Université de Marne-la-Vallée
Institut géographique national

THÈSE
pour obtenir le grade de
docteur de l'Université de Marne-la-Vallée

Spécialité : Informatique

présentée et soutenue publiquement par

Nils Gesbert

le 2 décembre 2005

Étude de la formalisation des spécifications de bases de données géographiques en vue de leur intégration

Study of the formalization of geographical database specifications for their integration

Directeur de thèse
Thérèse Libourel

Jury de soutenance :
Nadine Cullot, maître de conférences, HDR . . . . . rapporteur
Anne Doucet, professeur des universités . . . . . rapporteur
Gabriella Salzano, maître de conférences . . . . . examinateur
Sébastien Mustière, ingénieur IGN, docteur . . . . . encadrant
Thérèse Libourel, maître de conférences, HDR . . . . . directeur de thèse
Véronique Benzaken, professeur des universités . . . . . président du jury

Rapporteur absent du jury de soutenance : Robert Laurini, professeur des universités

Remerciements

Ce travail a été réalisé au laboratoire COGIT de l'IGN et je remercie l'ensemble de ses membres, avec qui j'ai beaucoup aimé travailler pendant mes quatre années de thèse. Je remercie plus particulièrement Sébastien, notamment pour tout l'encadrement et l'accompagnement de mon travail et pour sa disponibilité ; David, avec qui nous avons beaucoup travaillé ensemble surtout au début, et qui a largement contribué à délimiter le sujet de cette thèse ; et les autres membres de l'équipe UNIBA, BDMUL, INTGR, MULTI ou autre code équivalent.

Je remercie également Anne, pour son encadrement au tout début de la thèse, pour ses relectures et ses commentaires constructifs, et aussi pour son aide à l'organisation de la soutenance.

Merci aussi à Thérèse qui, depuis Montpellier et malgré la distance, a suivi et dirigé mon travail pendant ces quatre années.

Je remercie enfin les rapporteurs et les membres du jury pour l'intérêt qu'ils ont porté à mes travaux.

Table des matières

Chapitre 1

Contexte et problématique

En cet Empire, l'Art de la Cartographie fut poussé à une telle Perfection que la Carte d'une seule Province occupait toute une Ville et la Carte de l'Empire toute une Province. Avec le temps, ces Cartes Démesurées cessèrent de donner satisfaction et les Collèges de Cartographes levèrent une Carte de l'Empire qui avait le Format de l'Empire et qui coïncidait avec lui point par point. Moins passionnées pour l'Étude de la Cartographie, les Générations Suivantes réfléchirent que cette Carte Dilatée était inutile et, non sans impiété, elles l'abandonnèrent à l'Inclémence du Soleil et des Hivers. Dans les Déserts de l'Ouest subsistent des Ruines très abîmées de la Carte. Des Animaux et des Mendiants les habitent. Dans tout le Pays, il n'y a plus d'autre trace des Disciplines Géographiques.

Jorge Luis Borges

Où l'on décrira le domaine de l'information géographique et ses spécificités : tout d'abord, les entités géographiques sont généralement davantage que de simples parties de l'espace, et davantage également que des objets ayant simplement une position. Une entité géographique est à la fois un lieu et ce qui se trouve en ce lieu, et l'on ne peut en général pas considérer ces deux choses indépendamment.

D'autre part, l'espace géographique et les entités qui le composent peuvent être perçus de différentes façons, et en particulier à différentes échelles : les entités représentées sur un plan de ville au 1:10 000 et celles représentées sur un planisphère sont complètement différentes. Cependant, malgré la diversité des perceptions et des représentations possibles d'un même espace, la réalité du terrain reste la même.

Une particularité importante des bases de données géographiques par rapport aux bases de données « classiques » est leur non-exhaustivité intrinsèque : il est possible de recenser la totalité des clients d'une entreprise ou des livres d'une bibliothèque, mais on ne peut faire une liste exhaustive de l'ensemble des éléments constitutifs d'un paysage. La question de la sélection des éléments les plus caractéristiques lors de la constitution de la base de données se pose donc. Pour répondre à cette question en minimisant les ambiguïtés et les possibilités d'interprétations divergentes, les bases de données topographiques disposent de spécifications détaillées et volumineuses qui décrivent le contenu théorique de la base de données (« terrain nominal ») en fonction du terrain réel. Cette particularité se traduit par une différence notable dans le processus de constitution de ces bases de données par rapport aux bases de données classiques.

La diversité des représentations possibles d'un même espace géographique conduit à l'existence de bases de données également diverses. Typiquement, ces bases de données ont été constituées de façon relativement indépendante les unes des autres, dans des buts différents et par des personnes, voire des organismes, différents. Néanmoins, une grande partie de l'information qu'elles contiennent est identique ou directement liée; l'indépendance des différentes bases pose alors plusieurs problèmes, en particulier des problèmes de cohérence et de mise à jour. Ces problèmes poussent à vouloir intégrer ces bases de données. L'intégration consiste, dans un premier temps, à déterminer explicitement les liens entre les diverses représentations du monde à intégrer, c'est-à-dire à décrire quels types d'objets dans les différentes bases de données sont susceptibles de participer à la représentation d'une même entité du monde réel, dans quelles circonstances cela se produit et dans quelle mesure exactement ces représentations sont liées.

Afin de déterminer ces correspondances, il est nécessaire de s'appuyer sur la connaissance précise des relations entre terrain réel et représentation dans chacune des bases; autrement dit, sur les spécifications. Mais ces spécifications sont actuellement de volumineux documents sous forme textuelle, adaptés avant tout à l'utilisation pour la saisie des données ou à la consultation ponctuelle pour lever un doute sur un point précis. Nous proposons dans ce mémoire de représenter l'information qu'elles contiennent d'une façon plus formelle, permettant par la suite de faciliter la comparaison entre deux jeux de spécifications et donc la détermination des correspondances entre bases pour l'intégration.

1.1 L'information géographique et sa représentation

L'espace géographique est une réalité beaucoup trop complexe pour pouvoir être appréhendée dans son intégralité et son détail par un observateur humain ; cet observateur travaille sur une représentation simplifiée de cet espace, représentation dont les détails qui ne l'intéressent pas sont absents [Timpf, 1999]. Cette représentation mentale diffère selon la personne, selon sa culture de l'information géographique : a-t-elle l'habitude de manipuler des cartes ? voyage-t-elle souvent ? à quel type de paysage est-elle habituée ? etc., et selon la tâche à accomplir à l'aide de cette représentation : se repérer ? trouver le plus court chemin d'un point à un autre ? déterminer le meilleur endroit pour une certaine plantation ?

On peut considérer de façon abstraite que l'espace géographique est constitué d'*entités géographiques* qui correspondent aux concepts géographiques que l'on manie mentalement ; ainsi, les mers, les lacs, les montagnes, les bâtiments, les prés, les chemins, les déserts, les États, les continents, les haies, par exemple, sont autant d'entités géographiques. Ces entités sont de natures diverses [Smith et Mark, 1998] : elles peuvent être concrètes et bien définies, comme les bâtiments ou les lacs ; elles peuvent être bien réelles tout en étant difficiles à délimiter précisément, ainsi les golfes et les baies ; elles peuvent être uniquement le résultat de décisions humaines arbitraires, telles les limites administratives.

Toute représentation géométrique d'une entité géographique se fait à une certaine échelle. Certains concepts géographiques n'ont pas de représentation possible à des échelles trop grandes ou trop petites. Ainsi, par exemple, les frontières d'une forêt, d'une agglomération ou d'un massif montagneux ne sont pas représentables au-delà d'une certaine échelle, car ces entités sont mésoscopiques, c'est-à-dire qu'elles ne correspondent pas directement à des objets physiques atomiques et bien identifiables (à l'échelle humaine) comme pourrait l'être un bâtiment, mais plutôt à un type de paysage ; ainsi, à grande échelle, leur frontière n'est pas bien déterminée. Il est à noter qu'à *petite* échelle, ces entités « mésoscopiques » sont tout aussi réelles et bien délimitées physiquement que le sont les bâtiments à grande échelle et deviennent à leur tour atomiques : personne n'aurait l'idée, par exemple, de découper une agglomération en îlots urbains sur un planisphère à l'échelle 1:50 000 000. Inversement, à très grande échelle, celle d'un plan détaillé d'architecte par exemple, un bâtiment peut devenir une entité mésoscopique et une agglomération une entité qui n'a plus aucun sens.

On peut en fait considérer de façon générale, si l'on ne se limite pas à la géographie, que toute entité est trop mal délimitée pour avoir un sens à trop grande échelle et trop petite pour être représentée en tant que telle à trop petite échelle. Cependant, on n'utilise pas dans le domaine de l'information géographique d'échelles supérieures à celle d'un plan de ville détaillé ou inférieures à celle d'un petit planisphère, et certaines entités géographiques, telles que les grands fleuves, sont représentables à toute échelle. Mais leur représentation est différente : à grande échelle, un fleuve est une surface complexe, avec des niveaux de plus hautes et plus basses eaux ; à moyenne échelle, il peut n'être représenté que par un ensemble de bras d'eau ; à petite échelle, les îles sont invisibles et les bras se confondent en une seule ligne simple.

1.2 Les bases de données géographiques

Nous nous intéressons dans ce mémoire aux bases de données topographiques de type vecteur, telles que celles produites par l'Institut géographique national. Ces bases sont construites sur un modèle orienté objet ; elles disposent d'un schéma conceptuel dont les classes correspondent aux différents types d'entités représentés. Citons l'introduction des spécifications de l'une de ces bases, la BDCarto :

L'information est décomposée en entités géographiques et en liens entre ces entités, choisis pour leur importance vis-à-vis des besoins auxquels répond la BDCarto. [...]

Les entités géographiques sont regroupées en thèmes. Chaque thème correspond à un centre d'intérêt particulier (réseau routier, découpage administratif, hydrographie...) [...] Chaque thème est constitué :

- *d'objets sémantiques, chacun étant identifié dans la BDCarto par un identifiant (ex : un tronçon de route, une commune,..)*
- *et de liens entre objets sémantiques du thème ou inter-thèmes.*

Les objets sémantiques sont caractérisés par des attributs (largeur d'un tronçon de route, nom d'une commune, navigabilité d'un tronçon de cours d'eau...); ils sont regroupés en classes d'objets, qui sont des familles d'objets possédant les mêmes attributs. Les classes forment une partition de l'ensemble des objets contenus dans la base de données (un objet appartient à une classe et une seule).

