pas, dans sa version actuelle, toujours complètement approprié aux applications spatiales [Egenhofer 1991b, Laurini 1992] :

- Le langage d'interrogation doit accepter la composition d'opérateurs ce qui entraîne des modifications importantes dans le moteur de résolution requise par rapport aux techniques actuellement utilisées dans les SGBD relationnels classiques (i.e. pas de composition d'opérateurs);
- Si le modèle relationnel permet de combiner des tables par l'application d'opérateurs, l'application d'une opération spatiale (telle que l'intersection de deux types d'entités spatiales) peut entraîner la création d'un nouveau type d'entité spatiale ; dans ce cas la consistance des attributs des entités spatiales d'origine n'est pas garantie [Mainguenaud 1994].

La définition de modèles et langages standardisés pour données spatiales sont des conditions indispensables pour favoriser l'intégration, les échanges et l'utilisation finale des données spatiales. Si aucun standard dans la définition des données spatiales ne semble s'imposer aujourd'hui, diverses tentatives de normalisation des formats de données spatiales tentent de proposer un cadre commun pour la représentation des informations spatiales.

L'initiative OpenGIS, tente notamment de dresser un canevas pour l'intégration et la diffusion des données spatiales. Ce consortium international regroupant des partenaires industriels, académiques et institutionnels tente d'identifier un modèle spatial de référence (Open Geodata Model), des spécifications pour la manipulation de données spatiales (Information Communities Model) et une architecture logique et méthodologique permettant de mettre en oeuvre ces deux premiers niveaux de modèle et notamment les règles d'interopérabilité (Information Communities Model) [OPENGIS 1996]. L'architecture à plusieurs niveaux d'OpenGIS présente l'avantage de permettre une compatibilité avec les différentes normes et formats d'échanges proposées. Cette initiative semble réussir à associer la plupart des éditeurs de SIG, des groupes d'utilisateurs et des laboratoires de recherche ce qui est un facteur important pour la réussite de sa diffusion.

C'est par le consensus autour d'un modèle de représentation et d'un langage d'interrogation de données spatiales dans les bases de données que la situation de relative instabilité actuelle dans le domaine des bases de données spatiales devrait se clarifier. La qualité du modèle qui s'imposera devra être analysée selon la complexité des relations spatiales exprimées et les apports des nouveaux mécanismes implantés pour s'adapter aux opérations spatiales. Il faudra en particulier tenir compte du faible pouvoir d'abstraction des logiciels actuels de SIG ; en effet, le modèle de représentation peut influencer indirectement, dans une certaine mesure, la façon dont l'utilisateur va conceptualiser la base de données ce qui va à son tour influencer le type de problèmes que cet utilisateur tentera de résoudre [Armstrong 1990].

5. EXTENSION DU MODELE EXTERNE AU SPATIAL

La définition d'un concept de vue spatiale passe par l'extension des propriétés des modèles externes au domaine du spatial. Cette approche doit permettre de bénéficier des apports actuels des modèles externes sur le plan de leurs propriétés, comme de leurs solutions de mise en oeuvre. Nous proposons d'identifier dans un premier temps les caractéristiques spécifiques des données spatiales qui pourraient modifier ou enrichir la définition des modèles externes. Cette démarche constitue un préalable à la définition d'un modèle de vue spatiale, de langages de définition et de manipulation des données de la vue spatiale, et à l'illustration de son intérêt à partir d'un contexte applicatif significatif.

5.1. Premiers principes

La qualité d'un modèle externe est étroitement liée à celle du modèle de données support et au pouvoir d'expression des opérateurs utilisés lors de la définition des vues. Nous proposons de définir un modèle applicable au spatial et disposant de la richesse sémantique nécessaire à ce contexte spécifique. L'extension du concept de vue au domaine du spatial impose logiquement de disposer d'un modèle de données spatiales et des outils de traitements associés soit :

- (1) Un modèle de données spatiales et non spatiales de référence;
- (2) Un langage d'interrogation de données spatiales et non spatiales de référence;
- (3) Un langage de définition de la vue spatiale;
- (4) Un langage de gestion et de manipulation de la vue spatiale;
- (5) Des outils de gestion de la présentation des résultats.

Le modèle et le langage d'interrogation des données (1, 2) définissent le support de la construction de la vue spatiale. Les outils de définition et de gestion de la vue spatiale constituent le lieu d'extension logique du concept de vue classique (3, 4, 5). Nous proposons d'identifier, pour chacun de ces éléments, les caractéristiques essentielles devant permettre la définition de la vue spatiale. Les principes généraux de définition de la vue spatiale sont décrits par la figure 5.1.

Figure 5.1. Principes de la vue spatiale

Nous décrirons successivement les composantes relevant du modèle spatial et du langage d'interrogation de données spatiales dans la Section 5.1.1., de la gestion et de la manipulation des résultats dans la Section 5.1.2. et des outils de présentation dans la Section 5.1.3. La Section 5.1.4. développe une synthèse des principes de définition de la vue spatiale et une comparaison de principe avec les solutions actuellement proposées.

5.1.1. Modèle et langage d'interrogation de données spatiales

La prise en compte de plusieurs niveaux d'abstraction est une qualité importante d'un modèle spatial de référence. Elle autorise l'intégration, dans une même base de données, de plusieurs niveaux de perceptions assurant ainsi une plus large validité des opérations de manipulation et de visualisation. Le modèle spatial support de développement de la vue spatiale représente la spatialité des entités spatiales à partir de différents niveaux d'abstraction. Cette définition de plusieurs niveaux de perception de l'information spatiale définit ainsi une forme de représentation multiple. La sémantique spatiale d'une entité spatiale est spécifiée par un Type Abstrait de Données (TAD), nous avons défini dans la Section 2 ce type complexe spatial par un attribut Représentation_Spatiale (C1). La représentation spatiale d'une entité spatiale définit un ensemble de niveaux d'abstraction dont les composantes essentielles sont les suivantes :

- La géométrie de ce niveau d'abstraction, définie par une représentation cartographique décrite par un type spatial abstrait TAD (Représentation_Cartographique_type). Dans le contexte des applications géographiques, un type spatial géométrique couvre généralement un domaine de primitives géométriques de point, noeud, ligne, polygone et polygones complexes. Nous n'abordons pas l'étude et la spécification de ce domaine spatial géométrique largement développé dans les recherches actuelles en bases de données spatiales.
- L'échelle de validité de ce niveau d'abstraction (C2). Cette échelle de validité est définie sur un intervalle fermé qui fixe un cadre de manipulation pour les opérations spatiales d'interrogation et de visualisation (Echelle_Validité). La prise en compte de cette échelle de validité aux niveaux des instances est notamment utile dans le contexte de bases de données géographiques intégrant des données hétérogènes provenant de sources de saisie multiples.
- Les paramètres visuels stables, partie de la sémantique graphique d'une donnée spatiale (C3). Cette information est spécifiée dans notre modèle par un attribut Visualisation. Cette description des caractéristiques graphiques d'une donnée spatiale permet d'augmenter la sémantique et par conséquent l'autonomie des entités de la base par rapport aux applications. Cette information permet d'associer une valeur par défaut à une classe d'entités spatiales (ex: présentation graphique hachuré d'entités spatiales bâties) ou à un sous-ensemble d'entités spatiales d'une classe (ex: présentation graphique en fonction de la surface d'entités spatiales bâties), la modification ou l'affectation de nouveaux paramètres restant toujours possible au niveau de l'interface.

Le modèle de vue spatiale est défini à partir d'un langage d'interrogation des données spatiales et non spatiales de référence identifié dans la Section 3.3 (C4). Si le niveau d'abstraction d'une opération d'interrogation spatiale est connu et/ou souhaité par l'utilisateur, il sera spécifié par un attribut que nous qualifions d'échelle de manipulation (C5). Cette échelle de manipulation est définie sur un intervalle fermé borné par deux échelles. Cette échelle de manipulation est mise en correspondance avec les différentes échelles de validité des primitives géométriques de la représentation spatiale d'une entité spatiale qui participe à la définition d'une requête spatiale. Elle permet de définir les primitives spatiales à utiliser pour une opération d'interrogation. Nous assumons dans le contexte de

cette étude l'existence d'une et une seule primitive spatiale pour une échelle de manipulation donnée, l'existence de plusieurs primitives spatiales pour une même échelle entraîne en effet l'étude de conflits sémantiques et spatiaux qui débordent le cadre de cette recherche.

5.1.2. Gestion des résultats d'interrogation de données spatiales

La construction d'une ou d'un ensemble d'interrogations spatiales doit permettre la cohabitation sémantique de différentes requêtes spatiales éventuellement réalisées à partir d'espaces multidimensionnels de niveaux d'abstraction complémentaires. Ces espaces multidimensionnels sont mis en relation par des associations sémantiques (C6). Ces associations permettent de relier les différents espaces traduisant les niveaux d'abstraction successifs définissant l'environnement d'un parcours entre l'origine et l'extrémité d'un réseau défini à plusieurs niveaux d'échelles. L'association de ces espaces multidimensionnels ouvre une perspective fonctionnelle qui permet de coordonner des interrogations spatiales définies à des niveaux d'abstraction complémentaires. Cette possibilité, qui répond aux besoins des utilisateurs dans la perception d'un espace géographique, nous amène à identifier des outils de coordination des résultats d'interrogations de données spatiales et non spatiales soit :

- Définition de groupes d'entités spatiales résultat relevant d'une sémantique homogène (C7);
- Définition de l'importance relative de ces groupes (C8);
- Identification d'outils d'interrogation et de manipulation de ces groupes (C9);
- Identification d'outils de dérivation de ces groupes par de nouvelles interrogations (C10);
- Définition d'opérateurs logiques et spatiaux de manipulation de ces groupes (C11).

Ces groupes homogènes d'entités spatiales, résultat d'une interrogation spatiale, constituent des unités sémantiques de référence. Ils deviennent des composantes à part entière de la base de données spatiales (Base de données dérivée à partir des opérateurs de construction de la vue selon la Figure 5.1.). Ils sont potentiellement manipulables par des outils de gestion classique, et modélisables dans un contexte applicatif.

5.1.3. Présentation des résultats d'interrogation de données spatiales

Le contrôle d'une visualisation de résultats d'interrogations spatiales donne aux utilisateurs les moyens de définir des présentations répondant à leurs besoins applicatifs. Répondre à cet objectif impose de compléter les fonctions d'un langage d'interrogation et de gestion des résultats par des outils de gestion de la présentation de ces résultats. Un langage de gestion de la présentation de résultats d'interrogations spatiales doit notamment définir l'environnement géographique nécessaire à la compréhension sémantique de l'ensemble. La définition de ce contexte géographique relève de choix qui appartiennent en dernier ressort à l'utilisateur, les outils à identifier devant constituer une aide à l'expression de ces choix. Nous proposons de définir les outils de présentation de résultats d'interrogation suivants :

- Définition interactive de paramètres visuels par l'application de variables de Bertin ou de variables de mode d'affichage écran (ex: sur-brillance, contraste) (C12);

- Choix de la portée géographique (i.e. fenêtre spatiale de travail) de la présentation écran des résultats (C13);
- Description de la légende décrivant la sémiologie graphique utilisée dans la présentation des résultats (C14);
- Définition de toponymes et de symboles participant à la compréhension de la lecture des résultats (C15).

Cet ensemble d'outils graphiques constitue un cadre de référence de la gestion des résultats d'interrogations spatiales. Ces outils sont connus et définis dans les SIG, nous proposons de les appliquer au contexte de la vue spatiale afin d'enrichir son expression graphique et donc sa sémantique spatiale. Nous n'abordons pas dans cette caractérisation de la présentation de résultats d'interrogation spatiale les paramètres de manipulation de l'interface qui relèvent d'opérations de gestion de l'interface (ex: interrogation par pointé, gestion et arrangement des fenêtres).

5.1.4. Première synthèse

La Table 5.2. présente une analyse comparative indicative des propositions de recherche et de solutions logicielles étudiées en fonction de ces premiers principes de définition d'un concept de vue spatiale. Ce tableau a pour objectif de présenter des tendances générales pour situer les différentes composantes d'un concept de vue spatiale, il est réalisé à partir de notre compréhension des propriétés de ces propositions à partir de publications ou de documentations logicielles.

- non supporté + supporté en partie ++ supporté	GPL	Tioga	Modèle cartes thématiques	EDSS	MapInfo	Vision /Argis	Arc/Info
ADT spatial (C1)	+	++	++	-	-	-	-
Echelle de validité (C2)	-	+	-	-	-	-	-
Paramètres visuels (C3)	-	-	++	+	+	+	+
Prédicats et Opérateurs spatiaux (C4)	+	++	++	+	+	+	+
Echelle de manipulation (C5)	+	++	+	+	+	-	+
Espaces multidim. (C6)	-	++	-	-	-	-	-
Groupes d'entités (C7)	-	++	++	++	++	++	++
Rôle groupes (C8)	+	++	-	-	+	-	-
Manipulation groupes (C9)	-	-	-	-	+	-	-
Gestion groupes (C10)	-	+	-	+	+	-	-
Opérations groupes (C11)	+	+	-	?	+	-	-
Paramètres visuels (C12)	++	-	-	?	++	-	++
Portée géographique (C13)	++	++	++	++	++	++	++
Légende (C14)	++	?	++	+	++	-	+

Toponymes (C15)	++	+	++	++	++	++	++
-----------------	----	---	----	----	----	----	----

Table 5.2. Analyse comparative

Ce tableau comparatif présente les tendances générales suivantes :

- La plupart des modèles spatiaux utilisés donnent une définition incomplète de la sémantique spatiale des entités spatiales. Ces modèles spatiaux ne permettent pas la prise en compte d'une représentation multiple des entités spatiales, des intervalles d'échelle de validité associés.
- Les opérations d'interrogation spatiales et de présentation de résultats sont dissociées sur le plan de leur définition et de leur manipulation. Cette indépendance apporte une relative flexibilité dans la présentation graphique des résultats (i.e. le choix des modes de présentation se fait en partie à la lecture de ces résultats); elle ne permet cependant pas la prise en compte de paramètres visuels stable au niveau du modèle.
- Les outils de gestion des résultats des ces interrogations ne sont pas complètement disponibles; si les solutions orientées objets notamment abordent la gestion des objets dérivés résultant, leur application au contexte spatial reste un problème ouvert.

Le cadre de cette recherche consiste à construire la vue spatiale sur un modèle et un langage spatial de référence. Ce modèle doit supporter la représentation multiple, intégrer les caractéristiques sémantiques des représentations spatiales et les opérations spatiales nécessaires à la manipulation de ces données. La vue spatiale se caractérise en première identification par l'enchaînement d'un processus d'interrogation spatiale avec des opérations de présentation graphique de ces résultats. L'objectif central de cette étude consiste en la formalisation des mécanismes rendus nécessaires par cette extension et l'identification d'outils de gestion et de manipulation. La construction de la vue spatiale est réalisée sur un modèle spatial de référence et avec les outils de manipulation nécessaires (opérateurs d'interrogation et de présentation). Le modèle de la vue spatiale définit la sémantique des différents constituants tant sur le plan des propriétés que des opérations utilisées par sa construction. Un langage de définition et de manipulation constitue l'interface utilisateur de la vue spatiale.

5.2. Définition de la vue spatiale

L'objectif principal de ce modèle est l'identification des extensions logiques de la vue, telle qu'elle est définie par les modèles classiques, nécessaires à la définition d'un concept de vue spatiale. Nous proposons une formalisation _ partir d'un modèle et de langages de définition et de manipulation de données relationnels étendus aux données spatiales. Le modèle relationnel possède une sémantique et un ensemble d'opérateurs suffisant à notre démarche exploratoire. Les mécanismes de définition des vues du modèle relationnel sont de plus bien connus et stabilisés sur le plan de leurs propriétés, ce qui facilite _ la fois la faisabilit_ et la lisibilit_ de cette proposition.

Les principes du modèle de vue spatiale sont introduits dans la section 5.2.1. à partir d'une construction réalisée avec la base de données exemple de cette étude et accompagnée d'un ensemble de requêtes significatives. Le modèle de vue spatiale est défini dans la section 5.2.2. et des exemples de vues spatiales dans la section 5.2.3. La section 5.2.4. développe et illustre les principes de présentation graphique des vues spatiales. Le langage de définition de la vue spatiale est décrit dans la section 5.2.5., le langage de manipulation et la description des op_ateurs dans la section 5.2.6.

5.2.1 Principes

Nous considérons le modèle spatial et le langage d'interrogations spatiales et non spatiales de référence identifiés dans les sections précédentes. Nous proposons dans cette section une construction introductive à la vue spatiale à partir du schéma de données de la base exemple et de l'ensemble des requêtes de référence développées.

Nous identifions le cas fictif de deux utilisateurs notés respectivement U1 et U2 avec les objectifs suivants dans l'utilisation de la base de donn_es exemple :

- L'utilisateur U1 souhaite une représentation dérivée qui lui permette d'identifier les parties spatiales de bâti et des tronçons du réseau routier situées en zone inondable tout en les positionnant dans le contexte géographique environnant;
- L'utilisateur U2 souhaite une représentation dérivée identifiant des chemins optimaux permettant d'aller du noeud routier identifié 'n1' jusqu'au noeud routier identifié 'n2' tout en les situant par rapport aux zones inondables et au contexte géographique environnant.