Nous nous intéressons donc aux bases de données possédant ce type de structure, qu'on qualifie de bases de données géographiques *à objets*, par opposition aux bases de données géographiques *à champs continus* qui, davantage que des *entités* géographiques, représentent des phénomènes géographiques tels que par exemple le relief. Un tel phénomène peut être décrit en affectant une valeur (en l'occurrence l'altitude) à chaque point de la zone considérée. D'autre part, nous avons utilisé le terme de base de données *topographique*. Nous entendons par là que la base de données vise à décrire les éléments du *terrain*, de l'espace topographique ; ce n'est pas un terme consacré mais le terme de base de données géographique, plus vague, pourrait englober également des bases de données thématiques qui ne font pas partie de nos objets d'étude.

Typiquement, le schéma d'une base de données topographique à objets reproduit des classifications cartographiques, pour deux raisons : d'une part parce que ces bases de données sont un nouveau moyen de représenter l'espace topographique, et en tant que tel ont hérité de l'expérience des siècles passés en matière de représentation de l'espace topographique, donc de la cartographie. D'autre part parce que ces bases de données sont conçues entre autres dans le but de réaliser des cartes. Ainsi par exemple, à l'heure actuelle, dans la plupart des bases de données, les classes ne regroupent que des entités ayant le même type de représentation géométrique (point, ligne, ou surface).

Le contenu d'une base de données topographique, quant à lui, est une représentation d'une certaine étendue de l'espace topographique ; en général, cette représentation est prévue pour une échelle (ou une plage d'échelles limitée) donnée, même si ce n'est pas une nécessité. Il faut noter que la base de données n'étant pas en soi une représentation graphique de cet espace, elle n'a pas d'échelle à proprement parler ; néanmoins, la représentation des entités est généralement faite à un *niveau de détail*

géométrique donné, ce qui correspond à une certaine échelle. Dans certaines bases de données à vocation spécialisées, les différents thèmes peuvent être représentés à des niveaux de détail différents. Par exemple, une base de donnée orientée vers la circulation automobile peut représenter le réseau routier avec un niveau de détail très élévé (correspondant à celui d'un plan de ville) et ne mentionner les zones boisées qu'à titre indicatif, avec une précision de l'ordre de la carte au 1:100 000. Il est théoriquement possible également d'avoir plusieurs représentations géométriques d'une même entité à des niveaux de détail différents, dans la même base de données, si elle doit être très polyvalente ; mais actuellement, du moins chez les gros producteurs de données géographiques tels que l'IGN, on rencontre peu ce type de base de données multi-échelles dans la pratique : cela reste du domaine de la recherche.

On rencontre en revanche un autre type de représentation multiple, lié aux contraintes mentionnées plus haut sur la géométrie des objets (point, ligne ou surface) : il arrive qu'une même entité soit représentée par plusieurs objets différents (pour un unique niveau de détail), chacun d'entre eux contenant des informations sur un aspect particulier de l'entité. Ainsi, une rivière, lorsqu'elle dépasse une certaine largeur, est souvent représentée à la fois par un objet surfacique qui indique l'étendue spatiale précise du cours d'eau et par un objet linéaire, représentant l'axe principal et le sens de l'écoulement, qui permet de disposer d'une représentation topologique homogène de l'ensemble du réseau hydrographique. De même, un bâtiment couvrant une certaine surface au sol et muni par ailleurs d'une tour ou cheminée très haute peut se trouver représenté à la fois par un objet surfacique indiquant son contour en planimétrie et par un objet ponctuel indiquant l'emplacement précis et l'altitude du point culminant.

Spécificités des bases de données géographiques

Les bases de données géographiques telles que nous les considérons ici ont la particularité remarquable, par rapport à la plupart des bases de données classiques, de ne pas représenter exhaustivement leur domaine. En effet, si une base de données de bibliothèque possède une table des exemplaires et une table des lecteurs, cela signifie a priori qu'elle doit recenser la totalité des exemplaires et des lecteurs de la bibliothèque ; si une base de données d'entreprise possède une table des employés et une table des clients, cela suppose là encore qu'elle recense en principe tous les employés et tous les clients de l'entreprise. Ce type de base peut avoir des défauts d'exhaustivité accidentels, dus par exemple à un manque d'information ou à un oubli de saisie, mais n'est pas spécifiquement conçu pour ne représenter qu'une partie de son domaine. Il en va différemment des bases de données géographiques. En effet, pour la plupart des types d'entités géographiques, l'exhaustivité est tout simplement impossible, non à cause d'un manque d'information mais à cause de la surabondance d'information. Nous avons mentionné plus haut qu'une représentation de l'espace géographique, pour être utile et maniable, ne peut représenter cet espace dans toute sa complexité ; une part de la simplification opérée consiste à abstraire cet espace en un ensemble d'entités géographiques de différents types. Une autre part consiste à ne considérer que certains types d'entités. Une troisième part consiste à ne représenter que certains aspects de ces entités et à simplifier leur géométrie. Mais cela ne suffit pas à réduire la quantité d'information à un niveau utilisable : il est encore nécessaire

d'opérer une sélection parmi les entités de la plupart des types afin de ne conserver que les plus importantes ou les plus caractéristiques. Ainsi, si le schéma conceptuel d'une base de données géographique comprend une classe « sentier », cela ne signifie pas que cette base représente tous les sentiers de la zone qu'elle représente : ils sont généralement trop nombreux et certains sont insignifiants ; les représenter tous rendrait d'une part trop importants les délais de constitution et de mise à jour de la base de données, et d'autre part risquerait de noyer l'information utile dans le bruit. Cela signifie en fait qu'elle représente ceux des sentiers qui respectent un certain nombre de critères d'importance ou de représentativité justifiant leur inclusion dans la base.

Ces critères, appelés critères de *sélection*, sont généralement relativement complexes et issus de l'expertise cartographique. En effet, si l'on rencontre fréquemment des critères de taille par exemple (on ne représente pas les bâtiments trop petits), la notion qui entre en jeu dans cette sélection n'est pas celle d'importance absolue mais celle, plus ou moins abstraite et subjective, de trait caractéristique du paysage. Ainsi, un bâtiment isolé peut constituer un point de repère important, même s'il s'agit d'une construction de taille réduite ; en revanche, il n'est pas très utile de représenter individuellement tous les étangs et les canaux d'une zone marécageuse, même s'ils sont de taille importante. Mais l'interprétation de ce qui est ou non significatif fait appel au jugement et à l'expérience de l'opérateur qui saisit les données, ce qui implique une certaine liberté et donc de possibles variations d'un opérateur à l'autre, même si, dans le cas de l'IGN, ces variations sont limitées par l'existence d'un savoir-faire d'entreprise commun. Afin de guider ces choix subjectifs et de garantir un minimum d'homogénéité dans les choix de sélection au sein d'une même base de données, les critères de sélection à utiliser pour chaque classe de chaque base sont détaillés dans les spécifications de la base. Ces volumineux documents textuels indiquent également, pour les entités sélectionnées, comment on détermine à partir de leurs propriétés la géométrie et les attributs des objets créés dans la base pour les représenter (figure 1.1).

L'utilité de ces documents est double : d'une part, lors de la phase de saisie des données, ils permettent d'assurer que les différentes personnes participant à l'opération s'accordent sur les choix à faire. D'autre part, une fois la base de données constituée, ils permettent de documenter les choix qui ont été faits ; ils servent donc par exemple pour la mise à jour : il est nécessaire, pour déterminer comment une modification du terrain doit modifier le contenu de la base de données, de connaître précisément les règles qui ont mené du terrain à la base en premier lieu. Ils peuvent servir à l'utilisateur des données pour lui éviter des erreurs d'interprétation ; par exemple, si une personne a pris comme repère, pour tourner, « la première maison au bord de la rivière » en utilisant une base de données qui ne représente que les bâtiments de plus de 50 m^2 et rencontre tout d'abord un petit abri de taille inférieure à ce seuil, l'ignorance de ce critère de sélection pourrait la conduire à tourner trop tôt. Enfin, si l'on dispose de plusieurs bases de données différentes couvrant un même terrain, ils peuvent permettre de déterminer a priori quels types d'entités sont susceptibles d'être représentés dans les deux bases, sous quelles conditions et la façon dont ces représentations sont liées. Ceci peut alors permettre de déduire de l'étude des données des informations supplémentaires sur le terrain ou sur la qualité des données ; et ceci est nécessaire si l'on souhaite intégrer ces bases de données.

E1
Bâtiment

Type :	Simple	**Attributs** (* voir les spécifications générales) • *Signature électronique** • *Nature* • *Fonction* • *Altitude sol bâtiment* • *Source géométrique des données**
Localisation :	Surfacique tridimensionnelle	
Liens :		

Définition

Bâtiment de plus de 20 m^2.

Regroupement : Voir les différentes valeurs des attributs <nature> et <fonction>.

Sélection

Tous les bâtiments de plus de 50 m2 sont inclus.

Les bâtiments faisant entre 20 et 50 m2 sont sélectionnés en fonction de leur environnement* et de leur aspect**.

Les bâtiments de moins de 20 m2 sont représentés par un objet de classe <construction ponctuelle> s'ils sont très hauts, ou s'ils sont spécifiquement désignés sur la carte au 1 : 25 000 en cours (ex. monument, antenne,…).

* Les petits bâtiments isolés (plus de 100 m d'une habitation) de plus de 20 m2 sont inclus, alors que les petits bâtiments situés en ville ne le sont pas (ex, petit garage individuel, petit atelier, annexes diverses).

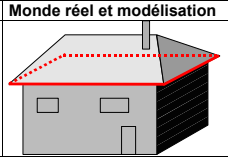
** Les petits bâtiments d'aspect précaire (cabanes de chantier, petits abris pour animaux,…) sont exclus.

Modélisation géométrique

Contour extérieur du bâtiment tel qu'il apparaît vu d'avion (le plus souvent, ce contour correspond à celui du toit); altitude* correspondant à ce contour (généralement l'altitude des gouttières).

* altitude de l'arête supérieure en cas de face verticale.

Seules les cours intérieures de plus de 10 m de large sont représentées par un trou dans la surface bâtie.

Description	Monde réel et modélisation	Modélisation géométrique
Modélisation d'une maison		

Plusieurs bâtiments contigus ou superposés de même « nature » et de même « fonction » sont généralement considérés comme un seul et même objet (seul le contour extérieur est saisi). Deux objets contigus ou superposés sont cependant représentés s'ils présentent les caractéristiques suivantes :

- différence de hauteur entre les deux bâtiments > 10 m environ (ou 3 étages) ;
- surface de chaque objet résultant de 400 m^2 environ ou plus.

FIG. 1.1 *Exemple de spécifications : la classe bâtiment de la BDTopo Pays de l'IGN.*

1.3 Le besoin d'intégration

En raison de la variété des représentations possibles d'un même terrain, suivant qu'on souhaite y mettre en valeur tel ou tel aspect thématique ou tout simplement suivant le niveau de détail souhaité, il existe souvent plusieurs bases de données géographiques différentes couvrant une même zone. Par exemple, l'IGN produit un certain nombre de bases de données couvrant tout ou partie de la France : la BDCarto couvre l'ensemble de la métropole à un niveau de détail correspondant au 1:50 000 environ. La BDTopo, dans son ancienne version dite « standard », était en fait un ensemble de bases de données indépendantes, appelées « feuilles » par référence à la cartographie, couvrant chacune une petite portion rectangulaire du territoire à un niveau de détail correspondant au 1:10 000. On peut cependant parler de « la » BDTopo standard dans la mesure où le schéma conceptuel et les spécifications sont uniques et valables pour l'ensemble des feuilles[1]. La BDTopo standard n'est plus mise à jour mais peut néanmoins encore servir pour certaines applications, en raison des différences de contenu avec la nouvelle BDTopo. La version actuelle de la BDTopo n'utilise plus le découpage en feuilles, mais elle est toujours séparée en plusieurs bases de données distinctes, correspondant maintenant aux départements ; cependant les raccords sont pris en compte dans la constitution de cette nouvelle version. Aucune des versions de la BDTopo ne couvre à l'heure actuelle la totalité du territoire français, ce qui rend d'autant plus indispensable l'existence de bases de données moins détaillées mais complètes (qui couvrent tout le territoire).

Ces bases de données ont généralement été constituées indépendamment les unes des autres, dans des buts et avec des moyens logiciels différents ; c'est le cas à l'IGN. Cet état de fait présente un certain nombre d'inconvénients. Il en présente du point de vue de l'utilisateur : de nombreuses applications peuvent nécessiter l'utilisation simultanée de données en provenance de plusieurs bases différentes, par exemple pour utiliser plusieurs niveaux de détail à la fois, ce qui peut servir notamment pour la recherche d'itinéraire [Car et Frank, 1994]. Or l'indépendance des différentes bases entraîne des risques d'incohérences, dues par exemple aux différences d'actualité des données, ou tout simplement à des erreurs de saisie ou des imprécisions bénignes séparément mais qui peuvent mener à des aberrations lorsqu'on considère l'ensemble des bases. Et même si l'on suppose les données cohérentes, il est nécessaire pour pouvoir exploiter utilement la diversité de l'information de mettre en relation ces représentations différentes d'un même terrain d'une façon ou d'une autre. Typiquement, cette mise en relation peut être faite visuellement, par la simple superposition des différentes représentations dans un SIG, ou même par leur juxtaposition, en comptant sur l'aptitude de l'œil humain à reconnaître les formes pour repérer les éléments qui se correspondent de l'une à l'autre, comme on pourrait le faire avec des cartes. Cependant, un ordinateur ne dispose pas spontanément de cette aptitude à la reconnaissance de formes, or les objets qui se correspondent peuvent être légèrement déformés ou décalés. Cette superposition des représentations peut donc convenir pour des analyses ne nécessitant pas réellement l'utilisation de l'informatique, mais si l'on veut profiter de ces données numériques autrement que comme l'équivalent

1. En réalité, la situation est plus complexe que cela car les spécifications ont été mises à jour régulièrement, les nouvelles feuilles saisies suivant les nouvelles spécifications tandis que les anciennes restaient dans l'ancienne version de la BDTopo jusqu'à la mise à jour suivante.

de cartes scannées et leur appliquer des traitements informatiques exploitant les différents niveaux de détail, on se rend compte que les algorithmes d'appariement mis en jeu pour retrouver automatiquement les données associées n'ont rien de trivial, d'autant moins que les schémas conceptuels des différentes bases de données n'ont pas été conçus pour faciliter cet appariement. Ainsi, dans la situation actuelle, il est pour ainsi dire impossible de réaliser une analyse géographique exploitant toute la richesse des données disponibles et les possibilités de l'informatique sans un travail préalable très important sur les données.

L'indépendance des bases présente aussi un inconvénient important pour le producteur de données : leurs mises à jour doivent également être réalisées indépendamment. Même si les données de base servant à ces mises à jour (informations collectées auprès d'organismes, prises de vues aériennes) peuvent être communes à l'ensemble des bases de données concernées, il n'en reste pas moins que la saisie de ces données sous forme numérique doit être réalisée une fois pour chaque base, ce qui augmente les risques d'incohérences entre ces bases et constitue un effort redondant.

Ces raisons poussent à envisager l'intégration de ces diverses bases de données. Cette intégration peut être plus ou moins poussée et se faire de différentes façons (voir chapitre 2), mais elle nécessite dans tous les cas la détermination des correspondances entre schémas conceptuels, c'est-à-dire qu'il faut trouver tout d'abord quelles classes des différentes bases sont susceptibles de participer aux représentations d'une même entité géographique, et ensuite les conditions auxquelles ces correspondances s'appliquent effectivement.

Une fois les schémas de données ainsi appariés, l'autre partie de l'établissement de correspondances nécessaire à l'intégration est la mise en relation des données elles-mêmes, qui consiste à trouver quels objets des différentes bases de données participent à des représentations de la même entité et doivent donc être reliés. Dans le cas de données géographiques, donc localisées, le critère principal permettant cette détermination est justement cette localisation ; pour cette raison, on parle d'appariement géométrique, même si d'autres critères peuvent également entrer en jeu). Quels que soient le degré et le mode d'intégration choisi pour les bases de données, ces étapes d'appariement interviennent nécessairement à un moment ou à un autre du processus.

L'étape à laquelle nous nous intéressons plus particulièrement dans ce mémoire est celle de l'appariement de schémas.

1.4 Objectif de cette thèse

Pour réaliser cet appariement, il est nécessaire de s'appuyer sur les spécifications des bases de données, qui, seules, contiennent l'information nécessaire à la détermination des correspondances entre les schémas. Or les spécifications n'existent actuellement que sous la forme de volumineux documents textuels. Ces documents sont relativement structurés mais ne sont néanmoins pas exploitables directement par un ordinateur. Notre travail de thèse a consisté à chercher comment représenter l'information contenue dans les spécifications d'une façon autant que possible formelle et homogène d'une base à l'autre.

Au chapitre 2, nous présenterons un état de l'art sur l'intégration de bases de données et plus précisément de bases de données géographiques ; nous nous intéresserons notamment aux travaux s'appuyant sur la sémantique des données pour réaliser l'intégration, cette sémantique pouvant être représentée à l'aide d'*ontologies*, un terme que nous définirons. Ceci nous permettra en particulier de détailler le contexte de notre travail brièvement décrit ici.

Nous proposerons alors au chapitre 3, en nous appuyant sur l'étude des spécifications telles qu'elles existent actuellement, des propositions pour un modèle formel de représentation des spécifications. Nous proposerons notamment de représenter ces spécifications sous forme de règles allant des entités du terrain, structurées par une ontologie, vers les objets de la base, structurés par le schéma de la base, et nous définirons un langage formel permettant d'exprimer ces règles. Nous étudierons l'instanciation du modèle ainsi défini sur les thèmes hydrographiques de deux bases de données de l'IGN, la BDTopo Pays et la BDCarto.

Au chapitre 4, nous présenterons tout d'abord des outils logiciels réalisés afin d'étudier la mise en œuvre pratique du modèle décrit au chapitre précédent. Nous étudierons alors la formalisation complète, à l'aide de notre langage, des spécifications concernant les cours d'eau dans la BDTopo Pays et la BDCarto, avant de conclure.

Chapitre 2

État de l'art

Où l'on mentionnera, en 2.1.1, des travaux de référence relatifs à l'intégration de bases de données hétérogènes et à l'interopérabilité des systèmes d'information ; on verra qu'il s'agit d'une problématique toujours d'actualité, en raison de la complexité du problème et de la prolifération des sources d'information, et cependant déjà ancienne, de sorte que des architectures de référence ont été définies et qu'il en existe d'ores et déjà plusieurs implémentations.

On verra en 2.1.2 que cette problématique est particulièrement importante dans le domaine de l'information géographique, où les données disponibles peuvent être très hétérogènes entre deux zones adjacentes aussi bien que d'un niveau de détail à un autre sur la même zone ; or un certain nombre d'applications nécessite l'utilisation simultanée de telles données hétérogènes. Il est donc souhaitable d'expliciter les liens entre les différentes bases de données, ce qui en faciliterait par ailleurs la maintenance et la mise à jour. De plus, la complexité des bases de données géographiques — ou plus généralement spatio-temporelles — en fait un cas à part nécessitant des solutions spécifiques, telles que des bases de données multi-représentations. Étant donnés le coût et le temps d'acquisition des données, une telle base multi-représentations devrait être le résultat de l'intégration de bases existantes, avec typiquement une architecture de bases de données fédérées. Le processus d'intégration comprend l'appariement des schémas et celui des données.