Une représentation dérivée de la base de données spatiales est définie par une liste de requêtes. Nous spécifions cette traduction par les composantes et les fonctions suivantes:

rep est une représentation dérivée

requête est une requête

R est l'ensemble des représentations dérivées

Req est l'ensemble des listes de requêtes

() représente le constructeur de liste

La traduction d'une représentation dérivée en une liste de requêtes est exprimée par la fonction **T** suivante :

$$T: R \quad \text{-->} \quad Req$$

$$T(rep) = (requête)$$

La réalisation de la représentation dérivée de l'utilisateur U1 est traduite par la liste de requêtes suivantes:

$$T(rep_{U1}) = (S_Bâti_Inondable, S_Route_Inondable, S_Bâti, S_Zone_Inondable, S_Route_e, S_Route_n)$$

S_Bâti_Inondable: Délivre les parties spatiales de bâti qui s'intersectent avec les zones inondables
 $S_Bâti_Inondable = \cap^S (Bâti, Zone_Inondable)$

S_Route_Inondable: Délivre les parties spatiales du réseau routier qui s'intersectent avec les zones inondables
 $S_Route_Inondable = (\cap^S(\text{arcs-réseau}(Route_e), Zone_Inondable)) \cup^S (\cap^S(\text{noeuds-réseau}(Route_n), Zone_Inondable))$

S_Bâti: Délivre les représentations spatiales de bâti
 $S_Bâti = \Pi (Bâti, Représentation_Spatiale)$

S_Zone_Inondable: Délivre les représentations spatiales des zones inondables
 $S_Zone_Inondable = \Pi (Zone_Inondable, Représentation_Spatiale)$

S_Route_e: Délivre les représentations spatiales des tronçons du réseau routier
 $S_Route_e = \Pi (Route_e, Représentation_Spatiale)$

S_Route_n: Délivre les représentations spatiales des noeuds du réseau routier
 $S_Route_n = \Pi (Route_n, Représentation_Spatiale)$

Le résultat des requêtes S_Bâti_Inondable et S_Route_Inondable traduisent l'objectif de l'utilisateur U1 dans la réalisation de sa représentation dérivée. Les requêtes suivantes (S_Bâti, S_Zone_Inondable, S_Route_e, S_Route_n) décrivent le contexte géographique environnant.

Cette démarche de traduction ne relève pas d'un processus complètement déterministe. Si la traduction de l'objectif de cette représentation dérivée exemple ne pose pas d'ambiguïté dans la définition des requêtes objectifs (en dehors de celles propres au pouvoir d'expression du langage d'interrogation et dans la mesure où la requête de l'utilisateur est clairement exprimée), la sélection des requêtes permettant de situer le contexte géographique environnant impose des choix pour des requêtes complexes où le nombre de relations participantes est élevé et/ou les critères de sélection réduisent la population d'origine des données. Cette variabilité augmente notamment lors du choix des requêtes qui permettent de situer le résultat dans le contexte géographique environnant. Ces caractéristiques apparaissent plus nettement dans la réalisation de la représentation dérivée de l'utilisateur U2 :

$\mathcal{T}(rep_U2)$ = (Chemins, Chemins_Constraints, S_Route_Inondable, S_Route_e, S_Route_n, S_Bâti)

Chemins: Délivre les chemins qui permettent d'aller du noeud routier identifié 'n7' jusqu'au noeud routier identifié 'n2'

Chemins = R1 = --> (Route_n, 1σ_{R1}, 2σ_{R1}, Route_e, null, null)

1σ_{R1} = σ (Route_n, Critère_{1R1}) Critère_{1R1} = (Id = 'n7')

2σ_{R1} = σ (Route_n, Critère_{2R1}) Critère_{2R1} = (Id = 'n2')

Le résultat de cette requête traduit une première solution pour l'objectif recherché par l'utilisateur U2 dans la réalisation de sa représentation dérivée.

Chemins_Constraints: Délivre les chemins d'une longueur totale inférieure à 1000 mètres qui permettent d'aller du noeud routier identifié 'n7' jusqu'au noeud routier identifié 'n2' en passant exclusivement par des tronçons de route de catégorie principales et non situés en zone inondable.

Chemins_Constraints = R2

= --> (Route_n, 1σ_{R2}, 2σ_{R2}, Route_e, Critère_{1R2}, Contrainte_Réseau_{1R2})

1σ_{R2} = σ (Route_n, Critère_{2R2}) Critère_{2R2} = (Id = 'n7')

2σ_{R2} = σ (Route_n, Critère_{3R2}) Critère_{3R2} = (Id = 'n2')

Critère_{1R2} = σ (Route_e, Critère_{4R2})

Critère_{4R2} = (Categorie = 'Principale' and not in σ (Route_e, Critère_{5R2}))

Critère_{5R2} = (∩^S (Route_e, Zone_Inondable))

Contrainte_Réseau_{1R2} = (Longueur < 1000)

Le résultat de cette requête traduit une deuxième solution pour l'objectif recherché par l'utilisateur U2 dans la réalisation de sa représentation dérivée.

Le contexte géographique environnant de cette représentation dérivée (*rep_U2*) est exprimé par les requêtes (S_Route_Inondable, S_Route_e, S_Route_n, S_Bâti) déjà utilisées dans la représentation dérivée précédente (*rep_U1*). Une même requête est donc utilisable dans la définition de plusieurs représentations dérivées, leurs rôles respectifs peuvent cependant varier. S_Route_Inondable joue en particulier un rôle différent dans ces deux exemples : elle fait partie de l'objectif principal pour la représentation dérivée de l'utilisateur U1 (*rep_U1*), et du contexte géographique environnant pour la représentation dérivée de l'utilisateur U2 (*rep_U2*).

Les requêtes Chemins et Chemins_Constraints traduisent deux résultats possibles pour répondre aux objectifs de l'utilisateur U2 dans la réalisation de sa représentation dérivée. Les autres requêtes utilisées (S_Route_Inondable, S_Route_e, S_Route_n, S_Bâti) décrivent le contexte géographique environnant. Ce deuxième exemple illustre le relatif degré de liberté offert lors de la traduction des objectifs d'une représentation dérivée en une liste de requêtes. Le choix entre l'ensemble des chemins possibles (requête Chemins) et un sous-ensemble de chemins correspondant à certaines contraintes et critères (requête Chemins_Constraints) n'est pas en effet

explicitement exprimé dans l'objectif recherché par l'utilisateur U2. Cette flexibilit_ est d'autant plus forte que les objectifs poursuivis par la représentation dérivée ne sont pas exprimables de façon univoque.

5.2.2. Modèle de la vue spatiale

Les représentations dérivées correspondent à un niveau de schéma externe de la base de données. La procédure de création d'une représentation dérivée est constituée à partir de l'identification des requêtes composantes et de la prise en compte de leurs rôles respectifs (e.g. objectif de la requ_te ou contexte environnant de la requ_te). Nous ne considérerons pas dans notre recherche les m_canismes de traduction des objectifs d'une repr_sentation d_riv_e en un ensemble de listes de requêtes. La démarche tente d'identifier la définition, la sémantique et les inter-relations de ces requêtes au sein d'une représentation dérivée. Cette approche permet au modèle proposé d'être indépendant du langage d'interrogation, les principes exposés restent valables pour d'autres modèles et langages de proches valeurs sémantiques.

Une représentation spatiale dérivée comporte un ensemble de requ_tes composantes. Nous qualifions chaque requ_te membre d'une repr_sentation spatiale d_riv_e d'**atome de vue spatiale**. Le modèle proposé est exprimé à partir de la notation des objets complexes [Adiba 1987]. Cette notation étend le pouvoir d'expression des types classiques par un ensemble de constructeurs complexes. Les constructeurs utilisés sont l'agrégation, l'ensemble, la liste et l'énuméré (respectivement notés [], { }, () et \diamond). Un atome de vue spatiale est défini par un type complexe (SVA_type) :

```
SVA_type = [Name: string ,
            Collection:      {string},
            Spatial_Query:   (string) ]
```

Le type décrivant l'atome de vue spatiale est composé du nom de cet atome (Name), de l'ensemble des noms des relations utilisées par cette requête (Collection), et par l'expression logique des requêtes définissant cet atome de vue spatiale (Spatial_Query).

Une collection désigne une relation qui participe à une requête définissant un atome de vue spatiale. Nous illustrons la définition de l'atome de vue spatiale _ partir de la requ_te exemple Chemins_Constraints (R2). L'atome de vue spatiale de la requ_te (R2) est d_fini par l'expression suivante :

```
SVA_type = [Name:          'Chemins_Constraints' ,
            Collection:     {'Route_n','Route_e'},
            Spatial_Query:  ('->(Route_n,1σR2,2σR2,Critère1R2, Contrainte_Réseau1R2')) ]
```

$$1\sigma_{R2} = \sigma(\text{Route}_n, \text{Critère}_{2R2}) \text{Critère}_{2R2} = (\underline{\text{Id}} = 'n7')$$

$$2\sigma_{R2} = \sigma(\text{Route}_n, \text{Critère}_{3R2}) \text{Critère}_{3R2} = (\underline{\text{Id}} = 'n2')$$

$$\text{Critère}_{1R2} = \sigma(\text{Route}_e, \text{Critère}_{4R2})$$

$$\text{Critère}_{4R2} = (\text{Categorie}='Principale' \text{ and not in } \sigma(\text{Route}_e, \text{Crit_re}_{5R2}))$$

$$\text{Critère}_{5R2} = (\cap^S(\text{Route}_e, \text{Zone_Inondable}))$$

Contrainte_Réseau_{1R2} = (Longueur < 1000)

Chaque atome de vue spatiale est, par définition, membre d'au moins une représentation dérivée. L'ensemble des atomes de vue spatiale est un sous-ensemble de l'ensemble des requêtes.

Chaque résultat de requête génère un groupe d'entités relevant d'une sémantique homogène. Le domaine résultat de définition d'un atome de vue spatiale est couvert par l'ensemble des relations dérivées ou non dérivées constituant le r_sultat de la requ_te. Une relation dérivée est composée d'un ensemble d'entités dérivées auxquelles correspondent de 1 à n entités de référence des collections de cet atome (i.e. les entités utilisées par le processus de réalisation de l'atome de vue spatiale).

Dans la réalisation d'une représentation dérivée, nous distinguons trois niveaux d'importance relative pour les atomes de vue spatiale (présentés par un ordre d'importance décroissant) :

- Les atomes de vue spatiale qui traduisent la ou les requêtes objectifs de la représentation d_riv_e (Objective);
- Les atomes de vue spatiale qui décrivent les relations composantes de la ou des requêtes objectifs de la représentation dérivée (Component);
- Les atomes de vue spatiale qui donnent un contexte géographique environnant à la ou aux requêtes objectifs de la représentation dérivée (Environment).

Nous spécifions cette notion de rôle par un attribut complexe (Order) qui gère l'importance relative des différents atomes de vues spatiales dans une même représentation d_riv_e. Cet attribut précise les rôles des atomes de vue spatiale dans la représentation d_riv_e, le type de cet attribut (Order_type) est constitué de trois attributs (Objective, Component, Environment) dont les domaines respectifs sont des ensembles d'atomes de vue spatiale représentés par leurs noms ({string})¹⁷:

Order_type =	[Objective:	{string},
	Component:	{string},
	Environment:	{string}]

Une représentation dérivée est définie pour une échelle de manipulation donnée qui traduit le niveau d'abstraction de l'utilisateur lors de la génération de la population des résultats des atomes de vue spatiale [Claramunt 1994, Woodruff 1995]. L'échelle de manipulation d'une représentation dérivée est décrite par un ensemble d'intervalles d'échelle. La prise en compte d'intervalles disjoints est en effet une propriété nécessaire à la manipulation ultérieure des vues spatiales (i.e. utilisation conjointe de vues spatiales définies pour des échelles de manipulations disjointes).

L'échelle de manipulation de la vue spatiale précise les représentations spatiales à utiliser par les requêtes composant la représentation d_riv_e (i.e. par une mise en correspondance de cette échelle de manipulation avec l'échelle de validité des représentations spatiales manipulées par les atomes de vue spatiale). Cette dernière propriété est essentielle pour des atomes de vue spatiale utilisant des collections intégrant des représentations spatiales multiples. L'échelle de manipulation agit comme un filtre qui permet de définir le niveau d'abstraction de la représentation dérivée et de choisir en conséquence les représentations spatiales des entités spatiales. L'échelle de manipulation d'une représentation dérivée est évaluée si au moins l'un des atomes composant est généré. Cette échelle de manipulation est spécifiée par un intervalle correspondant au niveau d'abstraction de la

¹⁷ Les attributs Objective, Component et Environment de l'attribut complexe Order de la vue spatiale sont spécifiés par des types chaînes de caractères (string) dans un souci de lisibilité. Ces attributs décrivent les noms des atomes de vue spatiale de la vue. Une spécification plus complète imposerait de définir ce dernier par le type de l'atome de vue spatiale (SVA_type). Si cette dernière option n'est pas retenue dans la représentation, elle est assumée.

représentation dérivée. L'échelle de manipulation est précisée par un attribut (Manipulation_Scale). Les intervalles utilisés sont centrés sur une échelle de référence (Scale_Reference) et borné par deux échelles (Min_Limit and Max_Limit). Les définitions respectives de ces attributs sont les suivantes :

```

Elt_Manipulation_Scale_type = [      Min_Limit:          float,
                                   Scale_Reference: float,
                                   Max_Limit:          float      ]

Manipulation_Scale_type =          { Elt_Manipulation_Scale_type }

```

A partir de l'introduction de ces composantes, nous pouvons proposer une définition de la vue spatiale. Une **vue spatiale** modélise une représentation dérivée. Une vue spatiale est identifiée par un nom (Name), elle est composée d'un ensemble ordonné d'atomes de vue spatiale, cet ordre est traduit par l'attribut (Order), l'échelle de manipulation de la vue spatiale est définie par l'attribut (Manipulation_Scale). La spécification de la vue spatiale est la suivante:

```

SV_type =      [Name:          string,
                Order:        Order_type,
                Manipulation_Scale: Manipulation_Scale_type ]

```

La gestion des rôles respectifs des atomes de vue spatiale apporte plusieurs avantages dans la définition d'une vue spatiale. La contribution respective de chaque atome de vue spatiale au sein de la vue spatiale est conservée dans la spécification d'une vue spatiale.

La gestion du rôle des atomes de vue spatiale facilite la gestion visuelle de la présentation des résultats. Les entités visualisées sont ordonnables en fonction des rôles définis pour les atomes de vue spatiale, ce qui introduit un paramètre de contrôle des affichages. En effet, les capacités de présentation de l'espace de visualisation ne sont pas extensibles graphiquement. Cet espace de visualisation se sature progressivement avec l'augmentation du nombre des données affichées. Quand l'échelle de manipulation décroît, les éléments linéaires sont surévalués par rapport aux éléments surfaciques, ou encore le nombre de classes représentées diminue, les "remplissages" de la carte augmentent. Les différents groupes d'entités visualisés interagissent cartographiquement et spatialement. La caractérisation des rôles des atomes de vue spatiale constitue une aide à la gestion des processus d'affichage d'une vue spatiale (i.e. l'importance du rôle d'un atome de vue spatiale se répercute sur l'importance relative et le degré de priorité de son affichage).

La notion d'ordre sur les atomes introduit un indice de gestion de la mise à jour des vues spatiales dans le cas de vues matérialisées. La population de la vue spatiale doit être recomposée si des collections appartenant à des atomes de vue spatiale constituant l'objectif de la requête ou composant de la requête sont mises à jour.

5.2.3. Exemples de vue spatiale

Sur la base des exemples des représentations dérivées des utilisateurs U1 et U2, nous présentons une construction de deux exemples de vue spatiale. Les atomes de vue spatiale utilisés par ces représentations dérivées sont spécifiés par les notations suivantes ¹⁸ :

```
[Name:          'S_Bâti_Inondable',
Collection:     { 'Bâti', 'Zone_Inondable' } ,
Spatial_Query: ('∩S (Bâti, Zone_Inondable) ) ]

[Name:          'S_Route_Inondable',
Collection:     { 'Routee', 'Routen', 'Zone_Inondable' } ,
Spatial_Query: ('(∩S (arcs-réseau(Routee), Zone_Inondable) )
                ∪S (∩S (noeuds-réseau(Routen), Zone_Inondable) ) ) ) ]

[Name:          'S_Bâti'
Collection:     { 'Bâti' } ,
Spatial_Query: ('∩ (Bâti, Représentation_Spatiale)') ]

[Name:          'S_Zone_Inondable',
Collection:     { 'Zone_Inondable' } ,
Spatial_Query: ('∩ (Zone_Inondable, Représentation_Spatiale)') ]

[Name:          'S_Routee',
Collection:     { 'Routee' } ,
Spatial_Query: ('∩ (Routee, Représentation_Spatiale)') ]

[Name:          'S_Routen',
Collection:     { 'Routen' } ,
Spatial_Query: ('∩ (Routen, Représentation_Spatiale)') ]

[Name:          'Chemins',
Collection:     { 'Routee', 'Routen' } ,
Spatial_Query: ('-> (Routen, 1σR1, 2σR1, Routee, null , null )') ]

1σR1 = σ (Routen, Critère1R1)           Critère1R1 = (Id = 'n7')
2σR1 = σ (Routen, Critère2R1)           Critère2R1 = (Id = 'n2')
```

```
[Name:          'Chemins_Constraints',
Collection:     { 'Routee', 'Routen', 'Zone_Inondable' } ,
Spatial_Query: ('-> (Routen, 1σR2, 2σR2, Critère1R2, Contrainte_Reseau1R2 )') ]

1σR2 = σ (Routen, Critère2R2)           Critre2R2 = (Id = 'n7')
2σR2 = σ (Routen, Critère3R2)           Critère3R2 = (Id = 'n2')
Critère1R2 = ∏(Routee, 3σR2)
```

¹⁸ Dans un objectif de lisibilité, les notations de l'algèbre relationnelle sont conservées dans l'expression des valeurs du type Spatial_Query. Elles correspondent aux expressions équivalentes du langage support (i.e. grammaire de type SQL_tendu).

$$3\sigma_{R2} = \sigma(\text{Route}_e, \text{Critère}_{4R2})$$

$$\text{Critère}_{1R2} = \sigma(\text{Route}_e, \text{Critère}_{4R2})$$

$$\text{Critère}_{4R2} = (\text{Categorie} = \text{'Principale'} \text{ and not in } \sigma(\text{Route}_e, \text{Critère}_{5R2}))$$

$$\text{Critère}_{5R2} = (\cap^S(\text{Route}_e, \text{Zone_Inondable}))$$

$$\text{Contrainte_Réseau}_{1R2} = (\text{Longueur} < 1000)$$

Nous construisons les vues spatiales des utilisateurs U1 et U2 sur la base de ces atomes de vue spatiale. Pour la représentation dérivée de l'utilisateur U1, nous définissons deux niveaux d'abstraction : Une première vue spatiale pour une échelle de manipulation centrée sur du 1/ 5 000 (U1_Local), une deuxième vue spatiale pour une échelle de manipulation centrée sur du 1/15 000 (U1_Regional) :

```
[Name:          'U1_Local' ,
Order:          [Objective:      {'S_Bâti_Inondable', 'S_Route_Inondable'},
                  Component:     {'S_Zone_Inondable'},
                  Environment:    {'S_Bâti', 'S_Route_e', 'S_Route_n'} ] ,
Manipulation_Scale: [Min_Limit:   1/10 000,
                    Scale_Reference: 1/5 000,
                    Max_Limit: 1/2 500 ] ]
```

```
[Name:          'U1_Regional' ,
Order:          [Objective:      {'S_Bâti_Inondable', 'S_Route_Inondable'}
                  Component:     {'S_Zone_Inondable'},
                  Environment:    {'S_Bâti', 'S_Route_e', 'S_Route_n'} ] ,
Manipulation_Scale: [Min_Limit:   1/20 000,
                    Scale_Reference: 1/15 000,
                    Max_Limit: 1/10 000 ] ]
```

Les atomes de vue spatiale, objectifs de la vue spatiale, sont décrits par des relations spatiales accompagnées des attributs thématiques (i.e. les données thématiques sont inférables à partir des références spatiales). Les atomes de vue spatiale participant aux requêtes objectives de la vue ou son environnement sont décrits plus simplement à partir d'une projection de leurs attributs spatiaux.

Pour la représentation dérivée de l'utilisateur U2, une vue spatiale est définie pour une échelle de manipulation centrée sur du 1/ 5 000 (U2_Local) :

```
[Name:          'U2_Local' ,
Order:          [Objective:      {'Chemins', 'Chemins_Constraints'},
                  Component:     {'S_Route_e', 'S_Route_n'},
                  Environment:    {'S_Route_Inondable', 'S_Bâti'} ] ,
Manipulation_Scale: [Min_Limit:   1/7 500,
                    Scale_Reference: 1/5 000,
                    Max_Limit: 1/2 500 ] ]
```

5.2.4. Présentation des résultats

La présentation de résultats d'interrogations spatiales est généralement réalisée à partir d'un langage de manipulation des différents résultats de requêtes accompagné d'une interface visuelle et interactive (ex: langages et variables visuels présentés sous une forme de métaphores). Nous intégrons les principes de construction de la présentation d'une vue spatiale autour des options de présentation suivantes :

- Définition interactive de paramètres visuels par l'application de variables de Bertin ou de variables de mode d'affichage écran (ex: sur-brillance, contraste);
- Choix de la portée géographique (i.e. fenêtre spatiale de travail) de la présentation des résultats d'interrogations spatiales;
- Possibilité de description de la légende décrivant la sémiologie graphique utilisée dans la présentation des résultats d'interrogations spatiales;
- Possibilité d'ajout de toponymes et de symboles permettant une amélioration de la compréhension des résultats.

Dans la mesure où la vue spatiale devient une composante de la base de données spatiale, les outils de manipulation graphique sont ceux du langage de manipulation et de l'interface graphique de cette base de données.

La Figure 5.3.¹⁹ présente le résultat spatial de ces deux vues spatiales exemples U1_Local et U1_Regional. L'échelle de manipulation est un facteur important dans la recherche des résultats produits par cet exemple compte-tenu de la prise en compte de la représentation multiple dans le modèle spatial. Si la précision géométrique des résultats est différente selon les échelles de manipulation utilisées, le changement de niveau d'abstraction modifie également le nombre et la qualité des instances résultant de l'intersection des instances de bâti (Bâti) et du réseau routier (Route_e) avec les instances de zone inondable (Zone_Inondable).

Figure 5.3. - Exemples de vues spatiales

Dans les deux cas de vues spatiales U1_Local et U1_Regional, les conflits visuels provenant des résultats de S_Bâti_Inondable et S_Bâti d'une part, et de S_Route_Inondable avec S_Route_e et S_Route_n d'autre part (ex: les parties spatiales de Bâti en Zone_Inondable et les parties spatiales de Bâti) sont résolus par les priorités d'affichage données aux requêtes membres de l'attribut Order.Objective (ex: les instances de S_Bâti_Inondable et S_Route_Inondable sont respectivement prioritaires par rapport aux instances de S_Bâti et de S_Route_e). La présentation graphique utilisée est définie pour les atomes de vue spatiale à partir de variables de Bertin, elle

¹⁹ Les graphismes de présentation des figures 5.3. et 5.4. illustrent les deux vues spatiales exemples, leur choix relève de l'interface graphique.

permet de distinguer les instances de bâti et de tronçons du réseau routier en zones inondables; la sémantique de cette présentation est décrite par une légende.

La Figure 5.4. présente la vue spatiale U2_Local, notons la cohabitation dans cette vue spatiale de plusieurs résultats des requêtes objectives (i.e. deux chemins sont proposés pour aller du noeud identifié n7 jusqu'au noeud identifié n2). L'atome de vue spatiale S_Route_Inondable n'apparaît pas visuellement dans la mesure où ses représentations spatiales se superposent spatialement à celles des requêtes objectives de la vue, ces dernières possédant en effet des priorités d'affichage supérieures.

Figure 5.4. - Exemple de vue spatiale

Ces exemples montrent qu'un atome de vue spatiale isolé a une signification sémantique relativement réduite. C'est la structure ordonnée d'atomes de vue spatiale qui contribue à la création de véritables représentations dérivées.

Le concept de vue spatiale contribue à la définition d'un niveau conceptuel de communication entre le besoin d'expression de représentations diverses et leur traduction en terme de requêtes compréhensibles par la base de données spatiales, et un niveau logique de gestion de la contribution de ces requêtes en terme d'expression graphique dans un environnement de présentation ordonné. La notion de vue spatiale propose un mode de communication entre la perception souhaitée par les utilisateurs et les fonctions d'interrogation d'une base de données spatiales. La vue spatiale donne un contexte de modélisation, de gestion et de présentation d'interrogations spatiales. L'outil proposé est indépendant du modèle de données et du langage d'interrogation, il est flexible dans la mesure où il permet l'intégration de multiples requêtes au sein de plusieurs vues spatiales définies avec des objectifs particuliers.

5.2.5. Langage de définition de la vue spatiale

Nous proposons d'identifier les bases d'un langage de définition de la vue spatiale. Ce langage couvre les mêmes fonctionnalités que les langages de définition utilisés par les SGBD classiques. Les opérateurs sont définis sur le domaine des atomes de vue spatiale et des vues spatiales. Nous présentons successivement la sémantique, les signatures et des exemples de ces opérateurs pour les atomes de vue spatiale et pour la vue spatiale.

DB_type représente le type qui modélise une base de données (le nom de la base représenté par une chaîne de caractères)²⁰ :

DB_type = string

Atomes de vue spatiale

Ces opérateurs permettent respectivement de créer un atome de vue spatiale (Create_SVA), de changer le nom d'un atome de vue spatiale (Change_Name_SVA) et d'effacer un atome de vue spatiale (Drop_SVA). L'opérateur (Change_Name_SVA) donne un nouveau nom (SVA_Name_type₂) à un atome de vue spatiale identifié par son actuel nom (SVA_Name_type₁). L'opérateur (Drop_SVA) efface un atome de vue spatiale identifié par son nom (SVA_Name_type) (à distinguer de la sémantique d'un opérateur Delete qui efface généralement les valeurs dans son usage).

SVA_Name_Type représente le nom d'un atome de vue spatiale :

SVA_Name_type = string

<u>Opérateur</u>	<u>Signature</u>	
Create_SVA	DB_type x SVA_type	--> DB_type
Change_Name_SVA	DB_type x SVA_Name_type ₁	
	x SVA_Name_type ₂	--> DB_type
Drop_SVA	DB_type x SVA_Name_type	--> DB_type

Nous illustrons ces opérateurs avec l'atome de vue spatiale exemple S_Bati_Inondable. Les opérations suivantes permettent respectivement de créer cet atome, de changer son nom et de l'effacer.

```
Create_SVA ( [ 'Base_Exemple', 'S_Bâti_Inondable',  
              { 'Bâti', 'Zone_Inondable' }, ( '∩S (Bâti, Zone_Inondable) ' ) ] )
```

```
Change_Name_SVA ('Base_Exemple', 'S_Bâti_Inondable', 'S_Bâti_Risque' )
```

```
Drop_SVA ('Base_Exemple', 'S_Bâti_Risque' )
```

Vue spatiale

²⁰ DB_type représente une base de données spatiales par son nom dans un souci de simplification de la notation.

Un premier sous-ensemble d'opérateurs permet de créer une vue spatiale (Create_SV), de changer le nom d'une vue spatiale (Change_Name_SV) et de supprimer une vue spatiale (Drop_SV). L'opérateur (Change_Name_SV) donne un nouveau nom (SV_Name_type₂) à une vue spatiale identifiée par son actuel nom (SV_Name_type₁). L'opérateur (Drop_SV) détruit une vue spatiale identifiée par son nom (SV_Name_type):

SV_Name_type représente le nom d'une vue spatiale :

SV_Name_type = string

<u>Opérateur</u>	<u>Signature</u>
Create_SV	DB_type x SV_type --> DB_type
Change_Name_SV	DB_type x SV_Name_type ₂ x SV_Name_type ₁ --> DB_type
Drop_SV	DB_type x SV_Name_type --> DB_type

Nous illustrons ces opérateurs avec la vue spatiale exemple U2_Local. Les opérations suivantes permettent respectivement de cr_er cette vue, de changer son nom et de la supprimer :

```
Create_SV ( 'Base_Exemple',
           [ 'U2_Local',
             {'Chemins','Chemins_Constraints'},
             {'S_Route_Inondable', 'S_Route_e', 'S_Route_n'},
             {'S_Bâti'},
             [ 1/10 000, 1/5 000, 1/2 500 ] ] )
```

```
Change_Name_SV ('Base_Exemple', 'U2_Local', 'U2')
```

```
Drop_SV ('Base_Exemple', 'U2')
```

Atomes et vue spatiale

L'opérateur (Add_SVA_to_SV) permet d'ajouter un atome de vue spatiale (SVA_type) au sein d'une vue spatiale identifiée par un nom (SV_Name_type) :

Le rôle d'un atome de vue spatiale est précisé par le type SV_Role_type

SV_Role_type = <'Objective', 'Component', 'Environment'>

<u>Opérateur</u>	<u>Signature</u>
Add_SVA_to_SV	DB_type x SV_Name_type x SV_Role_type x SVA_type --> DB_type

Nous illustrons ces opérateurs avec la vue spatiale exemple U2_Local. L'opération suivante permet d'insérer un atome de vue spatiale S_Zone_Inondable à la vue spatiale U2_Local avec une rôle *Environment*:

```
Add_SVA_to_SV ( 'Base_Exemple',
                 'U2_Local',
```

```
'Environment',
[ 'S_Zone_Inondable',
{ 'Zone_Inondable' },
( 'II (Zone_Inondable, Représentation_Spatiale)' ) ] )
```

L'opérateur (Move_SVA_From_to) permet de changer le rôle d'un atome de vue spatiale, identifié par son nom (SVA_Name_type), au sein d'une vue spatiale également identifiée par son nom (SV_Name_type). Le rôle de destination de cet atome de vue spatiale est précisé par le type (SV_Role_type) :

<u>Opérateur</u>	<u>Signature</u>
Move_SVA_From_to	DB_type x SVA_Name_type x SV_Name_type x SV_Role_type --> DB_type

Nous illustrons ce changement de rôle de l'atome de vue spatiale à partir de la vue spatiale exemple U2. L'opération suivante permet d'attribuer un rôle *Environment* à l'atome de vue spatiale S_Route_Inondable de la vue spatiale U2_Local :

```
Move_SVA_From_to ('Base_Exemple', 'S_Route_Inondable',
                  'U2_Local', 'Environment')
```

L'opérateur (Rmv_SVA_from_SV) permet de détruire un atome de vue spatiale identifié par son nom (SVA_Name_type) au sein d'une vue spatiale également identifiée par son nom (SV_Name_type).

<u>Opérateur</u>	<u>Signature</u>
Rmv_SVA_from_SV	DB_type x SVA_Name_type x SV_Name_type --> DB_type

L'opération suivante efface l'atome de vue spatiale (S_Zone_Inondable) de la vue spatiale (U2_Local):

```
Rmv_SVA_from_SV('Base_Exemple', 'S_Zone_Inondable', 'U2_Local')
```

L'opérateur (Change_SV_Scale) modifie (float) l'une des composantes de l'échelle de manipulation d'une vue spatiale identifiée par son nom (SV_Name_type). Dans la mesure où tous les champs de l'échelle de manipulation d'une vue spatiale possèdent une même sémantique donc le même domaine (réel), la fonction de modification est définie de façon générique :

La composante d'échelle de la vue spatiale à modifier est définie par le type (New_Scale_type) :

```
New_Scale_type = <' Min_Limit', 'Scale_Reference', 'Max_Limit'>
```

<u>Opérateur</u>	<u>Signature</u>
Change_SV_Scale	DB_type x SV_Name_type x New_Scale_type x float --> DB_type

Nous illustrons ce changement de rôle de l'atome de vue spatiale à partir de la vue spatiale exemple U2_Local. L'opération suivante permet d'attribuer une nouvelle valeur à la borne supérieure de l'échelle de manipulation de cette vue spatiale :

Change_SV_Scale('Base_Exemple', 'U2', 'Max_Limit', 1/1 000)

Le langage de définition des atomes et des vues spatiales s'accompagne de la gestion de contraintes de propagation sur les opérations de Drop_SVA (i.e. la destruction d'un atome entraîne la destruction en cascade de cet atome dans toutes les vues auxquelles il participe voire la destruction de la vue si cet atome était le dernier) et de contraintes plus générales d'intégrité référentielle (i.e. le changement de nom d'un atome de vue spatiale est valide pour chacune des vues spatiales auxquelles il participe).

5.2.6. Langage de manipulation de la vue spatiale

La vue spatiale est manipulable par les opérateurs classiques d'un langage de manipulation dans la mesure où elle est conçue comme une extension du mécanisme de vue. Ces opérateurs sont utilisables pour interroger des instances d'atomes de vue spatiale ou des instances de vue spatiale (1, 2 et 3). Nous proposons d'étudier dans cette section les opérateurs non conventionnels de la vue spatiale qui relèvent d'une sémantique particulière non couverte par les langages de manipulation classique. Ces groupes d'opérateurs sont caractérisés à partir des fonctions suivantes :

- 1) Interrogation et manipulation de ces différents groupes homogènes (langage de manipulation classique);
- 2) Gestion et dérivation, par de nouvelles interrogations, de ces groupes (langage de manipulation classique);
- 3) Application d'opérateurs logiques et spatiaux de manipulation sur ces groupes (manipulations spécifiques de la vue spatiale).

La définition d'un langage de manipulation de la vue spatiale passe par la définition d'opérateurs basiques sur les ensembles ordonnés d'atomes de vue spatiale et sur les échelles de manipulation de la vue spatiale.

Opérateurs basiques

Order:

Trois opérateurs basiques sont définis pour manipuler les attributs (Objective, Component et Environment) de l'attribut Order de la vue spatiale :

intersection (\cap_{order}), union (\cup_{order}) et différence (\ominus_{order}).

Le résultat de ces opérateurs sont typés par l'attribut Order_type. Ces opérateurs élémentaires sont des opérateurs ensemblistes (i.e. appliqués entre deux ensembles de SVA_type).