La partie du processus qui nous intéresse davantage dans ce mémoire est l'appariement de schémas. Cette étape fait très largement appel à la sémantique des données. Plus généralement, on peut noter que la bonne compréhension de la sémantique est une clef de l'interopérabilité (2.2). Cette sémantique peut être en partie représentée sous forme de métadonnées ou d'ontologies, un terme qu'on voit de plus en plus fréquemment et dont les significations peuvent être relativement diverses, en particulier par leur niveau d'abstraction (2.2.1). Beaucoup de travaux mentionnent l'utilité d'ontologies pour représenter les connaissances des experts d'un domaine d'application et faciliter l'intégration et l'interopérabilité. Concrètement, ces ontologies peuvent prendre des formes variées, notamment dans le domaine spatial.

Dans le cas précis qui nous intéresse ici, celui des bases de données topographiques vecteur, on peut considérer l'ensemble des concepts géographiques mentionné au chapitre 1 comme une ontologie du domaine. La source primordiale d'information sur les données d'une base, qui peut être vue comme un ensemble de métadonnées indiquant comment les données représentent le terrain, c'est-à-dire comment le schéma conceptuel de la base dérive de l'ontologie sous-jacente, est l'ensemble des spécifications de cette base (2.2.2). Cependant, cette source d'information n'existe actuellement que sous la forme de documents textuels, peu adaptés à des traitements automatisés. Nous proposons dans le présent mémoire un modèle de représentation des spécifications visant à faciliter leur exploitation.

Définitions

Avant de commencer ce chapitre, rappelons tout d'abord la terminologie classique des bases de données [Gardarin, 1999, chap. 2] : on appelle *modèle de données* un ensemble de notions théoriques permettant la description de bases de données. Ainsi le modèle objet comprend entre autres les notions de classe, d'objet et d'attribut tandis que le modèle relationnel comprend celles de table, de clef primaire et de clef étrangère. On appelle *schéma* d'une base de données la description de sa structure selon un certain modèle de données[1].

Dans un système de gestion de bases de données (SGBD), la *métabase* d'une base de données est une base de données dont la structure est prédéfinie et qui contient diverses informations sur la base de données ; par exemple, dans une base de données relationnelles, il existe une table qui contient la liste des tables. Le terme de *méta-données* désigne de façon beaucoup plus générale toutes données relatives à d'autres données ; il peut par exemple s'agir du schéma d'une base de données, de ses spécifications de contenu, d'informations sur la qualité des données...

2.1 Généralités sur l'intégration de bases de données

2.1.1 Bases de données classiques (non géographiques)

La quantité de données informatiques disponibles d'une façon ou d'une autre ne cesse de croître, dans tous les domaines. Or la création de ces données informatiques, qu'elle soit faite par saisie interactive ou à l'aide d'appareils de mesure, est généralement un processus très coûteux en temps et en moyens. La réutilisation des données existantes est donc un enjeu qui prend de plus en plus d'importance ; cependant, ces données se trouvent souvent distribuées dans des systèmes divers et plus ou moins incompatibles.

On appelle *interopérabilité* la capacité des systèmes informatiques à s'échanger des données de façon cohérente. Le problème de l'interopérabilité est lié à celui de l'*intégration* de données, qui consiste à constituer un tout cohérent à partir de données provenant de plusieurs sources différentes. Nous allons présenter des architectures logicielles de référence qui ont été proposées pour résoudre ce dernier problème.

Architectures de médiation

[Parent et Spaccapietra, 2000] distingue deux niveaux de complexité du problème. Le cas le plus complexe est celui où les données sont peu structurées et où l'on ne dispose que de peu d'information extérieure sur elles : il est alors nécessaire d'analyser leur contenu, par exemple à l'aide d'outils d'intelligence artificielle, pour pouvoir les classer. C'est typiquement le cas des données récupérées sur le web. L'architecture de référence pour la réalisation de systèmes d'information devant utiliser de telles données peu structurées, ou un nombre important et éventuellement variable de sources de données très hétérogènes, fait appel à des *médiateurs* (voir figure 2.1). Le concept de médiateur est défini dans [Wiederhold, 1992], où il est fait une distinction entre les *données*, qui correspondent à ce qu'on peut obtenir directement de sources

1. Il est courant, dans la communauté SIG, d'utiliser le terme de modèle pour désigner le schéma, le modèle étant alors appelé méta-modèle ; pour éviter toute ambiguïté, nous n'utiliserons que la terminologie classique des bases de données.

diverses, pouvant aller des données brutes transmises par un capteur à des données déjà traitées telles que des statistiques, et l'*information*, qui désigne les données interprétées et rendues compréhensibles pour l'utilisateur[1]. L'interprétation des données nécessite des connaissances et un savoir-faire spécifiques, détenus par les experts du domaine concerné et éventuellement par des livres sur le sujet, mais qu'on peut chercher à encoder informatiquement dans des bases de connaissances. On définit alors un médiateur comme un logiciel muni d'une base de connaissances lui permettant d'interpréter des données pour en tirer de l'information. Il est à noter que cette information peut à son tour être considérée comme des données par un système de plus haut niveau.

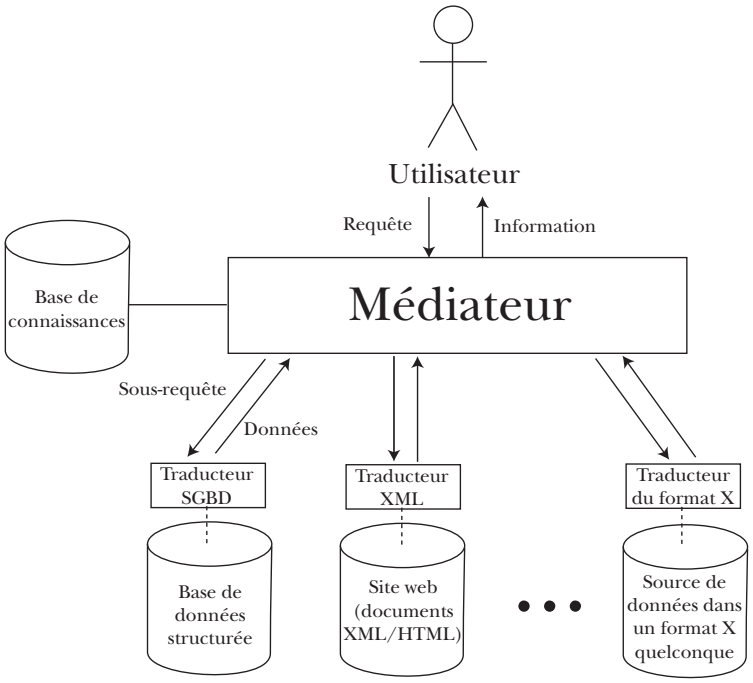


FIG. 2.1 *L'architecture de médiation : l'utilisateur soumet une requête au médiateur qui, à l'aide de sa base de connaissances, va décomposer cette requête en sous-requêtes adressées aux différentes sources de données disponibles, via des traducteurs spécifiques, puis intégrer les données résultantes pour fournir à l'utilisateur l'information demandée.*

De nombreux systèmes à base de médiateurs, se concentrant davantage sur l'un ou l'autre aspect du problème, ont été développés. On peut citer Information Manifold [Kirk *et al.*, 1995], qui permet d'intégrer des données plus ou moins structurées, en profitant de la structure lorsqu'elle existe, suivant une modélisation conceptuelle uniforme correspondant au domaine d'application choisi. La base de connaissances du système contient des descriptions formelles des contenus et des propriétés des différentes sources de données, écrites à l'aide de logiques de description et de règles de Horn [Levy et Rousset, 1996]. Elle permet de décomposer une requête de l'utilisateur en sous-requêtes adressées aux différentes sources de données avant d'intégrer

1. Nous utilisons dans le reste du mémoire le terme « information » dans son sens plus général usuel et non dans ce sens spécifique.

les réponses de ces sources pour obtenir le résultat renvoyé à l'utilisateur. Le système PICSEL [LRI, 2002], développé par le laboratoire de recherche en informatique d'Orsay, est similaire. Dans ces systèmes, la représentation globale du domaine d'application est fixe et unique et sert de référence, afin que l'utilisateur ait l'impression de manipuler une source de données unique. Les sources de données, quant à elles, sont décrites pour pouvoir être utilisées relativement à cette représentation globale ; en d'autres termes, les schémas locaux sont décrits en fonction du schéma intégré. Dans le cas de bases de données relationnelles, cette description peut être faite par le mécanisme des *vues* ([Gardarin, 1999, chap. 9]). Pour cette raison, l'approche utilisée par ces systèmes est appelée *Local as views* : les schémas locaux sont des vues.

D'autres systèmes, comme HERMES [Subrahmanian *et al.*, 1996], Garlic [Cody *et al.*, 1995] ou TSIMMIS [Chawathe *et al.*, 1994 ; Garcia-Molina *et al.*, 1997], utilisent l'approche *Global as views*, c'est-à-dire que la représentation *globale* de l'information est constituée par l'intégration de *vues* de chacune des sources de données, ces vues étant obtenues par l'intermédiaire de traducteurs (« wrappers ») qui peuvent être plus ou moins complexes. HERMES et TSIMMIS sont des systèmes modulaires pouvant utiliser plusieurs médiateurs et proposent des environnements logiciels et des langages déclaratifs en permettant la programmation rapide. Ainsi l'ajout d'une nouvelle source de données, au lieu de se traduire par l'ajout d'une description formelle dans une base de connaissances comme pour les systèmes Information Manifold et PICSEL, se traduit par la programmation de nouveaux traducteurs et médiateurs, et la représentation globale de l'information est modifiée.

Architectures de fédération

Le cas moins complexe est celui de l'intégration de bases de données structurées, où l'existence des schémas de données et des métadonnées permet d'aller plus loin dans l'intégration.