La signature de l'opérateur élémentaire d'intersection \cap_{order} est la suivante :

$$\cap_{\text{order}} : \{ \text{SVA_type} \} \times \{ \text{SVA_type} \} \rightarrow \{ \text{SVA_type} \}$$

Les spécifications de l'opérateur d'intersection \cap_{order} sont les suivantes :

$$A = \{ \text{SVA}_i / i = 1, \dots, n \}$$

$$B = \{ \text{SVA}_j / j = 1, \dots, n \}$$

$$C \leftarrow A \cap_{\text{order}} B = \{ \text{SVA} / \text{SVA} \in A \wedge \text{SVA} \in B \}$$

La signature de l'opérateur élémentaire d'union (\cup_{order}) est la suivante :

$$\cup_{\text{order}} : \{ \text{SVA_type} \} \times \{ \text{SVA_type} \} \rightarrow \{ \text{SVA_type} \}$$

Les spécifications de l'opérateur d'union \cup_{order} sont les suivantes :

$$A = \{ \text{SVA}_i / i = 1, \dots, n \}$$

$$B = \{ \text{SVA}_j / j = 1, \dots, n \}$$

$$C \leftarrow A \cup_{\text{order}} B = \{ \text{SVA} / \text{SVA} \in A \vee \text{SVA} \in B \}$$

La signature de l'opérateur élémentaire de différence ($-_{\text{order}}$) est la suivante :

$$-_{\text{order}} : \{ \text{SVA_type} \} \times \{ \text{SVA_type} \} \rightarrow \{ \text{SVA_type} \}$$

Les spécifications de l'opérateur de différence ($-_{\text{order}}$) sont les suivantes :

$$A = \{ \text{SVA}_i / i = 1, \dots, n \}$$

$$B = \{ \text{SVA}_j / j = 1, \dots, n \}$$

$$C \leftarrow A -_{\text{order}} B = \{ \text{SVA} / \text{SVA} \in A \wedge \text{SVA} \notin B \}$$

Manipulation_Scale

Deux opérateurs binaires sont définis pour manipuler l'attribut Manipulation_Scale:

- intersection (\cap_{scale}) et union (\cup_{scale}).

Le résultat de ces opérateurs est décrit par le type de l'attribut Manipulation_Scale. Soit Manipulation_Scale_Function une fonction de manipulation d'intervalles définissant l'échelle de manipulation _ partir des bornes (Min_Limit, Max_Limit) de deux _chelles de manipulation (ex: à partir d'une fonction flou ou d'une fonction linéaire).

La signature de cette fonction est la suivante :

$$\text{Manipulation_Scale_Function: Float} \times \text{Float} \rightarrow \text{Float}$$

Chaque opérateur binaire est défini à partir de deux ensembles typés par des attributs Manipulation_Scale_type. L'opérateur d'intersection est défini par l'application récursive et symétrique d'un opérateur élémentaire d'intersection d'intervalle (\cap_{VS}) entre le premier ensemble d'intervalles et chaque intervalle du deuxième ensemble. L'opérateur d'union est défini par l'application récursive et symétrique d'un opérateur élémentaire d'union d'intervalle (\cup_{VS}) entre le premier ensemble d'intervalles et chaque intervalle du deuxième ensemble. Les spécifications de ces opérateurs sont basés sur des manipulations d'intervalles. La Figure 5.5. présente les différentes configurations.

Figure 5.5. Configurations des intervalles d'échelles

La signature de l'opérateur élémentaire d'intersection (\cap_{VS}) est la suivante :

$\cap_{VS} : \text{Elt_Manipulation_Scale_type}$
 $\times \text{Elt_Manipulation_Scale_type} \rightarrow \text{Elt_Manipulation_Scale_type}$

Soit m_i une instance de Min_Limit, D_i une instance de Scale_Reference, M_i une instance de Max_Limit. Les spécifications de l'opérateur élémentaire d'intersection (\cap_{VS}) sont les suivantes:

$C \leftarrow A \cap_{VS} B$ avec $A = \{ (m_i, D_i, M_i) / i=1, \dots, n \}$ et $B = (m_0, D_0, M_0)$

(1) $\exists i / (m_i, D_i, M_i) \in A / m_i \leq m_0 \leq M_i \wedge m_i \leq M_0 \leq M_i$

$\Rightarrow C \leftarrow (A - \{ (m_i, D_i, M_i) \}) \cup_{\text{scale}} \{ (m_0, D_0, M_0) \}$

(2) $\exists i / (m_i, D_i, M_i) \in A / m_0 \leq m_i \leq M_0 \wedge m_i \leq M_0 \leq M_i$

$\Rightarrow C \leftarrow (A - \{ (m_i, D_i, M_i) \}) \cup_{\text{scale}} \{ (m_i, \text{Manipulation_Scale_Function}(m_i, M_0), M_0) \}$

(3) $\exists i / (m_i, D_i, M_i) \in A / m_i \leq m_0 \leq M_i \wedge m_0 \leq M_i \leq M_0$

$\Rightarrow C \leftarrow (A - \{ (m_i, D_i, M_i) \}) \cup_{\text{scale}} \{ (m_0, \text{Manipulation_Scale_Function}(m_0, M_i), M_i) \}$

(4) $\exists i / (m_i, D_i, M_i) \in A / m_0 \leq m_i \leq M_0 \wedge m_0 \leq M_i \leq M_0$

$\Rightarrow C \leftarrow A$

(5) $\neg(1) \wedge \neg(2) \wedge \neg(3) \wedge \neg(4) \Rightarrow C \leftarrow A$

La signature de l'opérateur élémentaire d'union (\cup_{VS}) est la suivante :

$\cup_{VS} : \text{Elt_Manipulation_Scale_type}$
 $\times \text{Elt_Manipulation_Scale_type} \rightarrow \text{Elt_Manipulation_Scale_type}$

Les spécifications de l'opérateur élémentaire d'union (\cup_{VS}) sont les suivantes :

$C \leftarrow A \cup_{\text{scale}} B$ avec $A = \{ (m_i, D_i, M_i) / i=1, \dots, n \}$ et $B = (m_0, D_0, M_0)$

(1) $\exists i / (m_i, D_i, M_i) i = 1, \dots, n \in A / \quad m_i \leq m_0 \leq M_0 \wedge \quad m_i \leq M_0 \leq M_i$
 $\Rightarrow C \leftarrow A$

(2) $\exists i / (m_i, D_i, M_i) i = 1, \dots, n \in A / \quad m_0 \leq m_i \leq M_0 \wedge \quad m_i \leq M_0 \leq M_i$
 $\Rightarrow C \leftarrow (A - \{(m_i, D_i, M_i)\}) \cup_{\text{scale}} \{(m_0, \text{Manipulation_Scale_Function}(m_0, M_i), M_i)\}$

(3) $\exists i / (m_i, D_i, M_i) i = 1, \dots, n \in A / \quad m_i \leq m_0 \leq M_i \wedge \quad m_0 \leq M_i \leq M_0$
 $\Rightarrow C \leftarrow (A - \{(m_i, D_i, M_i)\}) \cup_{\text{scale}} \{(m_i, \text{Manipulation_Scale_Function}(m_i, M_0), M_0)\}$

(4) $\exists i / (m_i, D_i, M_i) \in A / \quad m_0 \leq m_i \leq M_0 \wedge \quad m_0 \leq M_i \leq M_0$
 $\Rightarrow C \leftarrow (A - \{(m_i, D_i, M_i)\}) \cup_{\text{scale}} \{(m_0, D_0, M_0)\}$

(5) $\neg(1) \wedge \quad \neg(2) \wedge \quad \neg(3) \wedge \quad \neg(4)$
 $\Rightarrow C \leftarrow A \cup_{\text{scale}} \{(m_0, D_0, M_0)\}$

Opérateur d'intersection de vues spatiales

La sémantique de l'opérateur SV_Intersection délivre les atomes de vue spatiale appartenant à de mêmes ensembles d'atomes de deux vues spatiales et appartenant à un intervalle d'échelle commun. Le résultat de cet opérateur est une vue spatiale définie par un nom (SV_Name_type). La signature de cet opérateur est la suivante :

SV_Intersection : SV_Name_type x SV_type x SV_type -> SV_type

Les spécifications de l'opérateur SV_Intersection sont les suivantes :

Result ← SV_Intersection (SV1, SV2)

Result.Manipulation_Scale ← SV1.Manipulation_Scale \cap_{scale} SV2.Manipulation_Scale

Result.Order ← SV1.Order \cap_{order} SV2.Order

Nous illustrons cet opérateur à partir des vues spatiales U1_Local et U2_Local. Un seul atome de vue spatiale est membre de la vue spatiale résultante (S_Bati), dans ce cas comme un membre des mêmes ensembles d'atomes (Environment) pour les deux vues opérantes de l'opérateur d'intersection de vues (Figure 5.6.) :

SV_Intersection('Intersection', 'U1_Local', 'U2_Local') =

[Name: 'Intersection',
Order: [Objective: { },
Component: { },
Environment: { 'S_Bati' }],
Manipulation_Scale: [Min_Limit: 1/7 500,
Scale_Reference: 1/5 000,
Max_Limit: 1/2 500]]

Opérateur d'intégration de vues spatiales

La sémantique de l'opérateur SV_Integration délivre l'union des atomes de vue spatiale appartenant aux ensembles d'atomes de deux vues spatiales et appartenant à un intervalle d'échelle commun. Le résultat de cet opérateur est une vue spatiale définie par un nom (SV_Name_type). La signature de cet opérateur est la suivante :

SV_Integration : SV_Name_type x SV_type x SV_type -> SV_type

Les spécifications de l'opérateur SV_Integration sont les suivantes :

Result ← SV_Integration (SV1, SV2)

Result.Manipulation_Scale ← SV1.Manipulation_Scale \cap_{scale} SV2.Manipulation_Scale

Result.Order ← SV1.Order \cup_{order} SV2.Order

Nous illustrons cet opérateur à partir des vues spatiales U1_Local et U2_Local. Notons l'application d'une contrainte (valable pour l'ensemble des opérateurs de vue spatiale): les atomes de vue spatiale membres d'ensembles d'atomes distincts apparaissent au sein de l'ensemble de la plus grande importance s_mantique (ex: l'atome S_Zone_Inondable est respectivement membre des ensembles d'atomes Component et Environment pour les vues spatiales U1_Local et U2_Local, il apparaît seulement dans la vue résultat Integration comme un membre de l'ensemble Component). La vue spatiale résultante est définie par les éléments suivants (Figure 5.6.) :

SV_Integration('Integration', 'U1_Local', 'U2_Local') =

[Name: 'Integration',
Order: [Objective: {'S_Bâti_Inondable', 'S_Route_Inondable',
'Chemins','Chemins_Contraints'},
Component: {'S_Zone_Inondable', 'S_Route_e', 'S_Route_n'},
Environment: { 'S_Bati', 'S_Route_e', 'S_Route_n' }],
Manipulation_Scale: [Min_Limit: 1/7 500,
Scale_Reference: 1/5 000,
Max_Limit: 1/2 500]]

Opérateur de fusion de vues spatiales

La sémantique de l'opérateur SV_Merge délivre les atomes de vue spatiale appartenant à de mêmes ensembles des composants de deux vues spatiales et sans restriction sur les intervalles d'échelles. Dans ce cas, des discontinuités d'échelle peuvent apparaître. Le résultat de cet opérateur est une vue spatiale définie par un nom (SV_Name_type). La signature de cet opérateur est la suivante :

SV_Merge :

SV_Merge : SV_Name_type x SV_type x SV_type -> SV_type

Les spécifications de l'opérateur SV_Merge sont les suivantes :

Result ← SV_Merge (SV1, SV2)

Result.Manipulation_Scale ← SV1.Manipulation_Scale \cup_{scale} SV2.Manipulation_Scale

Result.Order ← SV1.Order \cup_{order} SV2.Order

Nous illustrons cet opérateur à partir des vues spatiales U1_Local et U2_Local. Cet exemple d'opérateur se distingue du précédent (SV_Integration) par la valeur résultante de l'échelle de manipulation (la valeur de la borne supérieure est 1/10 000 contre 1/7 500 dans le dernier cas). La vue spatiale résultante est définie par les éléments suivants (Figure 5.6.) :

SV_Merge('Merge', 'U1_Local', 'U2_Local') =

[Name: 'Merge',
Order: [Objective: {'S_Bâti_Inondable', 'S_Route_Inondable',
'Chemins','Chemins_Constraints'},
Component: { 'S_Zone_Inondable', 'S_Route_e', 'S_Route_n' },
Environment: { 'S_Bati', 'S_Route_e', 'S_Route_n' }],
Manipulation_Scale: [Min_Limit: 1/10 000,
Scale_Reference: 1/5 000,
Max_Limit: 1/2 500]]

Opérateur de différence de vues spatiales

La sémantique de l'opérateur SV_Difference délivre les atomes appartenants à la première vue spatiale (SV_type₁) et non à la deuxième vue spatiale (SV_type₂). Cet opérateur est non symétrique. Le résultat de cet opérateur est une vue spatiale définie par un nom (SV_Name_type). La signature de cet opérateur est la suivante:

SV_Difference :

SV_Difference : SV_Name_type x SV_type₁ x SV_type₂ -> SV_type

Les spécifications de l'opérateur SV_Difference sont les suivantes :

Result ← SV_Difference (SV1, SV2)

Result.Manipulation_Scale ← SV1.Manipulation_Scale

Result.Order ← SV1.Order -_{order} SV2.Order

Nous illustrons cet opérateur à partir des vues spatiales U1_Local et U2_Local. La vue spatiale résultante est définie par les éléments suivants (Figure 5.6.) :

SV_Difference('Difference', 'U1_Local', 'U2_Local') =

[Name: 'Difference',

Order: [Objective: {'S_Bâti_Inondable', 'S_Route_Inondable'},

Component: {'S_Zone_Inondable'},

Environment: {'S_Route_e', 'S_Route_n'}],

Manipulation_Scale: [Min_Limit: 1/10 000,

Scale_Reference: 1/5 000,

Max_Limit: 1/2 500]]

Figure 5.6. - Exemples d'opérateurs sur les vues spatiales

Les opérateurs d'intégration et de fusion de vues spatiales (respectivement SV_Integration et SV_Merge) donnent un résultat proche dans l'exemple de la Figure 5.6. (i.e. les ensembles d'atomes composants sont identiques, les échelles de manipulation résultantes sont différentes). Les résultats de ces deux opérateurs sont divergents lorsque les deux vues spatiales originales possèdent des intervalles d'échelle de manipulation disjoints. Dans ce dernier cas la vue spatiale résultante sera l'ensemble vide pour l'opérateur d'intégration, et l'union des ensembles d'atomes de vue spatiale et des intervalles d'échelle de manipulation pour l'opérateur de fusion. Du point de vue sémantique, l'opérateur d'intégration est utile pour des applications souhaitant manipuler des données spatiales sous la condition qu'elles soient définies à des échelles compatibles (i.e. intervalles d'échelle de manipulation non disjoints), l'opérateur de fusion est pertinent pour des applications souhaitant manipuler le plus grand nombre de données spatiales sur un territoire considéré. L'opérateur de différence (SV_Difference) est classiquement défini. Il présente une sémantique connue qui en fait un outil de manipulation facilement utilisable.

6. APPLICATION DE LA VUE SPATIALE A UN PROCESSUS DE NAVIGATION

L'application de la vue spatiale à un contexte de description de parcours implique l'introduction de quelques définitions de base de la théorie des graphes :

Définition 6.1. Un **graphe** est un couple (N, E) où N est un ensemble de noeuds et E un sous-ensemble du produit cartésien $N \times N$. Un noeud est défini par un type (Node_type).

$$N = \{n_1, \dots, n_p\}$$

$$E = \{(n_i, n_j) / n_i \in N, n_j \in N\}$$

$$\text{Node_type} = \text{string}$$

Définition 6.2. Un **réseau** est un graphe défini par un nom (Name) et des ensembles de noeuds et d'arcs identifiés. Il modélise un système de communication.

$$\text{Network_type} = [\text{Name: string, Nodes: \{Node_type\}, Edges: \{Arc_type\}}]$$

$$\text{Arc_type} = [\text{Name: string, } n_i : \text{Node_type, } n_j : \text{Node_type}]$$

L'exemple suivant illustre la fonction des atomes de vue spatiale en représentant un parcours entre une origine (EPFL_Entrance) et une destination (EPFL_Arrival) sur un réseau défini par deux relations représentant des noeuds (Route_EPFL_n) et des arcs (Route_EPFL_e). La représentation d'un parcours est construite à partir d'un atome de vue spatiale définissant son tracé (Track), un atome de vue spatiale représentant le réseau support de ce parcours (Route_EPFL) et un atome de vue spatiale situant le parcours par un contexte cadastral (Cadastre).

```
SVA: [ Name:           'Track'
      Collection:      {'Route_EPFL_n', 'Route_EPFL_e'}
      Spatial_Query:   '--> (EPFL_Entrance, EPFL_Arrival,
                           Route_EPFL_n, Route_EPFL_e, nul, nul) ' ]
```

```
SVA: [ Name:           'Route_EPFL'
      Collection:      {'Route_EPFL_e'}
      Spatial_Query:   'Π (Route_EPFL_e, Spatial_Rep) ' ]
```

```
SVA: [ Name:           'Cadastre'
      Collection:      {'Cadastre'}
      Spatial_Query:   'Π (Cadastre, Spatial_Rep) ' ]
```

La vue spatiale exemple (EPFL_campus) est construite à partir de ces atomes de vue spatiale (Figure 6.1.). Elle précise les rôles respectifs de chacun des atomes de vue spatiale (attribut Order). L'échelle de manipulation de cette vue spatiale est un intervalle centré sur l'échelle du 1/10 000.

```
SV: [Name:          'EPFL_campus',
     Order:        [ Objective:      {'Track'},
                    Component: {'Route_EPFL'},
                    Environment:  {'Cadastre'}  ],
     Manipulation_Scale:
     { [ Min_Limit:      1/25 000,
         Scale_Reference: 1/10 000,
         Max_Limit:      1/5 000  ] } ]
```

Figure 6.1. - Exemple de vue spatiale

6.1. Modèle du processus de navigation

Le modèle du processus de navigation est construit à partir d'un ensemble d'espaces représentés à différents niveaux d'abstraction et des liaisons sémantiques entre ces espaces: les collages. La sémantique des collages est précisée par la définition 6.3., celle du graphe du processus de navigation par la définition 6.4.