Si l'on souhaite malgré tout conserver l'autonomie des bases de données de départ, et en particulier la possibilité d'accéder directement à chacune d'entre elles et de continuer à les utiliser comme avant l'intégration, l'architecture de référence est celle d'un système de *bases de données fédérées* décrite par [Sheth et Larson, 1990] (figure 2.2). Un tel système peut être faiblement ou fortement couplé ; nous nous intéresserons uniquement au cas fortement couplé. Chaque base de données dispose d'un schéma *local* exprimé dans le modèle correspondant au SGBD (système de gestion de bases de données) qui la gère. On choisit en vue de la fédération un modèle commun, si possible plus riche que les modèles locaux, afin de pouvoir traduire tous les schémas locaux dans ce modèle commun sans perte d'information, obtenant ainsi des schémas *composants*. Éventuellement, les bases de données locales peuvent ne faire participer qu'une partie de leur contenu à la fédération en définissant un schéma d'*exportation*, sous-ensemble du schéma composant. Le schéma *fédéré* correspond alors à l'intégration des différents schémas d'exportation ; des correspondances sont définies entre ce schéma et chacun des schémas d'exportation afin de permettre l'accès aux données via le schéma fédéré. Ces correspondances doivent inclure un moyen de faire le lien entre les données des différentes bases correspondant à des représentations d'une même entité réelle. Ce lien est nécessaire pour traiter les requêtes, car une requête peut faire appel simultanément à des informations en provenance des deux bases mais concernant une même entité. Il est également nécessaire pour

la cohérence, car une information peut être présente dans les deux bases à la fois ; il faut donc s'assurer qu'elle contient bien la même chose dans les deux cas. Nous verrons dans le paragraphe suivant que dans le cas des données géographiques cette détermination des correspondances entre instances (ou appariement de données) est un problème à part entière.

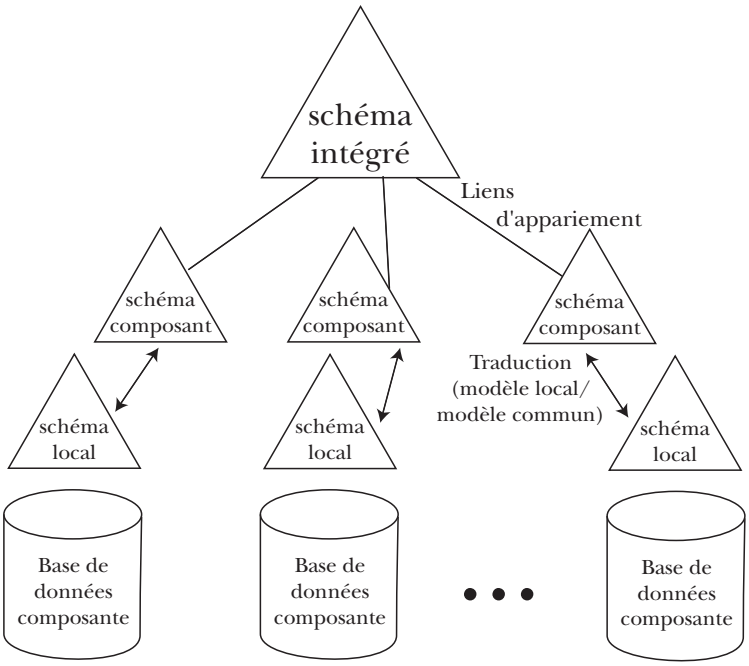


FIG. 2.2 *L'architecture de fédération : l'utilisateur peut accéder à la fédération des bases via le schéma intégré comme s'il s'agissait d'une unique base de données ; les éléments du schéma intégré sont reliés aux éléments correspondants des schémas composants par des liens d'appariement pré-établis.*

La médiation et la fédération présentent des points communs et certains systèmes peuvent être considérés comme relevant simultanément de l'une et de l'autre ; cependant, tandis que la médiation traite en général de la recherche et de l'interprétation d'information provenant de sources éventuellement nombreuses ou peu structurées et vise à être employée au sein d'un système d'information, la fédération consiste à permettre non seulement l'interrogation mais aussi la gestion intégrée d'un nombre fixe de bases de données structurées et son interface est celle d'un SGBD. En particulier, un système de bases de données fédérées peut permettre la mise à jour des données directement par le système global.

Étapes du processus d'intégration

[Parent et Spaccapietra, 2000] résume les principales étapes et difficultés du processus d'intégration menant à la réalisation d'un système de bases de données fédérées, ainsi que les différentes solutions envisageables pour chacune. Les principales étapes sont la pré-intégration, qui consiste à choisir le modèle de données commun et à définir les schémas d'exportation, la détermination des correspondances entre

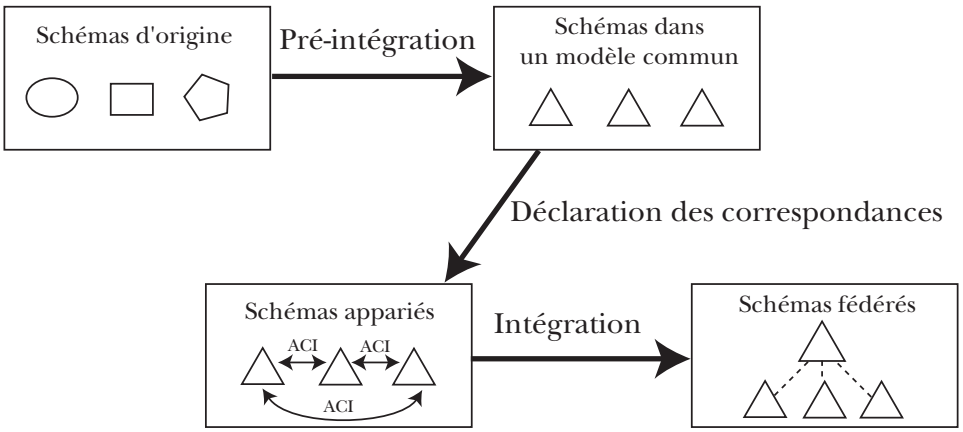


FIG. 2.3 *Les trois étapes du processus d'intégration de bases de données décrit par [Parent et Spaccapietra, 2000]*

schémas et enfin l'intégration proprement dite, c'est-à-dire la résolution des conflits, la définition du schéma fédéré et la création des liens entre ce schéma et les schémas locaux (figure 2.3). Les auteurs proposent un formalisme permettant la déclaration des correspondances entre bases de données : les ACI (assertions de correspondance interschémas). Ce formalisme décrit les correspondances au niveau des schémas ainsi que les règles permettant de déterminer quelles données se correspondent ; par exemple, si deux compagnies de location de voitures veulent intégrer leurs bases de données et que toutes deux ont une table des modèles de voitures disponibles, avec des informations différentes, l'ACI déclarera que les deux tables se correspondent, ce qui servira lors de la création du schéma intégré. Elle indiquera également comment on peut identifier les enregistrements qui se correspondent dans les deux tables, ce qui permettra l'accès aux données via le schéma intégré. Ces règles peuvent être relativement complexes : le nom du modèle peut s'appeler *nom* dans une des tables et *code* dans l'autre, être abrégé dans l'une des tables et pas dans l'autre, etc. L'ACI indiquera enfin quelles informations se trouvent dupliquées entre les deux bases, afin de gérer la cohérence entre les bases, de détecter les conflits, et de faciliter la mise à jour par l'intermédiaire du schéma intégré. Il se peut par exemple que chacune des deux tables contienne le nombre de sièges dans la voiture ; ces deux nombres doivent être identiques pour un même modèle. Nous reparlerons des ACI plus loin, dans le cas particulier des bases de données géographiques.

2.1.2 Cas particulier des bases de données géographiques

Utilité spécifique de l'intégration pour les données géographiques

L'hétérogénéité des données disponibles et le besoin d'utilisation simultanée de données en provenance de différentes sources sont particulièrement sensibles dans le cas des données géographiques pour plusieurs raisons. Non seulement il peut être nécessaire d'intégrer des informations relatives à des thèmes distincts, tels par exemple que l'hydrographie, la géologie et la végétation, qui peuvent provenir d'organismes différents, mais, même au sein d'un seul domaine thématique, on rencontre des pro-

blèmes liés au partitionnement pour peu qu'on s'intéresse à une zone comprenant des frontières. En effet, comme le rappelle [Laurini, 1996], les bases de données couvrant des zones adjacentes ont toutes les chances de ne pas se raccorder facilement, notamment en raison des différents choix de projection, des imprécisions contradictoires (figure 2.4) et, bien sûr, des choix de représentations différents de part et d'autre. L'auteur propose d'appliquer des « transformations élastiques » localisées sur des bandes de part et d'autre des frontières afin d'assurer le raccordement géométrique sans altérer les données d'origine.

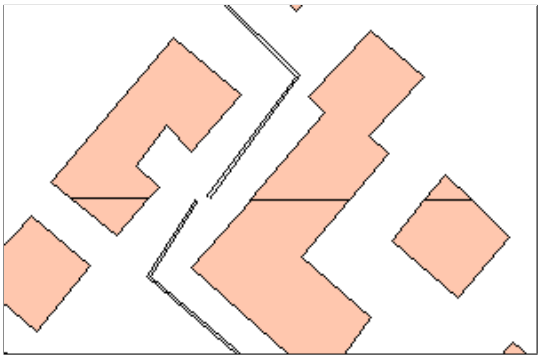


FIG. 2.4 *Limite entre deux feuilles BDTopo couvrant des zones adjacentes. On voit que même pour deux bases appartenant à la même série, qui sont donc globalement en cohérence, on peut trouver de petites imprécisions qui suffisent à poser des problèmes de raccordement.*

Mais le problème le plus spécifique aux bases de données géographiques n'est pas tant celui de l'utilisation simultanée d'informations sur des thèmes ou sur des zones différents que celui de l'utilisation simultanée d'informations sur un même thème et une même zone mais correspondant à des représentations différentes de la réalité, et en particulier à des niveaux de détail différents (figure 2.5). [Car et Frank, 1994] montre l'avantage de l'utilisation de plusieurs niveaux de détail pour l'exemple de la recherche d'itinéraire : il est nécessaire de disposer d'une représentation détaillée du réseau de transports considéré au voisinage des extrémités du parcours, qui doivent être rejointes avec exactitude. Mais les algorithmes de recherche de plus court chemin exacts, qui prennent en compte la représentation la plus détaillée sur l'ensemble du parcours, sont trop coûteux en temps de calcul dès que la taille et la densité du réseau sont importantes; il faut donc simplifier le problème en le structurant de façon hiérarchique, utilisant la représentation détaillée pour relier les extrémités du parcours au réseau principal et une représentation plus schématique pour naviguer sur ledit réseau principal. C'est d'ailleurs la démarche intuitivement adoptée par les gens en l'absence de système informatique. Un autre intérêt technique des données peu détaillées est qu'en raison de leur volume et de leurs coût et délai d'acquisition plus faibles, elles sont disponibles sur de plus grandes zones contiguës, et qu'on minimise les problèmes de raccordement en les utilisant là où elles suffisent; ceci peut même être une obligation si l'itinéraire doit traverser un endroit dont il n'existe pas du tout de représentation détaillée. Le point crucial de l'algorithme hiérarchique de recherche d'itinéraire est alors le passage d'un niveau de détail à l'autre, qui nécessite de faire le lien entre les différentes représentations de la réalité.

FIG. 2.5 *Différents niveaux de détail. À gauche, une représentation détaillée du réseau routier issue de Géoroute ; à droite, on a juxtaposé la même représentation sur Paris et une représentation moins détaillée, celle de la BDCarto, sur la banlieue.*

L'utilisation de plusieurs représentations différentes reliées entre elles pourrait également faciliter des analyses géographiques étudiant un phénomène et ses répercussions à différentes échelles.

Beaucoup d'utilisateurs des données bénéficieraient ainsi de l'existence de liens entre les diverses représentations disponibles, mais tel serait également le cas des organismes qui produisent et gèrent ces données. [Kilpeläinen, 2000] montre l'intérêt de tels liens pour la maintenance des bases de données, dans le cas où les représentations sont dérivées les unes des autres : ils permettent la propagation automatique sur les données dérivées des mises à jour effectuées sur les données de référence. [Badard et Lemarié, 1999, repris dans Badard, 2000] présente également un processus de propagation de mises à jour dont une partie importante consiste à trouver ou à retrouver ces liens s'ils n'ont pas été explicitement conservés ; ce processus a cependant été mis au point, lui aussi, dans un contexte où les différentes représentations dérivaient originellement les unes des autres. Mais, même dans le cas où l'on dispose de plusieurs bases de données constituées tout à fait indépendamment, il existe toujours des redondances entre les informations stockées dans chacune d'elles, pour peu qu'elles couvrent une zone commune. Les informations nécessaires à la mise à jour des différentes bases se recoupent donc en partie et, si les bases sont mises à jour indépendamment, on saisit plusieurs fois la même information, ce qui non seulement prend du temps mais multiplie les risques d'erreur. Les travaux mentionnés plus haut pourraient être adaptés à la mise à jour simultanée de plusieurs représentations

quelconques d'un même lieu, à condition qu'on sache déterminer les liens entre ces représentations.

La mise en relation des différentes bases de données permettrait également de détecter et de traiter les incohérences entre elles [Egenhofer *et al.*, 1994 ; Sheeren *et al.*, 2004]. Dans un contexte de contrôle qualité, elle permettrait l'utilisation d'une des bases, supposée de meilleure qualité, comme référence pour le contrôle de l'autre [Bel Hadj Ali, 2001].

Bases de données géographiques multi-représentations

Depuis plusieurs années, de nombreuses recherches s'intéressent, pour les raisons citées ci-dessus, aux bases de données géographiques *multi-représentations* ou à représentations multiples [Devogele *et al.*, 2002]. Les bases de données dites multi-échelles en constituent un cas particulier. Ce dernier terme constitue un abus de langage puisque, les données n'étant pas représentées graphiquement dans les bases, elles n'ont pas d'échelle à proprement parler. Le terme d'échelle n'est donc utilisé que pour signifier ce que nous appelons *niveau de détail* dans ce mémoire : une base de données « au 1:25 000 » désigne une base contenant des données à un niveau de détail équivalent à celui d'une carte au 1:25 000 [Ruas et Bianchin, 2002].

[Rigaux, 1994], après avoir rappelé les principales raisons conduisant à la coexistence de plusieurs représentations d'un même terrain, explique pourquoi il n'est pas envisageable, afin d'éviter les problèmes liés à l'indépendance des représentations, de stocker uniquement les données les plus détaillées possibles et d'en dériver dynamiquement, à la demande, les données moins détaillées, qui seraient donc automatiquement à jour relativement aux données de référence. L'une des raisons est que le processus de *généralisation*, qui consiste à créer une représentation synthétique à partir d'une représentation détaillée, est beaucoup plus complexe qu'on pourrait le penser [Ruas, 2002] : s'il existe d'ores et déjà un certain nombre de prototypes pour son automatisation, des retouches manuelles restent toujours nécessaires et, de toute façon, les temps de calcul sont trop longs pour faire de la dérivation à la demande. L'autre raison est qu'un tel modèle supposerait qu'on commence par acquérir des données extrêmement détaillées pour réaliser seulement ensuite les représentations moins détaillées. Or en raison des différences de quantités de données, et donc de coûts et de délais, entre les deux cas, on dispose toujours de la représentation la moins détaillée en premier. À titre d'exemple, la base de données de niveau de détail équivalent au 1:50 000 de l'IGN, la BDCarto, couvre la France métropolitaine depuis une quinzaine d'années tandis que la BDTopo, équivalente au 1:10 000, est encore inachevée. En termes de volume de données, celui total de la BDCarto est évalué à 2,5 Go ; pour la BDTopo dans sa version standard, celui moyen d'une feuille (550 km^2 en moyenne) est d'environ 500 Mo. Si elle était achevée, elle comprendrait 1096 feuilles, soit un volume total d'environ 500 Go, 200 fois plus que la BDCarto[1]. On pourrait ajouter le fait que la différence de point de vue se réduit rarement à une simple différence de niveau de détail, ainsi les données moins détaillées ne sont pas toutes dérivables des données détaillées. Par exemple, figure 2.6-a, la description la moins

1. Ces estimations dont bien sûr très imprécises, le volume de données par km^2 n'étant pas constant et dépendant par ailleurs du format de stockage des données.

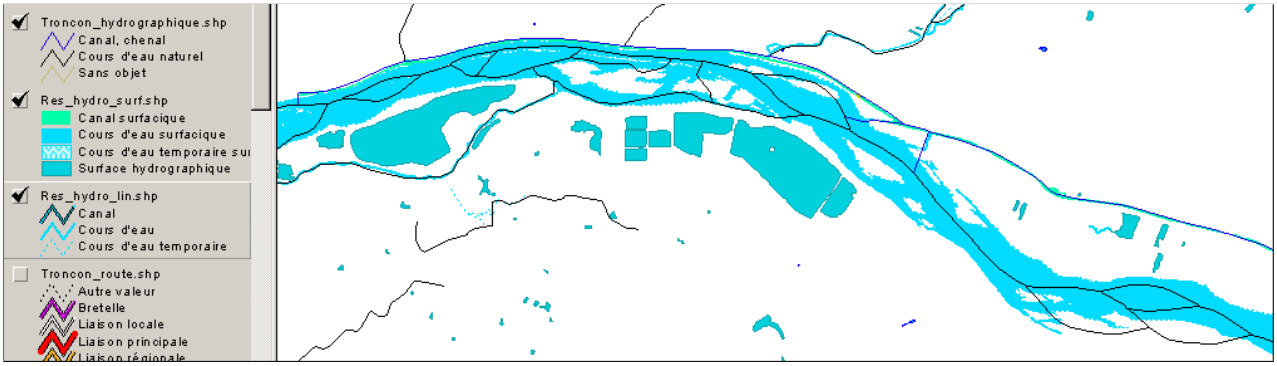
détaillée (en noir) vise à décrire l'écoulement de l'eau, tandis que la description détaillée (en bleu) décrit les surfaces occupées par l'eau. La représentation noire ne se déduit pas de façon évidente de la représentation bleue. On peut mentionner, dans le même ordre d'idées, que la BDCarto (peu détaillée) contient des informations sur la navigabilité des cours d'eau qui sont absentes de la BDTopo (détaillée).

Il est donc souhaitable de disposer d'une architecture plus complexe permettant la gestion simultanée de plusieurs représentations. [Timpf, 1998] propose un modèle à plusieurs niveaux de détail où les objets sont reliés par des structures hiérarchiques, un objet d'un niveau donné pouvant être associé à un ou plusieurs objets du niveau plus détaillé. Ces hiérarchies peuvent être de différents types, selon la nature des objets qu'elles concernent [Timpf, 1999] : on distingue des hiérarchies d'*agrégation/décomposition*, correspondant à un regroupement spatial de plusieurs objets de même type, des hiérarchies de *généralisation/spécialisation*[1], correspondant au regroupement sémantique en un seul type plus général d'objets de différents types, et des hiérarchies de *filtrage* correspondant à la suppression des éléments trop peu significatifs. L'auteur propose un exemple (figure 2.7) où les réseaux hydrographique et de transport sont structurés par une hiérarchie de filtrage (à gauche). À cette hiérarchie du réseau correspond une hiérarchie d'agrégation-décomposition sur les îlots (« containers »), un îlot étant défini comme une portion connexe de l'espace découpé par le réseau sélectionné. La troisième hiérarchie correspond aux zones d'occupation du sol, qui subdivisent les îlots ; il s'agit d'une hiérarchie de généralisation/spécialisation. La dernière hiérarchie, qui correspond à des éléments tels que les bâtiments, ressemble à un filtrage, mais elle est un peu plus complexe. En effet, il existe une méthode de généralisation cartographique consistant à représenter un groupe d'objets par un autre groupe d'objets, plus simple, représentatif de la structure du groupe détaillé. Cette méthode ne permet pas d'établir des liens individuels entre les objets des différents niveaux de détail, c'est pourquoi la hiérarchie des éléments fait apparaître des groupes.