Définition 6.3. Un **collage** modélise un changement d'espace dans la description d'un processus de navigation (ex: passage d'une représentation régionale à une représentation locale). Ces espaces sont représentés par des vues spatiales distinctes. Un collage est identifié par un nom (Name), il traduit une relation sémantique orientée entre deux vues spatiales identifiées par leur nom (SV_from, SV_to) (ex: une vue spatiale représentant un réseau routier à une échelle régionale et une deuxième vue spatiale représentant un réseau routier à une échelle locale). L'orientation traduit le sens du parcours.

Définition 6.4. Un **graphe de processus de navigation** est une application du concept de réseau. Il est défini par un type (Processus_Graphe_type). Ce type précise le nom (Name), l'ensemble des noeuds (Nodes) et l'ensemble des arcs (Edges) du graphe de processus de navigation. Les noeuds modélisent les vues spatiales, les arcs modélisent les collages. Ce graphe représente le support logique de réalisation des processus de navigation.

```
Collage_type = [ Name:      string ,
                 SV_from:  string ,
                 SV_to:    string ]
```

```
Processus_Graphe_type = [ Name:      string ,
                           Nodes:    { Node_type } ,
                           Edges:    { Collage_type } ]
```

Le graphe du processus de navigation est obtenu en utilisant les vues spatiales comme noeuds et les collages comme arcs (Figure 6.2.).


```

Manipulation_Scale: { [ Min_Limit: 1/75 000 ,
                        Scale_Reference: 1/50 000 ,
                        Max_Limit: 1/25 000 ] } ]
Collage_type: [ Name: 'EPFL_Arrival' ,
                SV_from: 'Highway' ,
                SV_to: 'EPFL_Campus' ]

Connection_type = [ Name: 'C2' ,
                   Section_from_Name: 'Highway_to_EPFL' ,
                   Section_to_Name: 'Track' ,
                   Node_Origin: 'Highway_Exit' ,
                   Node_End: 'EPFL_Entrance' ]

```

Figure 6.3. - Collage et connexion

Lors du collage de deux vues spatiales, une connexion n'est pas requise si le processus de navigation que l'on souhaite représenter est défini à un niveau d'abstraction général. Le collage assure la concordance sémantique des sections, la connexion la matérialisation spatiale de cette concordance. La définition suivante qualifie les types de connexion.

Définition 6.7. Une **connexion convergente** (respectivement **divergente**) est une connexion dont les instances (Origin et End) représentant les noeuds sont spatialement confondues (respectivement non confondues) aux tolérances géométriques près. Les connexions convergentes assurent la continuité spatiale (respectivement la discontinuité spatiale) du processus de navigation.

Les connexions divergentes apportent une relative souplesse lors de la réalisation de connexions entre des sections représentées à des niveaux d'abstraction de première approximation (sans obligation de parfaite continuité spatiale). Les connexions divergentes permettent une forme de raisonnement par défaut, c'est à dire assumer qu'il existe une solution de connexion et que les détails seront identifiés par l'utilisateur lors de la réalisation du processus de navigation. Une connexion peut associer deux instances de collections spatiales définies par des représentations spatiales de types différents mais correspondant à une même entité spatiale du monde réel (ex: une section autoroutière avec une représentation spatiale de type surface dans une première vue spatiale et une section autoroutière avec une représentation spatiale de type ligne dans une deuxième vue spatiale). Une connexion peut associer deux entités spatiales situées sur des zones non périphériques de la vue spatiale (continuité spatiale mais non visuelle entre les sections). La caractérisation d'une connexion peut bénéficier de l'apport d'éléments perçus comme l'association d'images photographiques ou de symboles la décrivant. Ces éléments d'informations peuvent constituer des repères visuels pour une meilleure compréhension de la description d'un processus de navigation.

Le parcours du processus de navigation est désigné par la notion de **route**. La route est définie à partir des constructeurs de collage et de vue spatiale précédemment énoncés.

Définition 6.8. Une **route** traduit la vision ordonnée d'un graphe de processus de navigation. Elle décrit la représentation mentale du parcours d'un processus de navigation à travers des espaces multidimensionnels repr_s par des vues spatiales et li_s par des collages. Une route est d_finie par un nom (Name) et par son parcours (Navigation_type). Ce parcours est repr_s par une vue spatiale suivie d'une liste de paires collage - vue spatiale.

```

Navigation_type = [ SV_Name: Node_type ,
                  Sequence: ( [Collage_Name: string ,
                              SV_Name: Node_type ] ) ]

Route_type = [ Name: string,
              Trace: Navigation_type ]

```

La construction de la route est une fonction (Trace) du graphe du processus de navigation :

```
Trace: Processus_Graphe_type --> Route_type
```

6.2. Manipulation du processus de navigation

Cette section présente les principes de manipulation du modèle de processus de navigation. La sous-section 6.2.1. indique les principes du langage de définition, la sous-section 6.2.2. définit les opérateurs externes et la sous-section 6.2.3 les opérateurs internes de changement de niveau d'abstraction du processus de navigation.

6.2.1. Langage de définition

Les composants du mod_le du processus de navigation sont pr_cis_s _ partir d'un langage classique de d_finition permettant les op_rations de cr_ation et de destruction. La d_finition des vues spatiales et des collages permettent la constitution du graphe du processus de navigation (Processus_Graphe_type). La création d'une vue spatiale est une démarche exploratoire où l'utilisateur procède par itération avant d'identifier l'échelle de manipulation correspondante à son niveau d'abstraction et un ordre cohérent sur les atomes de vue spatiale.

La route est construite en fonction des objectifs de l'utilisateur et à partir de la définition d'un ordre sur le graphe du processus de navigation (Route_type). Ce parcours sera défini par les sections précisant son tracé dans chaque vue spatiale et par les connexions entre ces sections si les diff_rents niveaux d_abstraction correspondants le permettent. L_ensemble des op_rations de cr_ation prennent la forme d'un op_rateur du type create <structure du composant>. Inversement les op_rations de destruction prennent une forme delete <nom du composant>.

6.2.2. Langage de manipulation

Le langage de manipulation du processus de navigation permet d'identifier, de visualiser et d'interroger ses différents composants. La construction de ces opérateurs relèvent de mécanismes classiques, nous en présentons des exemples pour chaque type de fonction.

Opérateurs d'identification

Ils permettent l'extraction d'une vue spatiale (SV_Extract) à partir de son nom, et les extractions de la vue spatiale d'origine (SV_Origin) et de la dernière vue spatiale (SV_End) d'un graphe de processus de navigation (Processus_Graphe_type). Un opérateur permet d'identifier la vue spatiale suivant immédiatement une vue spatiale dans un graphe de processus de navigation (SV_Browsing). Dans le même sens, un opérateur d'identifie un collage (Collage_Extract) membre d'un graphe de processus de navigation (Processus_Graphe_type) à partir de son nom.

SV_Extract:	Processus_Graphe_type x string	-->	SV_type
SV_Origin:	Processus_Graphe_type	-->	SV_type
SV_End:	Processus_Graphe_type	-->	SV_type

SV_Origin définit la vue spatiale modélisée par le noeud ni dans le graphe tel que :

$\Gamma^-(ni) = _$

SV_End définit la vue spatiale modélisée par le noeud ni dans le graphe tel que:

$\Gamma^+(ni) = _$

SV_Browsing: Processus_Graphe_type x SV_type--> SV_type

Collage_Extract: Processus_Graphe_type x string --> Collage_type

Opérateurs de visualisation

Ils réalisent la visualisation d'un collage (Collage_Display), d'une vue spatiale (SV_Display) et d'une route (Route_Display). Les éléments affichés sont définis par des types (respectivement Collage_Display_type, SV_Display_type et Route_Display_type).

Collage_Display:	Collage_type	-->	Collage_Display_type
SV_Display:	SV_type	-->	SV_Display_type
Route_Display:	Route_type	-->	Route_Display_type

Opérateurs de sélection

Ils sont construits à partir de requêtes (modélisées sous une forme de chaîne de caractères) portant sur les structures des routes. Ces opérateurs permettent de sélectionner une route par son nom (Route_Select_by_Name) et de sélectionner une route ayant une vue spatiale pour membre (Route_Select_by_SV_Name).

```
Route_Select_by_Name:          string    -->    Route_type
Route_Select_by_SV_Name:      string -->    Route_type
```

6.2.3. Langage de manipulation - niveau interne

Une route peut être exprimée en fonction de plusieurs niveaux d'abstractions. A chaque niveau d'abstraction correspond une séquence spécifique de vues spatiales et de collages. Une vue spatiale peut être décomposée en plusieurs vues spatiales reliées par des collages, cette opération se réalise à partir d'un opérateur de développement (Develop). Inversement, une séquence de vues spatiales reliées par des collages peut être regroupée en une vue spatiale, cette opération se réalise à partir d'un opérateur de regroupement (Undevelop). Ces opérateurs permettent les changements de niveau d'abstraction dans la représentation d'une route [Mainguenaud 1995].

Opérateur Develop

La signature de l'opérateur Develop est la suivante :

```
P_Graphe_type    x P_Graphe_type    x Node_type -->    P_Graphe_type
```

Le premier Processus_Graphe_type (Processus_Graphe_type) est le graphe modélisant la route initiale. Le deuxième Processus_Graphe_type précise une partie de la route initiale. Node_type est le noeud du graphe à développer (la vue spatiale dont le niveau de détail va augmenter). Ce noeud est obtenu par exemple en utilisant le nom de la vue spatiale obtenue par la fonction SV_Extract sur un point_graphique_d_fini sur un Route_Display. Le résultat est un Processus_Graphe_type (un graphe modélisant une route avec un niveau de détail plus précis que la route initiale).

Les spécifications de cet opérateur s'appuient sur les propriétés des graphes modélisés par les Processus_Graphe_type. En effet ces graphes modélisent des routes (sans cycles). Ils ont donc une origine, une extrémité. Trois cas de figures peuvent alors se présenter dans le cadre d'un développement : (a) le noeud développé est le noeud origine du graphe; (b) le noeud développé est le noeud extrémité du graphe; (c) le noeud développé n'est ni l'origine, ni l'extrémité du graphe (les noeuds modélisent les vues spatiales, les arcs les collages). Les spécifications de l'opérateur Develop peuvent donc se ramener à une étude sur les graphes.

```
G1 (N1, E1) x G2 (N2,E2) x N -> G3 (N3, E3)
```

Soient S1 et S2 les sections modélisées par les graphes G1 et G2.

Les spécifications du point (a) sont les suivantes pour le développement du noeud n_d (Figure 6.4.) :

$$\begin{aligned}
 e_{out} \text{ est définie comme: } & e_{out} \in E1 / \exists n_{out} \in N1 / e_{out} : (n_d, n_{out}) \\
 N3 = N1 - \{n_d\} \cup N2 \\
 E3 = E1 - \{e_{out}\} \cup E2 \cup \{ & (\text{End}(S2), n_{out}) \}
 \end{aligned}$$

Figure 6.4. - Développement en tête de graphe

Les spécifications du point (b) sont les suivantes pour le développement du noeud n_d (Figure 6.5.) :

$$\begin{aligned}
 e_{in} \text{ est définie comme } & e_{in} \in E1 / \exists n_{in} \in N1 / e_{in} : (n_{in}, n_d) \\
 N3 = N1 - \{n_d\} \cup N2 \\
 E3 = E1 - \{e_{in}\} \cup E2 \cup \{ & (n_{in}, \text{Origin}(S2))\}
 \end{aligned}$$

Figure 6.5. - Développement en queue de graphe

Les spécifications du point (c) sont les suivantes pour le développement du noeud n_d (Figure 6.6.) :

$$\begin{aligned}
 e_{in} = e_{in} \in E1 / \exists n_{in} \in N1 / & e_{in} : (n_{in}, n_d) \\
 e_{out} = e_{out} \in E1 / \exists n_{out} \in N1 / & e_{out} : (n_d, n_{out}) \\
 N3 = N1 - \{n_d\} \cup N2 \\
 E3 = E1 - \{e_{in}, e_{out}\} \cup E2 \cup \{ & (n_{in}, \text{Origin}(S2)), (\text{End}(S2), n_{out})\}
 \end{aligned}$$

Figure 6.6. - Développement en cours de graphe

Opérateur Undevelop

La signature de l'opérateur Undevelop est la suivante :

$$P_Graphe_type \quad x \quad P_Graphe_type \quad x \quad Node_type \quad \rightarrow \quad P_Graphe_type$$

Le premier Processus_Graphe_type est le graphe modélisant la route initiale. Le deuxième Processus_Graphe_type est le graphe modélisant les retraits d'informations (i.e., une sous-route de la route initiale - un graphe connexe issu de l'application d'un opérateur Develop). Node_type est le noeud du graphe qui représente l'abstraction de la sous-route supprimée du graphe initial (i.e., la vue spatiale représentant la sous-route). Ce noeud peut être obtenu par le même processus que l'opérateur Undevelop dans [Mainguenaud 1995]. Le résultat est un Processus_Graphe_type (un graphe modélisant une route avec un niveau de détail moins important que la route initiale).

Les spécifications de cet opérateur s'appuient également sur les propriétés des graphes modélisés par les Processus_Graphe_type. A l'image de l'opérateur Develop, ces graphes modélisent des routes (sans cycles). Ils ont donc une origine, une extrémité. Trois cas de figures peuvent alors se présenter dans le cadre d'un regroupement : (a) le noeud regroupant se positionne en noeud origine du graphe; (b) le noeud regroupant se positionne en noeud extrémité du graphe; (c) le noeud regroupant n'est positionné ni en origine, ni en extrémité du graphe.

Les spécifications de l'opérateur Undevelop peuvent donc également se ramener à une étude sur les graphes :

$$G1 (N1, E1) \times G2 (N2, E2) \times N \rightarrow G3 (N3, E3)$$

Soient S1 et S2 les sections modélisées par les graphes G1 et G2.

Les spécifications du point (a) sont les suivantes pour le regroupement du noeud nu (Figure 6.7.) :

$$e_{out} \text{ est définie comme: } e_{out} \in E1 / \exists n_{out} \in N1 / e_{out} : (\text{End}(S2), n_{out})$$

$$N3 = N1 \cup \{n_u\} - N2$$

$$E3 = E1 - \{e_{out}\} - E2 \cup \{(n_u, n_{out})\}$$

Figure 6.7. - Regroupement en tête de graphe

Les spécifications du point (b) sont les suivantes pour le regroupement du noeud n_u (Figure 6.8.) :

$$\begin{aligned}
 e_{in} & \text{ est définie comme } e_{in} \in E1 / \exists n_{in} \in N1 / e_{in} : (n_{in}, \text{Origin}(S2)) \\
 N3 & = N1 \cup \{n_u\} - N2 \\
 E3 & = E1 - \{e_{in}\} - E2 \cup \{ (n_{in}, n_u) \}
 \end{aligned}$$

Figure 6.8. - Regroupement en queue de graphe

Les spécifications du point (c) sont les suivantes pour le développement du noeud n_u (Figure 6.9.) :

$$\begin{aligned}
 e_{in} & = e_{in} \in E1 / \exists n_{in} \in N1 / e_{in} : (n_{in}, \text{Origin}(S2)) \\
 e_{out} & = e_{out} \in E1 / \exists n_{out} \in N1 / e_{out} : (\text{End}(S2), n_{out}) \\
 N3 & = N1 \cup \{n_u\} - N2 \\
 E3 & = E1 - \{e_{in}, e_{out}\} - E2 \cup \{ (n_{in}, n_u), (n_u, n_{out}) \}
 \end{aligned}$$

Figure 6.9. - Regroupement en cours de graphe

Une vue spatiale qui est l'objet d'un opérateur de développement est qualifiée de vue spatiale maître (Master_SV). La figure suivante illustre l'application de l'opérateur de développement (Develop) appliqué sur une vue spatiale EPFL (Figure 6.10.). Nous introduisons une nouvelle vue spatiale Office qui donne une représentation symbolique du lieu d'arrivée de cet exemple de route.

Le concept de graphe assure la continuité de la route lors de l'application de ces opérateurs. Cette définition **Erreur ! Source du renvoi introuvable.** permet les changements d'abstraction dans la représentation de la route et l'utilisation de vues spatiales correspondantes à chacun de ces niveaux d'abstraction. Le modèle permet de représenter des absences de connaissance dans la description de la route (ex: un utilisateur sait qu'une route sort d'un aéroport mais est incapable de décrire son parcours sur le site de cet aéroport).

L'échelle d'une route sera l'union des échelles de manipulation vues spatiales qui la composent. Compte-tenu de l'hétérogénéité des espaces présentés le plus souvent dans un processus de navigation, cette notion d'échelle donne une vision d'ensemble des niveaux d'abstraction présents. La représentation ordonnée de la route permet l'application d'un opérateur textuel (Textual) qui en donne une expression textuelle à partir du nom des vues spatiales correspondantes. La forme textuelle est construite à partir d'une forme `_From <Nom de la vue initiale>_` suivie d'une séquence de `_Path to <Nom de la vue suivante>_` sur les vues spatiales définies dans la partie Sequence du type Navigation_type. La signature de l'opérateur textuel est donnée par l'expression suivante :

Textual: Route_type --> string

Figure 6.10. - Application de l'opérateur de développement

6.3. Exemple d'Application

Soit l'exemple d'un processus de navigation démarrant du site de l_a_roport, et parcourant un r_seau autoroutier jusqu'au site de l_EPFL (Gen_ve_Airport_to_EPFL). A un niveau g_n_ral, un premier exemple de route (Gen_ve_Airport_to_EPFL) est d_fini.

```
Route :  
[  
  Name: 'Genève_Airport_to_EPFL',  
  Trace: [ SV_Name: 'Airport_site',  
          Sequence: ( [ Collage_Name: 'Airport_exit' ,  
                      SV_Name: 'Highway' ],  
                    [ Collage_Name: 'EPFL_Arrival' ,  
                      SV_Name: 'EPFL' ] ) ] ]
```

Les vues spatiales Airport_site et EPFL présentent des symboles significatifs à ce niveau d_abstraction. L_application de l_op_rateur Textual documente la visualisation pr_sent_e par la forme textuelle **_From Airport_site Path to Highway Path to EPFL**'. La Figure 6.11. présente la visualisation de cette route.