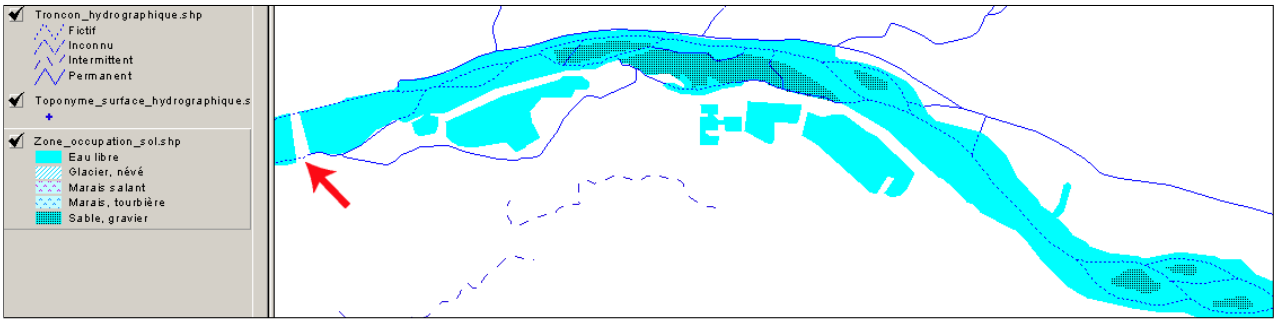
[Vangenot, 2001] s'intéresse à trois types de représentation multiple d'une entité : la résolution spatiale multiple, le point de vue multiple et la classification multiple. La résolution spatiale correspond au niveau de détail géométrique. Les bases de données multi-échelles mentionnées plus haut sont des bases gérant le premier de ces types de multi-représentation, c'est-à-dire permettant à une unique entité de posséder des représentations différentes à des niveaux de détail différents. La gestion du deuxième, consistant à fournir des représentations différentes suivant le point de vue de l'utilisateur, est un problème bien connu pour les bases de données classiques et le modèle relationnel permet, pour le résoudre, la définition de vues [Gardarin, 1999, chap. 9]. Diverses propositions ont été faites pour doter le modèle objet, plus complexe, d'un mécanisme similaire. [Claramunt, 2002] en propose une adaptation spécifique pour les bases de données spatiales.

Quant à la classification multiple des objets, le modèle objet classique ne la permet que de façon très limitée : un objet appartient à une classe de base et à toutes les classes plus générales dont cette classe est une spécialisation, à l'exclusion de toute autre. La classification multiple fait donc appel au concept de *rôle*, qui est défini comme une

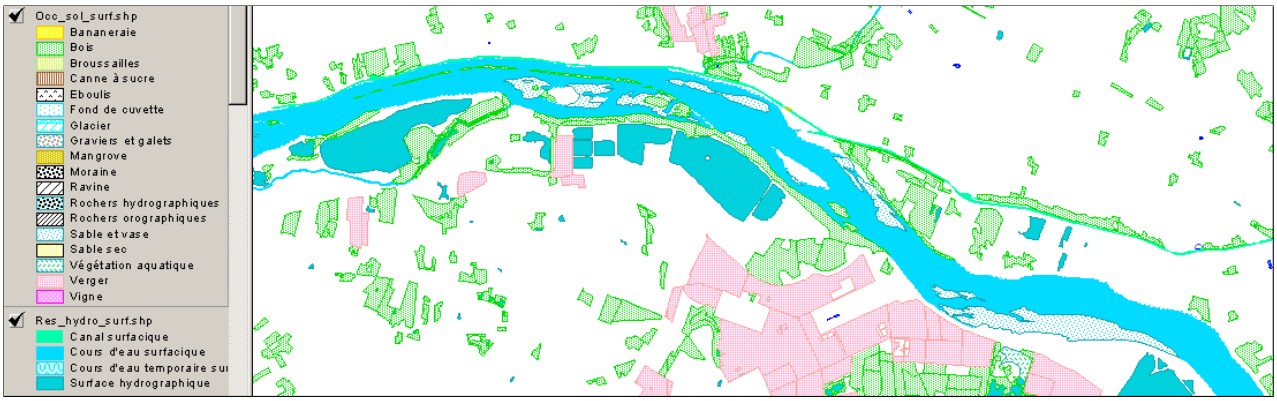
1. Ceci n'a aucun rapport avec la généralisation cartographique mentionnée plus haut.



a. Superposition de deux représentations complètement différentes de l'hydrographie : celles de la BDCarto (tronçon_hydrographique dans la légende) et de la BDTopo (res_hydro_surf et res_hydro_lin). On voit qu'au-delà de la différence de niveau de détail, la Loire n'est pas conceptualisée de la même façon dans les deux cas.



b. Superposition du thème hydrographie et de la partie hydrographique du thème occupation du sol de la BDCarto. On voit ici que l'emprise du fleuve apparaît bien dans la base, mais sous forme de zones d'occupation du sol, qui ne portent aucun attribut sémantique et ne sont pas explicitement reliées aux objets décrivant le réseau hydrographique. Les tronçons hydrographiques sont ici classés suivant l'attribut « état » et non l'attribut « nature » comme précédemment, ce qui permet de remarquer que l'état « fictif » indique la superposition, dans la base, du tronçon et d'une surface d'eau ; c'est là le seul indice sémantique d'une relation entre les deux thèmes. Le rôle très graphique de la zone d'occupation du sol est par ailleurs mis en évidence par l'interruption du tracé au niveau du pont (flèche rouge). L'intervalle est occupé par une zone d'occupation du sol de type « bâti », non représentée ici.



c. Hydrographie et occupation du sol de la BDTopo. On voit que si l'on souhaite comparer le tracé du fleuve entre les deux bases, il est nécessaire de prendre en compte les zones de sable pour aboutir à une comparaison raisonnablement cohérente, ce qui n'était pas absolument évident a priori.

FIG. 2.6 *Au-delà de la différence de niveau de détail, la différence de point de vue. La BDCarto possède un thème hydrographique axé sur la structure de réseau et les écoulements principaux, et un thème occupation du sol à but essentiellement cartographique (il s'agit en fait d'indiquer les grands types de paysage : zone d'eau, zone de forêt, zone urbaine, etc.) Dans la BDTopo, plus détaillée, le thème hydrographique se concentre sur la représentation précise des surfaces en eau sans chercher à décrire les écoulements ; les classes relatives à l'occupation du sol, quant à elles, visent plutôt à représenter individuellement les éléments du paysage que le type de paysage.*

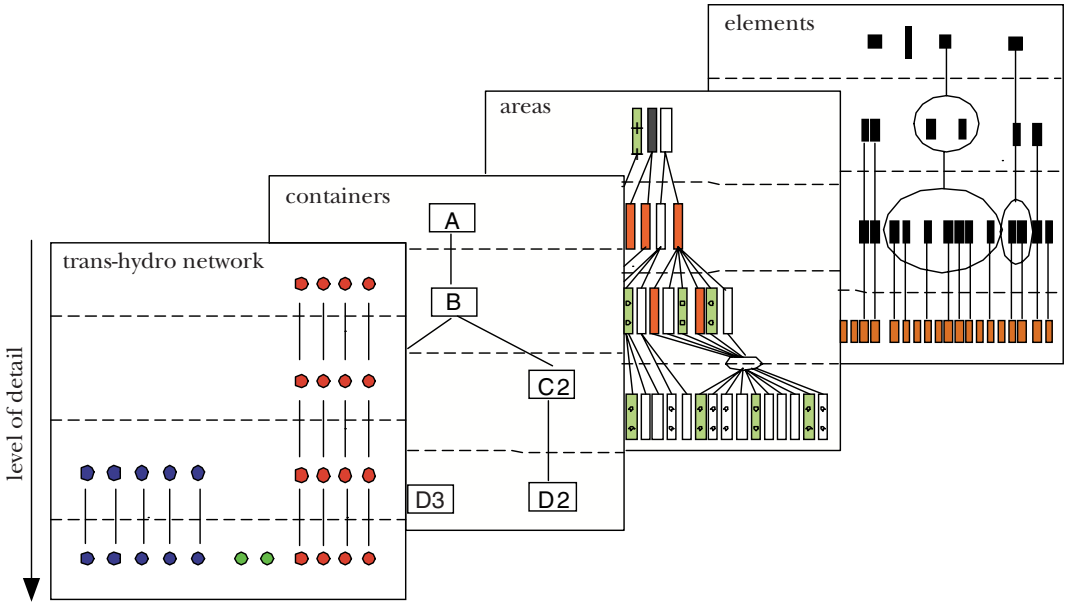


FIG. 2.7 *Figure extraite de [Timpf, 1998], montrant les différentes hiérarchies utilisées par le modèle Map Cube.*

classe à ceci près qu'un objet peut avoir (jouer) plusieurs rôles simultanément[1]. Dans la plupart des modèles incluant des rôles, les objets sont membres d'une unique classe de base, correspondant en quelque sorte à leur nature, invariable, qui leur permet d'être compatibles avec le modèle objet usuel. Ils peuvent d'autre part avoir un ou plusieurs rôles et en changer, ce qui correspondrait à leurs fonctions, d'où le nom de rôle. Les rôles ont été traités par de nombreux travaux, notamment [Coulondre, 2000].

Toujours pour la classification multiple, le modèle de données MADS (Modeling of Application Data with Spatio-temporal features [Parent *et al.*, 1997]), qui s'appuie sur le modèle entité-association en lui ajoutant des éléments spécifiques pour la modélisation de données spatio-temporelles, propose des liens « may-be », qui indiquent simplement que les instances d'un type *peuvent* également être des instances d'un autre type.

[Vangenot, 2001] propose ensuite d'étendre le modèle MADS pour la multi-représentation, à l'aide de deux mécanismes qui peuvent être utilisés tous deux au sein d'un même schéma, les estampilles et les liens inter-représentations. Le système d'estampilles fonctionne de la façon suivante : chaque représentation est symbolisée par une estampille. Chaque élément du schéma (type d'association, type d'objet, attribut) peut porter une ou plusieurs estampilles indiquant dans quelles représentations il est visible. Par ailleurs un attribut commun à plusieurs représentations, c'est-à-dire qui porte plusieurs estampilles, peut avoir une valeur par représentation. Typiquement, un objet géographique multi-représenté possédera un attribut « géométrie »

1. Le concept de rôle ici mentionné correspond à une véritable classification alternative et ne doit pas être confondu avec l'utilisation, fréquente, du mot « rôle » pour désigner simplement les extrémités d'une association, dans les modèles objet ou entité-association.

dans toutes les représentations, mais la valeur de cet attribut sera une géométrie plus ou moins détaillée suivant la représentation qu'on considère. Ceci correspond à un estampillage des *valeurs* de l'attribut. Il est également possible d'estampiller les instances, c'est-à-dire les objets eux-mêmes. Ainsi un type d'objet peut être commun à plusieurs représentations mais certains des objets de ce type n'exister que dans certaines de ces représentations. Une estampille est décrite par un couple (point de vue, résolution) ; les liens is-a et may-be peuvent relier des types portant des estampilles différentes, ce qui permet l'orthogonalité entre les trois types de multi-représentation mentionnés plus haut.