Figure 6.11. - Exemple de route à un niveau d'abstraction général

Un deuxième exemple de route donne une définition de processus de navigation à un niveau d'abstraction plus détaillé. L'objectif de cette seconde route est une présentation plus détaillée des vues spatiales qui composent ce processus de navigation. La vue spatiale _forme symbolique de l_exemple pr_c_dent (EPFL) est d_compos_e par l_application de l_op_rateur de d_veloppement.

```
Route :  
[  
  Name: 'Genève_Airport_to_EPFL',  
  Trace: [ SV_Name: 'Airport_site',  
          Sequence: ( [ Collage_Name: 'Airport_exit' ,  
                      SV_Name: 'Highway' ],  
                    [ Collage_Name: 'EPFL_campus_in' ,  
                      SV_Name: 'EPFL_campus' ],  
                    [ Collage_Name: 'Office_Arrival' ,  
                      SV_Name: 'Office' ] ) ] ]
```

La vue spatiale EPFL joue le r_le d'une vue spatiale ma"tre (Master_SV). La vue spatiale Office donne un signal visuel sur le lieu de destination. La description textuelle utilis_e est en cons_quence plus compl_te que celle de l_exemple pr_c_dent. La Figure 6.12. illustre ce deuxième exemple de représentation de la route.

Figure 6.12. - Exemple de route à un niveau d'abstraction de détail

Ces schémas de construction de routes sont extensibles avec le temps, ils peuvent s'enrichir de nouvelles informations issues d'expérience ou d'apprentissage des utilisateurs.

La séquence d'une route permet le déclenchement d'événements de type condition-action (condition = collage, action = initialisation du déplacement sur une vue spatiale) nécessaires pour mettre en oeuvre un processus de navigation [Kuipers 1978]. Un collage donne une condition de passage pour la poursuite de la navigation à travers une section. Les actions à entreprendre (ex: right, left, until, next, selon [Maaß 1993] pour le parcours de la route et les opérateurs de positionnement associés [Kuipers 1978] peuvent exploiter ce modèle. Cette perspective est applicable pour les opérations de guidage en temps réel.

6.4. Discussion

La représentation coordonnée d'une route avec des espaces multidimensionnels associés répond à un besoin des utilisateurs de démarches de navigation. Elle doit permettre de conserver la mémoire de ces déplacements et d'en faciliter leur assimilation, de les r_utiliser et d_assurer leur mise _ jour.

Le modèle proposé permet de situer une action de d_placement dans son contexte g_ographique _ des niveaux d_abstraction compl_mentaires qui acceptent des connaissances partielles. La vue spatiale donne un cadre de repr_sentation d_un processus de navigation _ caract_re informationnel. Elle associe la visualisation d'une route avec les espaces multidimensionnels qui permettent de la situer, avec des points de repères visuels significatifs et avec un langage textuel de description. La continuité de la représentation de la route est assurée par le concept de graphe appliqué aux collages et aux vues spatiales. Ces collages sont matérialisés spatialement par des connexions. La trace d'une route dans chaque espace représenté par une vue spatiale est décrite par la notion de section. Des opérateurs de d_veloppement et de regroupement permettent les changements de niveau d_abstraction dans la repr_sentation d_une route.

Cette application de la vue spatiale est en mesure de s'appliquer à des contextes de présentation de schémas de guidage routier pour visiteurs, _ l__tude pr_alable de parcours touristiques ou professionnels. Les navigations a_riennes ou nautiques sont d_autres cas d_applications potentiellement adapt_es.

7. CONCLUSION

Cette étude tend à fournir plusieurs résultats convergents. La démonstration de l'intérêt d'un concept de vue spatiale dans un contexte d'application géographique est le premier apport que le développement de cette recherche contribue à produire. Il apparaît en effet clairement que les bénéfices classiques des vues sont renforcés dans le cas des applications spatiales. Le potentiel d'application du concept de vue spatiale dans un contexte de bases de données spatiales est relativement large, un grand nombre d'organisations publiques et privées sont en effet notamment gestionnaires et utilisatrices de données géographiques. Le coût de la donnée spatiale étant à fortiori très élevé, les organismes concernés tendent à définir des règles méthodologiques et institutionnelles de partage et d'échange de données géographiques qui renforcent les besoins d'utilisation dérivés des données disponibles. La complexité des applications spatiales et la difficulté de modéliser à priori et de façon définitive une application spatiale entraîne une pression forte en terme d'évolution du schéma d'une base de données géographiques. Le grand nombre d'applications potentielles génère enfin une grande variété dans les besoins de traitement des données spatiales, d'où un intérêt marqué pour la définition d'un niveau de schéma externe permettant la dérivation de données spatiales en fonction des besoins des utilisateurs.

Cette thèse identifie les moyens nécessaires à la définition d'une vue spatiale en terme de modèle spatial, de langage d'interrogation, et de sémantique produite par la vue spatiale. La vue spatiale proposée permet de décrire des représentations cognitives et structurées d'un espace géographique, son potentiel descriptif est donc relativement large. La vue spatiale est définie pour une échelle de manipulation donnée, ce qui apporte une bonne flexibilité au modèle dans la mesure où cette notion permet, en prolongement au niveau du modèle spatial, de supporter la représentation de données spatiales définies à différents niveaux d'abstraction. Le concept d'atome de vue spatiale, qui modélise une interrogation spatiale et/ou non spatiale apporte une incrémentation souple à la vue spatiale. Il permet de manipuler les vues spatiales en terme de définition de l'importance sémantique du rôle des différentes interrogations au sein de la vue spatiale. Une notion d'ordre est définie selon le rôle joué par les requêtes au sein de la vue spatiale: les atomes objectifs de la vue spatiale, les atomes présentant des données spatiales participant aux atomes objectifs de la vue spatiale et les atomes présentant le contexte géographique environnant de la vue spatiale. Un langage de manipulation des vues spatiales est proposé. Il exploite les propriétés des notions d'ordre sur les atomes de vue spatiale et de l'échelle de manipulation de la vue spatiale. Il comporte des opérateurs d'intersection, d'intégration, de fusion et de différence sur les vues spatiales.

Le modèle de vue spatiale est illustré dans un contexte de modèle cartographique classique et dans celui d'une représentation navigationnelle proche d'un modèle spatial cognitif. L'application de la vue spatiale à un cas de représentation d'un processus de navigation amène notamment la définition de propriétés particulières des vues spatiales. Des concepts de collages de vues spatiales, représentant des espaces multi-dimensionnels et des connexions entre entités de ces espaces respectifs, sont identifiés et modélisés. L'application d'opérateurs de graphe permet des changements de niveau d'abstraction dans la représentation de processus de navigation.

La présentation d'applications dans des contextes applicatifs complémentaires apporte une première démonstration de l'intérêt du concept et de son adaptation dans la mesure où le contexte navigational permet en prolongement d'identifier des propriétés sémantiques dérivées. Les contextes applicatifs identifiés dans cette étude peuvent s'élargir à d'autres domaines utilisant plusieurs niveaux d'abstraction dans la représentation de données

spatiales. Ce dernier cadre applicatif constitue le terrain privilégié d'application générique de la vue spatiale: des bases de données spatiales utilisées par un nombre important d'utilisateurs (i.e. une demande forte de définition d'un schéma externe), combinant plusieurs espaces de représentation dans l'utilisation des données spatiales et comportant un degré significatif de dérivation possible des données spatiales. Les domaines couverts sont assez large, de la gestion d'application de SIG classiques à la mise en oeuvre de projets plus novateurs dans le domaine de la gestion de déplacements terrestres ou aériens.

Une contrainte est apparue lors du développement de cette recherche: l'absence de consensus sur les modèles spatiaux, les langages d'interrogation et de visualisation des données spatiales. Ce manque d'une référence stable impose en effet d'identifier un ensemble de propriétés spatiales (i.e. modèle et langage d'interrogation) pour chacun des composants de la vue spatiale afin de maximiser la portée de notre proposition. Cette absence de standardisation (où plus contraignant la multiplication des actions de normalisation actuellement en cours dans le domaine des bases de données spatiales) pose un problème plus général aux recherches en bases de données spatiales, il est en effet difficile de se baser sur un support de référence stable dans toute démarche de recherche. Le phénomène de représentation multiple reste également difficile à modéliser, la propagation des propriétés sémantiques des différents niveaux de représentation est un domaine de recherche toujours d'actualité.

Le modèle de vue spatiale proposé, et les concepts identifiés à travers sa définition, apportent quelques problématiques d'étude qui pourraient être abordées en prolongement de cette thèse. Au niveau du modèle spatial de la vue spatiale, la sémantique des informations alphanumériques et thématiques est relativement bien définie, par contre celle des données et opérateurs réseaux restent encore à approfondir. Une telle démarche permettrait de mieux contrôler la sémantique résultante des opérations réseaux au sein de la vue spatiale et leur coordination avec les résultats d'opérations alphanumériques et spatiales.

La vue spatiale est définie comme un modèle, la définition d'un environnement d'interface utilisateur qui intégrerait à la fois des opérateurs visuels et la gestion graphique de la présentation de vues spatiales et de ses composants apporterait un niveau d'interaction orienté utilisateur indispensable à son utilisation dans un contexte réel. Parmi les pistes à privilégier, la traduction du langage de définition et de manipulation des vues spatiales en un langage de manipulation visuel représente un potentiel de recherche et d'application intéressant.

Ces prolongements doivent s'accompagner, en parallèle ou au préalable, d'une validation et d'un enrichissement par le prototypage de l'ensemble des concepts et des notions identifiés et proposés à travers le modèle de vue spatiale. Deux options de validation, complémentaires, sont possibles. Du point de vue de la validation des concepts avancés, un prototypage rapide à partir de solutions actuelles de SIG. Cette solution, si elle ne permet pas l'implémentation véritable de la vue spatiale au niveau d'une base de données spatiales, constitue une démarche exploratoire potentielle importante sur le plan de la validation et de l'enrichissement des concepts avancés. Un tel outil permettrait notamment de tester les notions proposées dans un contexte d'utilisation. Parmi les directions applicatives intéressantes, citons la gestion de processus dynamiques dans des espaces géographiques, et notamment la gestion de parcours en milieu urbain (e.g. planification de trajets de bus).

La deuxième direction de validation est celle d'une véritable implémentation du concept de vue spatiale à partir d'un moteur de base de données spatiales. Une telle démarche nécessite cependant la mise à disposition d'un outil et d'un environnement de développement ouvert. Le coût des développements est ici relativement important, un tel

projet relève d'une réalisation à mener soit en collaboration avec un éditeur de SIG ou de base de données spatiales, ou dans le cadre d'une recherche de doctorat qui pourrait aborder l'implémentation d'une vue spatiale avec ses outils de manipulation.

Les recherches en bases de données spatiales pour des applications de SIG restent encore un domaine largement ouvert, la sémantique des applications géographiques n'est pas encore complètement stabilisée, les outils de manipulation et d'analyse sont multiples et difficilement représentables par un ensemble d'opérateurs et de fonctions fini. La grande variété d'applications de SIG, et des sciences et techniques contribuant à ce domaine, en font un cadre de multi-disciplinarité par excellence qui facilite les rencontres à travers les applications, les projets, les développements et les recherches. Nul doute que le développement des SIG permettra de générer de belles réussites sur les plans technologiques, environnementaux et humains.

Références

- [Abel 1992] Abel, D. J., Yap, S. K., Ackland, R. G., Cameron, M. A., Smith, D. F. et Walker, F. G., Environmental decision support system project: An exploration of alternative architectures for geographic information systems, *International Journal of GIS*, vol. 6, n. 3, pp. 193-204, 1992.
- [Abiteboul 1988] Abiteboul, S., *On the Manipulation of Complex Objects*, Rapport Technique, INRIA et Hebrew University, 1988.
- [Abiteboul 1991] Abiteboul, S. et Bonner, A., Object and views, *ACM SIGMOD Conference on Management of Data*, pp. 238-247, 1991.
- [Adiba 1987] Adiba, M. E., Modeling complex objects for multimedia databases, *Entity-Relationship Approach : Ten Years of Experience in Information Modeling*, Spaccapietra, S. Eds., Dijon, pp. 89-117, 1987.
- [Ady 1990] Ady, B. et Currie, A., Incorporating interpretive concepts from a discipline in a GIS: A geoscience example, *GIS/LIS Conference Proceedings*, pp. 787-793, 1990.
- [ANSI 1975] ANSI-X3/SPARC Study Group on Data Base Management System, Technical Report, ACM SIGMOD, 7 (2), 1975.
- [Aref 1991] Aref, W. G. et Samet, H., Extending a DBMS with spatial operations, *Advances in Spatial Databases*, Günther, O. et Schek, H.-J. Eds, Springer-Verlag, Zurich, pp. 299-318, 1991.
- [Armstrong 1989] Armstrong, M. P. et Desham, P. J., Database organisation strategies for spatial decision support systems, *International Journal of GIS*, Taylor and Francis, vol. 4, n. 1, pp. 3-20, 1989.
- [Aronoff 1989] Aronoff, S., *Geographical Information Systems: A Management Perspective*, WDL Publications, Ottawa, Canada, 1989.
- [Assmann 1986] Assmann, K., Venema, R. et Hohne, K. H., The ISQL language: A software tool for the development of pictorial information systems in medicine, *Visual Languages*, Chang, S. K., Ichikawa, T. et Ligomenides, P. Eds., Plenum Press, pp. 261-284, 1986.
- [Bachi 1968] Bachi, R., *Graphical Rational Patterns*, Universities Press, Jerusalem, Israël, 1968.
- [Banerjee 1987] Banerjee, J., Kim, W., Kim, H. J., et Korth, H. F., Semantics and implementation of schema evolution in object-oriented databases, *ACM-SIGMOD Conference on Management of Data*, San Francisco, Californie, pp. 311-322, 1987.
- [Banting 1989] Banting, D., Geographic information models, *Annual General Meeting of the Canadian Institute of Surveying and Mapping*, Halifax, Nouvelle-Ecosse, Canada, 1989.
- [Barrera 1981] Barrera, R. et Buchmann, A., Schema definition and query language for a geographical database system, *IEEE Transactions on Computer Architecture: Pattern Analysis and image Database Management*, vol. 11, pp. 250-256, 1981.
- [Barthet 1993] Barthet, M. F., *Interface Homme-machine et Bases de Données*, Cours Postgrade en Informatique, EPFL, Lausanne, 1993.
- [Beard 1991] Beard K., Constraints on rule formation, *Map Generalization*, Buttenfield, B. P. et McMaster, R. B. Eds, Longman Scientific and Technical, Royaume-Uni, pp.121-135, 1991.
- [Becker 1990] Becker, B. et Widmayer, P., *Spatial Priority Search and Access Techniques for Scaleless Maps*, Rapport Technique, Institute for Informatics, University of Freiburg, Allemagne, 1990.
- [Bedard 1989] Bedard, Y., Mise en place d'un cadre bidimensionnel de classification des SIRS, *Mensuration Photogrammétrique et Génie rural*, 10/89, Zurich, pp. 572-577, 1989.
- [Bertin 1983] Bertin, J., *Semiology of graphics*, Madison, Univ. of Wisconsin Press, 1983.
- [Bertino 1988] Bertino, E. et Haas, L. M., Views and security in distributed database management systems: Concepts and issues, *International Conference on Extending Data Base Technology*, Venise, Italie, pp. 155-169, 1988.
- [Bertino 1992] Bertino, E., A view mechanism for object-oriented databases, *Advances in Database Technology Conference - EDBT'92*, Pirotte, A., Delobel, C. et Gottlob, G. Eds., Springer-Verlag, Vienne, Autriche, pp. 136-151, 1992.
- [Boursier 1992] Boursier, P. et Mainguenaud, M., Spatial query languages: Extended SQL vs. visual languages vs. hypermaps, *5th International Symposium on Spatial Data Handling*, Charleston, pp. 249-259, 1992.

- [Brassel 1987] Brassel, K. E et Weibel, R., A review and conceptual framework of automated map generalization, *International Journal of Geographical Information Systems*, Taylor et Francis, 2 (3), pp. 229-244, 1987.
- [Brodie 1984] Brodie, M., On the development of data model, *Conceptual Modeling: Perspectives from Artificial Intelligence, Database and Programming Languages*, Brodie, M., Mylopoulos J. et Schmidt S. Eds., Springer-Verlag, New York, pp. 19-48, 1984.
- [Brunet 1990] Brunet, R., *Le Territoire dans les Turbulences*, Collection Géographiques, GIP Reclus, Montpellier, 1990.
- [Burrough 1986] Burrough, P. A., *Principles of Geographical Information Systems for Land Resources Assesment*, Oxford Sciences Publications, 1986.
- [Buttenfield 1989] Buttenfield, B. P. et De Lotto, J., *Multiple Representation: Report 89-3 on the Specialist Meeting*, NCGIA, Santa Barbara, 1989.
- [Buttenfield 1990] Buttenfield, B. P. et Ganter, J. H., Visualization and GIS: What should we see? What might we miss, *4th International Symposium on Spatial Data Handling*, IGU, Zurich, pp. 307-316, 1990.
- [Buttenfield 1991] Buttenfield, B. P. et Mackaness, W. A., Visualization, *GIS Principles and Applications*, Maguire Goodchild et Rhind Eds., Longman, New York, pp. 427-443, 1991.
- [Calcinelli 1991] Calcinelli, D. et Mainguenaud, M., The management of the ambiguities in a graphical query language for GIS, *Advances in Spatial Databases*, Günther, O. et Schek, H.-J. Eds., Springer-Verlag, Zurich, Suisse, pp. 99-118, 1991.
- [Calcinelli 1994] Calcinelli, D. et Mainguenaud, M., Cigales, A visual query language for geographical information system: The user interface, *Journal of Visual Language and Computing*, Academic Press, (5) pp. 113-132, 1994.
- [Card 1983] Card, S. K., Moran, T. P. et Newell, A., *The Psychology of Human-computer Interaction*, Lawrence Erlbaum Associates, 1983.
- [Ceri 1991] Ceri, S. et Widom, J., Deriving production rules for incremental view maintenance, *17th Int. Conference on Very Large Data Bases*, Lohman, G., Sernadas, A. et Camps, R. Eds., Barcelone, Espagne, pp. 577-589, 1991.
- [Chang 1988] Chang, S. K., Yan, C. W., Dimitroff, D. C. et Arndt, T., An intelligent image database system, *IEEE Transaction Software Engineering*, 14 (6), pp. 681-688, 1988.
- [Chen 1989] Chen, S., A spherical model for navigation and spatial reasoning, *Mapping and Spatial Modelling for Navigation*, Pau, L. F. Ed., Springer-Verlag, Fano, Danemark, pp. 59-72, 1989.
- [Claramunt 1994] Claramunt, C. et Mainguenaud, M., Identification of a definition formalism for a spatial view, *Advanced Geographic Data Modeling*, Molenaar, M. et de Hoop, S. Eds., Delft, Pays-Bas, pp. 191-203, 1994.
- [Claramunt 1995a] Claramunt, C. et Mainguenaud, M., Dynamic and flexible vision of a spatial database. *DEXA'95 International Workshop on Database and Expert System Applications*, Revell, N. et Min Tjoa, A. Eds., Omnipress, Londres, Royaume-Uni, pp. 483-492, 1995.
- [Claramunt 1995b] Claramunt, C. et Thériault, M., Managing time in GIS: An event-oriented approach, *Recent Advances in Temporal Databases*, Clifford, J. et Tuzhilin, A. Eds., Springer-Verlag, Zurich, pp. 23-42, 1995.
- [Claramunt 1996a] Claramunt, C. et Thériault, M., A spatial data model for representing spatio-temporal processes. *VIIth International Conference on Spatial Data Handling*, Kraak, M.-J. et Molenaar, M. Eds., Taylor et Francis, Delft, pp. 27-43, 1996.
- [Claramunt 1996b] Claramunt, C. et Mainguenaud, M., A spatial data model for navigation knowledge, *VIIIth International Conference on Spatial Data Handling*, Kraak, M.-J. et Molenaar, M. Eds., Taylor et Francis, Delft, pp. 345-357, 1996.
- [Codd 1970] Codd, E., A Relational model for large shared data banks, *Communications of the Association for Computing Machinery*, 13 (6), pp. 377-387, 1970.
- [Coppock 1991] Coppock, J. T. and Rhind, D. W., The history of GIS, *GIS: Principles and Applications*, Maguire, D. J., Goodchild, M. F. et Rhind, D. W. Eds., Longman Scientific and Technical, Royaume-Uni, pp. 21-43, 1991.
- [Curnow 1990] Curnow, W., *Art and Cartography in New Zealand*, City Art Gallery, Wellington, 1990.

- [Dadam 1986] Dadam, P. et al., A DBMS prototype to support extended NF2 relations: An integrated view on flat tables and hierarchies, *ACM SIGMOD*, Washington, pp. 356-367, 1986.
- [Date 1986] Date, D. J., *Relational Databases: Selected Writings*, Addison-Wesley, 1986.
- [David 1991] David, B., *Modélisation, Représentation et Gestion d'Information Géographique*. Thèse de doctorat, Paris VI, 1991.
- [Dayal 1985] Dayal, U. et Smith, J. M., PROBE: A knowledge-oriented database management system, *Knowledge Base Management Systems: Integrating Artificial Intelligence and Knowledge Base Management Systems*, pp. 227-257, 1985.
- [Devogele 1996] Devogele, T., Trevisan, J. et Raynal, L., Building a multi-scale database with scale-transition relationships, *VIIth International Conference on Spatial Data Handling*, Kraak, M-J. et Molenaar, M. Eds., Taylor et Francis, I:6, pp. 19-33, Delft, 1996.
- [Downs 1977] Downs, R., M et Stea, D., *Maps in Minds: Reflections in Cognitive Mapping*, Harper et Row Eds., New-York, 1977.
- [Egenhofer 1987] Egenhofer, M., An extended SQL syntax to treat spatial objects, *2nd International Seminar on Trends and Concerns of Spatial Sciences*, New Brunswick, 1987.
- [Egenhofer 1990a] Egenhofer, M., Interaction with GISs via spatial queries, *Visual Languages and Computing*, 1 (4), pp. 389-413, 1990.
- [Egenhofer 1990b] Egenhofer, M., Manipulating the graphical representation of query results in geographic information systems, *IEEE Workshop on Visual Languages*, Skokie, Illinois, pp. 119-124, 1990.
- [Egenhofer 1991a] Egenhofer, M., Reasoning about binary topological relations, *Advances in Spatial Databases*, Günther, O. et Schek, H.-J. Eds, Springer-Verlag, Zurich, Suisse, pp. 143-160, 1991.
- [Egenhofer 1991b] Egenhofer, M. J., Why not SQL, *International Journal of Geographical Information Systems*, Taylor et Francis, 1993, 5 (2), pp. 230-245, 1991.
- [Egenhofer 1992] Egenhofer, M. J. et Herring, J. R., *Categorizing Binary Topological Relationships between Regions, Lines, and Points in Geographic Databases*, Department of Survey Engineering, University of Maine, 1992.
- [Egenhofer 1995a] Egenhofer, M. J. et Bruns, H. T., Visual map algebra: A direct manipulation user interface for GIS, *Visual Database System Conference*, Spaccapietra, S. et Jain, R. Eds., Chapman et Hall, Lausanne, pp. 211-226, 1995.
- [Egenhofer 1995b] Egenhofer, M. J. et D. Mark, D., Naive geography, *Lecture Notes in Computer Science*, Vol. 988, A. Frank and W. Kuhn (eds.), pp. 1-15, 1995.
- [Elmasri 1989] Elmasri, R. et Navathe, S., *Fundamentals of Database Systems*, Benjamin et Cummings série, Californie, 1989.
- [ESRI 1994] *Understanding GIS: the ARC/INFO method*, ESRI INC., version 7 for Unix and Open VMS, 1994.
- [ETHZ 1991] *Transfert de Données*, Actes des Journées d'Etude, ETHZ, Zurich, 1991.
- [Fedra 1991] Fedra, K., A computer-based approach to environmental impact assessment, *Workshop on Indicators and Indices for Environmental Impact Assessment and Risk Analysis*, pp. 11-40, 1991.
- [Ferguson 1977] Ferguson, E. S., The mind's eye: Non verbal thought in technology, *Science*, 197 (4306), pp. 827-836, 1977.
- [Feuchtwanger 1993] Feuchtwanger, M., *Towards a Geographic Semantic Database Model*, PhD Thesis, Simon Fraser University, Department of Geography, Burnaby, B. C., Canada, 1993.
- [Forer 1990] Forer, P., Poiker, T., Penny, J. et Decker, G., New technology and new concepts for map use, *4th International Symposium on Spatial Data Handling*, IGU, Zurich, pp. 335-344, 1990.
- [Frank 1982] Frank, A. U., Mapquery - Database query language for retrieval of geometric data and its graphical representation, *ACM Computer Graphics*, vol. 16, n. 3., pp. 199-207, 1982.
- [Frank 1986] Frank, A. U. et Kuhn, W., Cell graphs: A provable correct method for the storage of geometry. *2nd International Symposium on Spatial Data Handling*, IGU Commission, IGU, Seattle, pp. 411-436, 1986.

- [Frank 1991] Frank, A. U. et Mark, D., Language issues for GIS, *Visualization of Spatial Data Quality*, NCGIA Research Initiative 7 Specialist Meeting Report, Castine, Maine, 1991.
- [Frank 1994a] Frank, A. U., Qualitative spatial reasoning about cardinal directions, *Journal of Visual Languages and Computing*, 3, pp. 343-371, 1994.
- [Frank 1994b] Frank, A. U., Qualitative temporal reasoning in GIS - Ordered time scales, *6th International Symposium on Spatial Data Handling Conference*, Waugh, T. C. et Healey, R. C. Eds., Londres, Taylor et Francis, pp. 410-430, 1994.
- [Frank 1995] Frank, A. U., et Timpf, S., Multiple representations for cartographic objects in a multi-scale tree - An intelligent graphical zoom, *Computers and Graphics*, Elsevier Science Ltd, Londres, vol 18, n 6, pp. 823-829, 1995.
- [Galton 1995] Galton, A., Towards a qualitative theory of movement, *Spatial Information Theory: A theoretical basis for GIS*, Frank, A. U. et Kuhn, W. Eds., Berlin, Springer-Verlag, pp. 377-396, 1995.
- [Ganter 1988] Ganter, J. H., Interactive graphics: Linking the human to the model, *GIS/LIS Conference Proceedings*, 1988, 230-239, 1988.
- [Gardarin 1987] Gardarin, G. et al., Sabrina, a relational database system developed in a research environment, *Technologies and Sciences of Informatics*, AFCET-Gauthier, Willey and Sons Ltd, 1987.
- [Gardarin 1991] Gardarin, G., *Bases de Données: Les Systèmes et leurs Langages*, Eyrolles, Paris, 1991.
- [Gatrell 1991] Gatrell, A. C., Concepts of space and geographical data, *GIS: Principles and Applications*, Maguire, D. J., Goodchild, M. F. et Rhind, D. W. Eds., Longman Scientific and Technical, Royaume-Uni, pp. 119-134, 1991.
- [Goldberg 1983] Goldberg, A. et Robson, D., *Smalltalk-80: The Language and its Implementation*, Addison-Wesley, 1983.
- [Günther 1990] Günther, O., Buchmann, P., Research Issues in Spatial Databases, *Data Engineering Bulletin*, 13(4): 35-42, 1990.
- [Günther 1991] Günther, O. et Bilmes, J., Tree-based access methods for spatial databases: Implementation and performance evaluation, *IEEE Transactions on Knowledge and Data Engineering*, 3 (3), pp. 342-356, 1991.
- [Güting 1989] Güting, R. H., GRAL: An extensible relational database system for geometric applications, *15th International Conference on Very Large Data Bases*, Brighton, pp. 33-44, 1989.
- [Güting 1993] Güting, R. H. et Schneider, M., Realms: A foundation for spatial data types in database systems, *Advances in Spatial Databases*, Abel, D. and Ooi, B. C. Eds, Springer-Verlag, Singapour, pp 14-35, 1993.
- [Güting 1994] Güting, R. H., An introduction to spatial database systems, *VLDB Journal*, Schek, H.-J. Ed., Vol. 3., pp. 357-399, 1994.
- [Güting 1995] Güting, R. H., de Ridder, T. et Schneider, M., Implementation of the ROSE algebra: Efficient algorithms for Realm-based spatial data types, *Advances in Spatial Databases*, Egenhofer, M. J. et Herring, J. R., Springer-Verlag, Portland, pp 216-239, 1995.
- [Gupta 1991] Gupta, A., Weymouth, T. E. et Jain, R., An extended object-oriented data model for large image bases, *Advances in Spatial Databases*, Günther, O. et Schek, H.-J. Eds., Springer-Verlag, Zurich, pp. 45-61, 1991.
- [Guptill 1991] Guptill, S. C., Spatial data exchange and standardization, *GIS Principles and Applications*, Maguire, D. J., Goodchild, M. F. et Rhind, D. W. Eds., Longman Scientific and Technical, Royaume-Uni, pp. 513-530, 1991.
- [Guttman 1984] Guttman, A., R-Trees: A dynamic index structure for spatial searching, *ACM SIGMOD Conference on Management of Data*, Boston, pp. 47-57, 1984.
- [Haarslev 1990] Haarslev, V. et Möller, R., A framework for visualizing object-oriented systems, *IEEE Workshop on Visual Languages*, Skokie, Illinois, 1990.
- [Haas 1991] Haas, L. et Cody, W. F., Exploiting extensible DBMS in integrated GIS, *Advances in Spatial Databases*, Günther, O. et Schek, H.-J. Eds, Springer-Verlag, Zurich, pp. 423-450, 1991.
- [Haller 1990] Haller, S. M. et Mark, D. M., Knowledge representation for understanding geographical locatives, *4th International Symposium on Spatial Data Handling*, IGU, Zurich, pp. 465-477, 1990.

- [Hazelton 1991] Hazelton, N. W. J., *Integrating Time, Dynamic Modelling and Geographical Information Systems: Development of Four-dimensional GIS*, PhD Thesis, University of Melbourne, Australia, 1991.
- [Heres 1993] Heres, L. et al., *GDF: A proposed standard for digital road maps to be used in car navigation systems*, Rapport Technique, Philips CE/Cis-Lab, Eindhoven, 1993.
- [Hernandez 1993] Hernandez, D., Maintaining qualitative spatial knowledge, *Spatial Information Theory*, Frank, A. U. et Campari, I. Eds., Springer-Verlag, pp. 36-53, 1993.
- [Hildebrandt 1993] Hildebrandt, J. et Tang, K., *Symbolic Two and Three Dimensional Picture Retrieval*, Rapport Technique, DSTO, Information Technology Division, Jamison, Australie, 1993.
- [Hogan 1993], Hogan, R. L., Developing a spatial feature register, *Annual URISA Conference*, Atlanta, 2, pp. 117-128, 1993.
- [Imhoff 1982] Imhoff E., *Cartographic Relief Presentation*, Gruyter, Berlin, 1982.
- [Jones 1989] Jones, C. B., Data structures for three-dimensional spatial information systems in geology, *International of Geographical Information Systems*, Taylor and Francis, vol. 3, n.1, pp. 15-31, 1989.
- [Joseph 1988] Joseph, T. et Cardenas, A.F., PICQUERY: A high level query language for pictorial database management, *IEEE Transaction Software Engineering*, 14 (6), pp. 630-638, 1988.
- [Keates 1982] Keates, J., S., *Understanding Maps*, Longman, Londres, 1982.
- [Kim 1989] Kim, W., A model of queries in object-oriented databases, *15th Int. Conference on Very Large Data Bases*, pp. 423-432, 1989.
- [Kirby 1990] Kirby, K. C. et Pazner, M., Graphic map algebra, *4th International Symposium on Spatial Data Handling*, IGU, Zurich, pp. 413-422, 1990.
- [Kirpatrick 1989] Kirpatrick, K., The reduction and representation of reality: Problems and potential of GIS, *15th New Zealand Geographical Society Conference*, Dunedin, pp. 88-99, 1989.
- [Kolovson 1993] Kolovson, P.C., Neimat, M. et Potamianos, S., Interoperability of spatial and attribute data managers: A case study, *Advances in Spatial Databases*, Abel, D. J. et Ooi, B. C. Eds. Springer-Verlag, Singapour, pp. 239-263, 1993.
- [Kriegel 1993] Kriegel, H. P., Brinkhoff, T. et Schneider, R., Efficient spatial query processing in geographic database systems, *IEEE Data Engineering Bulletin*, 6 (3), pp. 10-15, 1993.
- [Kuipers 1978] Kuiper, B., Modeling spatial knowledge. *Cognitive Science*, Vol. 2, pp. 129-153, 1978.
- [Lagacherie 1992] Lagacherie, P., *Formalisation de la Distribution des Sols pour Automatiser la Cartographie Pédologique à Partir d'un Secteur Pris comme Référence*, Thèse d'Etat, Université Montpellier II, 1992.
- [Lagrange 1994] Lagrange, J. P. et Ruas, A., Geographic information modelling: GIS and generalisation, *VI International Conference on Spatial Data Handling*, Waugh, T. C. et Healey, R. C. Eds., Taylor et Francis, pp. 1099-1117, 1994.
- [Langran 1992] Langran, G., *Time in Geographical Information Systems*, Taylor et Francis, Londres, 1992.
- [Larue 1993] Larue, T., Pastre, D. et Viémont, Y., Strong integration of spatial domains and operators in a relational database system, *Advances in Spatial Databases*, Abel, D. J. et Ooi, B. C. Eds., Springer-Verlag, Singapour, pp. 53-71, 1993.
- [Laurini 1992] Laurini, R. et Thompson, D., *Fundamentals of Spatial Information Systems*, Academic Press, Londres, 1992.
- [Lindsay 1990], Lindsay, B. et Haas, L., Extensibility in the Starburst Experimental Database System, Database Systems of the 90s, Blaser, A. Ed., Springer-Verlag, Berlin, pp. 217-248, 1990.
- [Lorie 1983] Lorie, R. et Pouffe, W., Complex objects and their use in design transactions, *ACM SIGMOD Engineering Design Applications*, San José, USA, pp. 115-121, 1983.
- [Lorie 1991] Lorie, R., The use of a complex object language in geographic data management, *Advances in Spatial Databases*, Günther, O. et Schek, H.-J. Eds., Springer-Verlag, Zurich, pp. 319-337, 1991.
- [Lynch 1960] Lynch, K., *The image of the city*, MIT Press, Massachusetts, USA, 1960.

- [Maaß 1993] Maaß, W., A Cognitive model for the process of multimodal, incremental route descriptions, *Theories and methods of spatio-temporal reasoning in geographic space*, Frank, A. U., Campari, I et Formentini, U. Eds, Springer-Verlag, Ile d'Elbe, Italie, pp. 1-13, 1993.
- [MacCormick 1987] McCormick, B. H., Defanti, T. A. et Brown, M. D., Visualization in scientific computing, *SIGGRAPH Computer Graphics Newsletter*, 21 (6), pp. 345-352, 1987.
- [MacEachren 1992] MacEachren, A., Buttenfield, B. P., Campbell, J. C. et Monmonier, M. S., *Visualizations: Geography's Inner World*, University Press, New Jersey, 1992.
- [MacMaster 1987] Mac Master, R. B. et Thrower, N. J., University cartographic education in the United States: Tracing the routes, *International Cartographic Association Conference*, Morelia, vol 2, pp. 343-359, 1987.
- [Mainguenaud 1990] Mainguenaud, M., et Portier, M. A., CIGALES: A graphical query language for GIS, *International Symposium on Spatial Data Handling*, IGU, Zurich, Suisse, pp. 393-404, 1990.
- [Mainguenaud 1994] Mainguenaud, M., Consistency of geographical information system results, *Computer Environment and Urban Systems*, Vol. 18, pp. 333-342, 1994.
- [Mainguenaud 1995] Mainguenaud, M., The modelling of the geographic information system network component, *International Journal of Geographical Information Systems*, 9 (6), Taylor et Francis, pp. 575-593, 1995.
- [Mainguenaud 1997] Mainguenaud, M., Constraint-based queries in a geographical database for network facilities, *Computer Environment and Urban Systems*, vol. 20, n. 2, pp. 139-151, 1997.
- [Manola 1986] Manola, F. et Dayal, U., PDM: An object-oriented model data model, *International Workshop on Object-Oriented Databases Systems*, 1986.
- [MapInfo 1994] *MapInfo Motif Reference*, version 3.0, MapInfo Corporation, New-York, 1994.
- [Marble 1984] Marble, D. F., Calkins, H. W et Peuquet, D. J., *Basic Readings in Geographic Information Systems*, SPAD Systems Ltd, 1984.
- [Mark 1986] Mark, D. M. et McGranaghan, M., Effective provision of navigation assistance for drivers: A cognitive science approach, *Auto-Carto Conference*, Londres, vol. 2, pp. 399-408, 1986.
- [Mark 1988] Mark, D. M. et al., Cognitive and Linguistics Aspects of Geographical Space, *NCGIA Report*, Santa Barbara, Californie, 1988.
- [Mark 1990] Mark, D. M. et Frank, A. U., *Experiential and Formals Models of Geographic Space*, NCGIA Technical Report 90-10, 1990.
- [Mark 1991] Mark, D. M. et Frank, A. U., *Cognitive and Linguistic Aspects of Geographic Space*, Kluwer Academic Publishers, NATO ASI series, 1991.
- [Medyckyj-Scott 1994] Medyckyj-Scott, D. et Hearnshaw, H. M. Eds, *Human Factors in Geographical Information Systems*, Behalven Press, Londres, 1994.
- [Morehouse 1989] Morehouse, M. S., The architecture of ARC/INFO, *Auto Carto 9 Conference*, Baltimore, Maryland, USA, 1989.
- [Morton 1966] Morton, G. M., *A Computer-Oriented Geodetic Database and a new Technique in File Sequencing*, Rapport Technique, IBM Corporation, Canada, 1966.
- [Motro 1987] Motro, A., Superviews: Virtual integration of multiple databases, *IEEE Tose*, 13 (7), pp. 785-798, 1987.
- [Müller 1990] Müller, J. C., Rule based generalization: Potentials and impediments, *4th International Symposium on Spatial Data Handling*, Brassel, R. Ed., Zurich, pp. 317-334, 1990.
- [Müller 1995] Müller, J. C., Lagrange, J. P., Weibel, R. et Salg, F., Generalization: State of the art and issues, *GIS and Generalization*, Müller, J. C., Lagrange, J. P. et Weibel, R. Eds., Taylor et Francis, pp. 3-17, 1995.
- [NCGIA 1991] NCGIA Research Initiative 7, *Visualization of Spatial Data Quality*, Scientific Report for the Specialist Meeting, Beard, K. et Buttenfield, B. Eds., Castline, Maine, 1991.
- [NCGIA 1993] NCGIA Research Initiative 3, *Multiple Representations*, Closing Report, Buttenfield, B. Eds., State University of New York at Buffalo, 1993.
- [Nievergelt 1984] Nievergelt, J., Hinterberger, H. et Sevcik, K. C., The Grid File: An adaptable, symmetric multikey file structure, *ACM Transactions on Database Systems*, 9, 1984, pp. 38-71.

- [Nyerges 1991] Nyerges, T. L., Representing geographical meaning, *Map Generalization*, Buttenfield B. P. et Mac Master R. B. Eds., Longman, Londres, pp. 59-85, 1991.
- [Ooi 1989] Ooi, B. C., Sacks-Davis, R., McDonell, K. J., Extending a DBMS for geographic application, *5th International Conference on Data Engineering*, pp. 590-597, 1989.
- [Ooi 1990] Ooi, B. C., *Efficient Query Processing in Geographic Information Systems*, Springer-Verlag, 1990.
- [OPENGIS 1996] *The OpenGIS Guide, Introduction to Interoperable Geoprocessing*, Open GIS Consortium, Inc., Buehler, K. et McKee, L. Eds., Massachusetts, USA, 1996.
- [Orenstein 1988] Orenstein, J. A. et Manola, F.A., PROBE spatial data modeling and query Processing in an image database application, *IEEE Transactions on Software Engineering*, 14 (6), pp. 611-629, 1988.
- [Palmer 1988] Palmer, B. L. et Frank, A. U., Spatial Languages, *IIIth International Symposium on Spatial Data Handling*, IGU, Australie, pp. 201-209, 1988.
- [Pau 1989] Pau, L. F., *Survey, mapping and spatial modelling for navigation*, Pau, L. F. Ed., Springer-Verlag, Fano, Danemark, pp. 1-9, 1989.
- [Peuquet 1977] Peuquet, D. J., *Raster Data Handling in Geographic Information Systems*, SUNY Buffalo Geographic Information Systems Laboratory, State University of New York, Buffalo, 1977.
- [Peuquet 1984] Peuquet, D. J., A conceptual framework and comparison of spatial data models, *Cartographica*, 21 (4), pp. 66-113, 1984.
- [Peuquet 1994] Peuquet, D. J., It's about time: A conceptual framework for the representation of temporal dynamics in geographic information systems, *Annals of the Association of the American Geographers*, 84 (3), pp. 441-461, 1994.
- [Peuquet 1995] Peuquet, D. J. et Duan, N., An event-based spatiotemporal data model (ESTDM) for temporal analysis of geographical data, *International Journal of Geographical Information Systems*, 9 (1), pp. 7-24, 1995.
- [Poiker 1975] Poiker, T. K. et Chrisman, N. R., Cartographic Data Structure, *The American Cartographer*, 2-1, pp. 55-69, 1975.
- [Pullar 1988] Pullar, D. V. et Egenhofer, M. J., Towards the defaction and use of topological relations among spatial objects, *IIIth International Symposium on Spatial Data Handling*, Marble, D. F. Ed., IGU, Sydney, pp. 211-224, 1988.
- [Puppo 1995] Puppo, E. et Detorri, G., Towards a formal model for multiresolution spatial maps, *Advances in Spatial Databases*, Egenhofer, M. J. , et Herrings, J. R. Eds., Springer-Verlag, Portland, Maine, pp. 152-169, 1995.
- [Radermacher 1991] Radermacher, F. J., The importance of metaknowledge for environmental information systems, *Advances in Spatial Databases*, Günther, O. et Schek, H.-J. Eds., Springer-Verlag, Zurich, pp. 35-44, 1991.
- [Ramirez 1993] Ramirez, J. R., Development of a cartographic language, *Spatial Information Theory*, Frank, A. U. et Campari, I. Eds., Springer-Verlag, pp. 92-112, 1993.
- [Raper 1989] Raper, J. F. Ed., *Three Dimensional Applications in Geographical Information Systems*, Taylor et Francis, Londres, 1989.
- [Raper 1991] Raper J. et Bundock, M. S., UGIX: A layer based model for a GIS user interface, *Cognitive and Linguistic Aspects of Geographic Space*, Mark, D. M. et Frank, A. U. Eds, Kluwer Academic, pp. 449-475, 1991.
- [Rennison 1995] Rennison, E. et Strausfeld, L., 1995, The Millennium project, *Spatial Information theory: A theoretical basis for GIS*, Frank, A. U. et Kuhn, W. Eds., Springer-Verlag, Semmering, Autriche, pp. 69-91, 1995.
- [Rhind 1990] Rhind, D. W., 1995, Global databases and GIS, *The Association for Geographic Information Yearbook*, Taylor et Francis, Londres, pp. 218-223, 1990.
- [Rigaux 1994] Rigaux, P. et Scholl, M., La représentation multiple dans les bases de données géographiques, *Revue Internationale de Géomatique*, Vol. 4, n. 2, Hermès, Paris, pp. 137-164, 1994.
- [Roussopoulos 1984] Roussopoulos, N. et Leifker, D., An introduction to PSQL: A pictorial structural query language, *IEEE Workshop on Visual Language*, pp. 77-87, 1984.
- [Rundesteiner 1992] Rundesteiner, E. A., Multiview: A methodology for supporting multiple views in object oriented databases, *18th VLDB International Conference on Very Large Databases*, Vancouver, Canada, 1992.

- [Samet 1990] Samet, H., *The Design and Analysis of Spatial Data Structures*, Addison Wesley, New-York, 1990.
- [dos Santos 1995] dos Dantos, S. C., Design and implementation of object-oriented views, *6th Database and Expert Systems Applications Conference*, Revell, N. et Min Tjoa, A. Eds., Springer-Verlag, Londres, pp. 91-102, 1995.
- [Scholl 1989] Scholl, M. et Voisard, A., Thematic map modelling, *1th International Symposium on Large Spatial Databases*, Santa-Barbara, USA, Springer-Verlag, pp. 167-190, 1989.
- [Scholl 1991] Scholl, M., Laasch, C. et Tresch, M., Updatable views in object oriented databases, *2nd International DOOD Conference on Deductive and Object-Oriented Databases*, Munich, 1991.
- [Scholl 1992] Scholl, M. et Voisard, A., Object-oriented databases systems for geographic applications: An experiment with O2. *The O2 Book*, Bancilhon, F., Delobel, C. et Kanellakis, P. Eds., Morgan Kaufman, San Mateo, Californie, 1992.
- [Schrefl 1984] Schrefl, M., Tjoa, A. et Wagner, R., Comparison criteria for semantic data models, *IEEE Conference on Data Engineering*, Los Angeles, pp. 120-125, 1984.
- [Scott 1987] Scott, D. J., *Mental Imagery and Visualization: Their Role in Map Use*, PhD Thesis, Department of Geography, School of Economy, Londres , 1987.
- [Shannon 1949] Shannon, C. F. et Weaver, W., *The Mathematical Theory of Communication*, Urbana, University of Illinois Press., 1949.
- [Smith 1987] Smith, T. R., Menon, S., Star, J. L. et Estes, J. L., Requirements and principles for the implementation and construction of large-scale GIS, *International Journal of Geographical Information Systems*, Taylor et Francis, 1 (1), pp. 13-31, 1987.
- [Staes 1990] Staes, F., Laenens, E. et Tarantino, L., Towards a flexible user interface for knowledge bases, *IEEE Workshop on Visual Languages*, Skokie, Illinois, pp. 143-148. 1990.
- [Stemple 1986] Stemple, D., Sheard, T. et Bunker. R., Abstract data types in databases: Specification, manipulation and access, *Data Engineering Conference*, Los Angeles, pp. 590-597, 1986.
- [Stevens 1946] Stevens, S., On the theory of scales of measurement, *Science Magazine*, 103, pp. 677-680, 1986.
- [Stonebraker 1993a] Stonebraker, M., Chen J., Nathan, N., Paxson, C. et Wu, J., Tioga: Providing data management support for scientific visualization applications, *Very Large DataBase Conference*, Dublin, Irlande, pp. 25-38, 1993.
- [Stonebraker 1993b] Stonebraker, M., The SEQUOIA 2000 project, *Advances in Spatial Databases*, Abel, D. J. et Ooi, B. C. Eds., Springer-Verlag, Singapour, pp. 397-412, 1993.
- [Svensson 1991] Svensson, P. et Huang, Z., Geo-SAL: A query language for spatial data analysis, *Advances in Spatial Databases*, Günther, O. et Schek, H.-J. Eds., Springer-Verlag, Zurich, pp. 119-142, 1991.
- [Tanaka 1988a] Tanaka, M. et Ichikawa, T., A visual user interface for map information retrieval based on semantic significance, *IEEE Transaction on Software Engineering*, 14 (6), pp. 666-670, 1988.
- [Tanaka 1988b] Tanaka, K., Yoshikawa, M. et Ishihara, K., Schema virtualization in object-oriented databases, *4th IEEE International Conference on Data Engineering*, Los Angeles, pp. 23-30, 1988.
- [Thériault 1996] Thériault, M., *Systèmes d'Information Géographique: Concepts fondamentaux*, LATIG, Département de Géographie, Université Laval, Québec, Canada, 1996.
- [Thorndyke 1993] Thorndyke, P. W. et Hayes-Roth, B., *Differences in Spatial Knowledge Acquired from Maps and Navigation*, Office of Naval Research, Santa Monica, USA, 1993.
- [Timpf 1995] Timpf, S. et Frank, A. U., A multi-scale DAG for cartographic objects, *Auto-Carto XII Conference*, pp. 157-163, 1995.
- [Tomlin 1990] Tomlin, C. D., *Geographic Information Systems and Cartographic Modelling*, Prentice-Hall, Englewood Cliffs, New Jersey, 1991.
- [Tverski 1993] Tverskiy, B., Cognitive maps, cognitive collages, and spatial mental models, *Theories and Methods of Spatio-temporal Reasoning in Geographic Space*, Frank, A. U., Campari, I et Formentini, U. Eds., Springer-Verlag, Ile d'Elbe, Italie, pp. 14-24, 1993.
- [Tydac 1993], *SPANS GIS Reference Manual*, Intera Tydac technologies Inc, Ontario, Canada, 1993.

- [USGS 1991] *U.S. Geological Survey Spatial Data Transfer Standard*, Rapport Technique, 1991.
- [Van Oosterom 1991] Van Oosterom, P. J. M., Building a GIS on the top of the open DBMS POSTGRES, *EGIS Conference*, Bruxelles, pp. 775-787, 1991.
- [Van Oosterom 1993] Van Oosterom, P. J. M., *Reactive Data Structures for Geographic Information Systems*, Oxford: Oxford University Press, 1993.
- [Vidale 1993] Vidale, L., Claramunt, C., de Sède, M.-H. et Prélaz-Droux, R., Environmental analysis and GIS: The importance of meta-information, *9th AM/FM International European Conference*, Strasbourg, France, pp. 283-286, 1993.
- [Vijbrief 1992] Vijbrief, T. et Van Oosterom, P. J. M., The GEO++ system: An extensible GIS, *5th International Symposium on Spatial Data Handling*, Charleston, pp. 40-50, 1992.
- [VISION 1995] *Vision Administrator*, Unix Reference, SystemHouse Inc., 1995.
- [Voisard 1991] Voisard, A., Towards a toolbox for geographical user interfaces, *Advances in Spatial Databases*, Günther, O. et Schek, H.-J. Eds., Springer-Verlag, Zurich, pp. 75-98, 1991.
- [Waugh 1987] Waugh, T. C. et Healey, R. G., The Geoview design, a relational data base approach to geographical data handling, *International Journal of Geographical Information Systems*, vol. 1, n. 2, pp. 101-118, 1987.
- [Weinsenbaum 1976] Weinzenbaum, J., *Computer Power and Human Reason*, Freeman, San Francisco, 1976.
- [Wilson 1990] Wilson, P. M., Get your desktop metaphor off my drafting table: User interface design for spatial data handling, *4th International Symposium on Spatial Data Handling*, IGU, Zurich, pp. 455-462, 1990.
- [Wood 1968] Wood, J., Visual perception and map design, *Cartographic Journal*, 5 (1), pp. 54-64, 1968.
- [Wood 1972] Wood, J., Human Factors in Cartographic Communication, *Cartographic Journal*, 9 (2), pp. 123-132, 1972.
- [Wood 1993] Wood, T. et Venters, N., Meeting the demand for different views of a consistently referenced digital road network, *GIS'93 Conference*, Blenheim Online Ed., Birmingham, Royaume-Uni, pp. 47-56, 1993.
- [Woodruff 1995] Woodruff, A. et al., Zooming and tunneling in Tioga: Supporting navigation in multidimensional space, *Visual Database System Conference*, Spaccapietra, S. and Jain, R. Eds., Chapman et Hall, Lausanne, Suisse, pp. 323-332, 1995.
- [Worboys 1994] Worboys, M., A unified model of spatial and temporal information, *Computer Journal*, 37 (1), 1994, pp. 26-34, 1994.
- [Worboys 1995] Worboys, M., *GIS: A Computing Perspective*, Taylor et Francis, Londres, 1995.
- [Ye 1995] Ye, X., Parent, C. et Spaccapietra, S., On the specification of views in DOOD system, *Deductive Object Oriented Databases DOOD'95*, Singapour, pp. 539-556, 1995.