Le second mécanisme, complémentaire de celui des estampilles, est celui des associations inter-représentations. Il est utilisé quand un même phénomène du monde réel doit être représenté par des instances différentes dans les différentes représentations. Cela peut être par exemple parce que le phénomène n'est pas représenté par le même nombre d'objets dans les différentes représentations, ou parce que la base de données multi-représentations provient de l'intégration de bases de données existantes dont les instances n'ont pas été fusionnées. Ces associations peuvent être de différents types : identité si à une instance dans une représentation correspond toujours exactement une instance dans l'autre, agrégation si une instance dans une représentation peut correspondre à un groupe d'instances dans l'autre, SetToSet (ensemble à ensemble) dans le cas général.

Cette proposition a été mise en œuvre par le projet européen MurMur (Multipe representation, Multiple resolution) et a été testée pour la représentation, sous forme d'un unique schéma multi-représentations, d'extraits de schémas correspondant à la représentation du réseau routier dans trois bases de données différentes de l'IGN [Balley *et al.*, 2004]. Sur la figure 2.8, le type d'objet Route (en bas) est présent dans toutes les représentations : il porte les trois estampilles. Les estampilles situées sur sa droite correspondent aux attributs, qui sont également présents dans toutes les représentations. Ce type d'objet Route est relié à trois types d'association différents, portant des estampilles qui indiquent que dans la représentation bleue, une route est composée d'objets de type Tronçon_T, que dans la représentation rouge, elle est composée d'objets de type Tronçon_C, et de même pour la représentation jaune avec le type Tronçon_G. On peut voir un exemple de liens inter-représentations dans la partie supérieure du schéma : le type d'objet Noeud_C, dans la représentation rouge, peut correspondre à un groupe d'objets de type Tronçon_T dans la représentation bleue (à gauche). Il s'agit d'une association d'agrégation. On a le même type de lien vers la droite, avec la représentation jaune. On peut également remarquer que dans la partie centrale du schéma, certains types d'associations, représentés avec des liens jaunes, héritent d'autres types plus généraux. Cela signifie que pour certaines instances, l'association SetToSet entre les représentations est en fait une relation d'identité.

[Friis-Christensen, 2003] définit un MRSL (Multiple representation schema language) dont l'idée principale consiste à définir des classes d'intégration (i-classes), par opposition aux classes de représentation (r-classes). Une i-classe représente un type d'entité et est reliée à différentes r-classes correspondant aux différentes représentations possibles pour ce type d'entité (figure 2.9) ; les r-classes peuvent être considérées comme des rôles associés à l'i-classe. L'auteur présente un système de

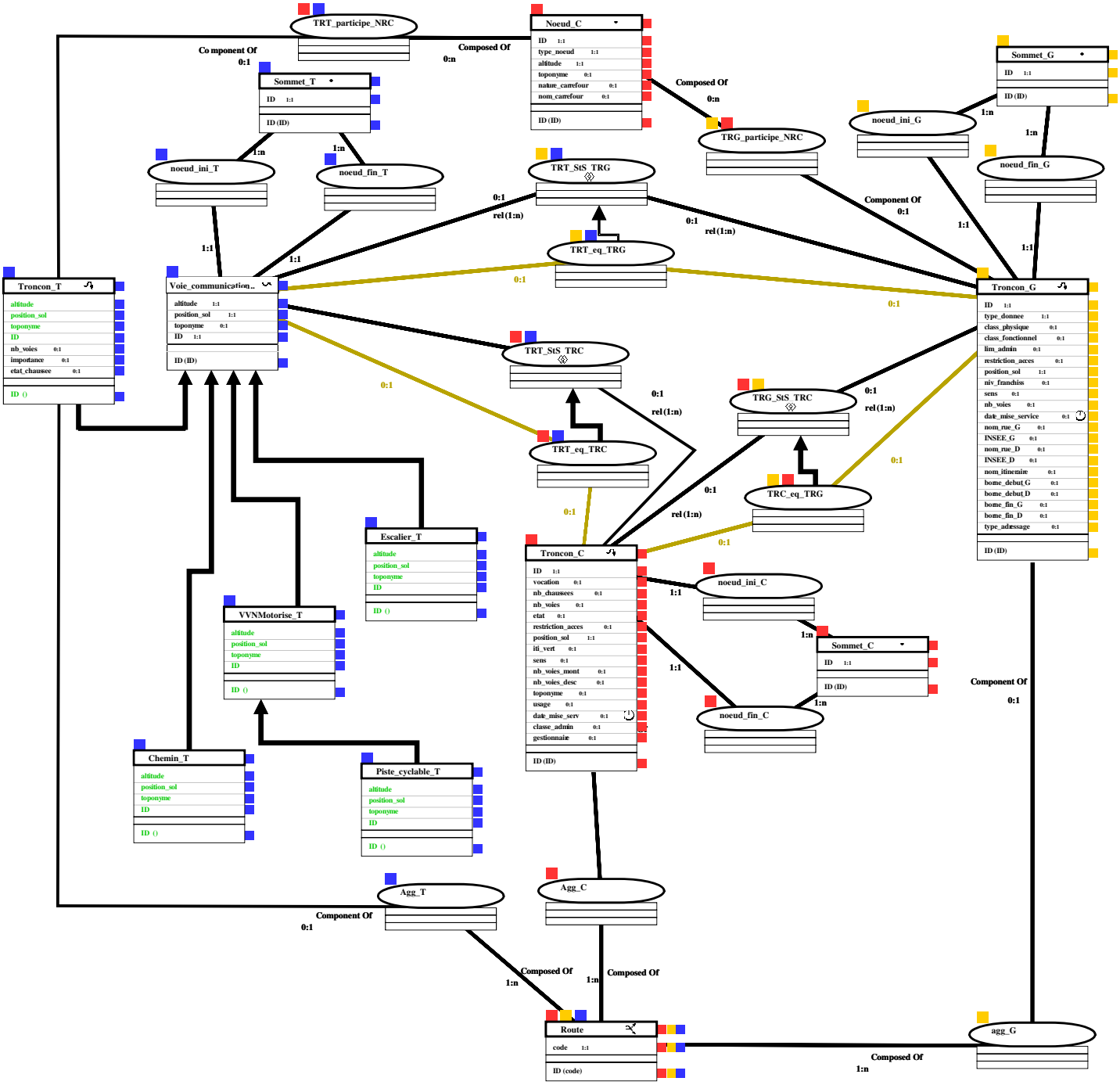


FIG. 2.8 *Exemple de schéma réalisé avec l'éditeur MADS du projet MurMur. L'estampille bleue correspond à la BDTopo, la rouge à la BDCarto et la jaune à Géoroute.*

gestion des représentations multiples (multiple representation management system, MRMS) s'appuyant sur ce langage et dont l'objectif principal est le maintien de la cohérence entre les différentes représentations. Celles-ci sont indépendantes les unes des autres et situées dans des bases de données distinctes ; les i-objets, instances des i-classes, sont stockés au sein du MRMS. Ils sont créés par l'appariement des données et contiennent des règles de correspondance permettant, lors d'une mise à jour, de vérifier que celle-ci n'a pas altéré la cohérence entre représentations et de restaurer cette cohérence dans le cas contraire. Il est à noter que le but de ce système n'est pas la consultation unifiée des différentes bases de données : le schéma multi-représentations est volontairement réduit à la représentation des informations communes à plusieurs des schémas d'origine, c'est-à-dire que les entités mono-représentées n'apparaissent jamais sous forme d'i-objets et que les i-classes ne référencent qu'un minimum d'attributs.

Sur la figure 2.9, TM (technical map), BR (buildings register), et T10 (topographic map) correspondent à trois bases de données différentes. TM est d'un niveau de détail équivalent au 1:5 000, T10 correspond au 1:10 000, BR contient des informations thématiques et ne représente les bâtiments spatialement que par des points. Il existe une correspondance 1:1 entre les bâtiments TM et BR ; l'i-classe « building » sert à gérer cette correspondance. En revanche, dans T10 les bâtiments peuvent être agrégés, ce qui correspond à une relation 1:n ; l'i-classe « detached_building » permet de gérer cette agrégation. La mention « master » sur le lien entre building et TM_building indique que si l'objet de la base TM est détruit, l'objet correspondant dans la base BR doit l'être aussi. Les attributs notés m1, m2 et m3 sont des attributs *maîtres*, c'est-à-dire que c'est à partir d'eux que les attributs des i-classes sont calculés. Chaque i-classe inclut des « Value Correspondances » (VC) qui permettent de vérifier la cohérence. Par exemple t10.shape = shape dans la classe detached_building indique que l'attribut shape du bâtiment t10 doit être égal à l'attribut shape du detached_building, qui est défini comme l'agrégation des attributs shape des building correspondants, c'est-à-dire des tm_building correspondants. On remarque que les i-classes ne possèdent que les attributs qui sont utiles pour le maintien de la cohérence.

Processus d'intégration dans le cas particulier des données géographiques

Pour les diverses raisons mentionnées plus haut, relatives notamment au coût d'acquisition des données, la réalisation d'une base de données multi-représentations devrait s'appuyer sur les bases de données déjà existantes : on est confronté au problème de l'intégration de bases de données géographiques [Devogele *et al.*, 1998]. L'architecture de bases de données fédérées permettrait une gestion centralisée adaptée notamment au maintien de la cohérence entre représentations et à la propagation des mises à jour, tout en conservant l'autonomie des bases préexistantes; elle semble donc un choix judicieux.

Si l'on suit le processus d'intégration décrit par [Parent et Spaccapietra, 2000], déjà mentionné plus haut, les étapes préalables à la définition du schéma intégré sont la pré-intégration et la déclaration des correspondances entre les schémas à l'aide d'ACI (assertions de correspondance interschémas). Une ACI, dans le cas d'un modèle orienté-objet, décrit la relation entre deux classes qui se correspondent et inclut un certain nombre de clauses. Certaines de ces clauses donnent des informations sur les relations entre les contenus des instances qui se correspondent, notamment sur les

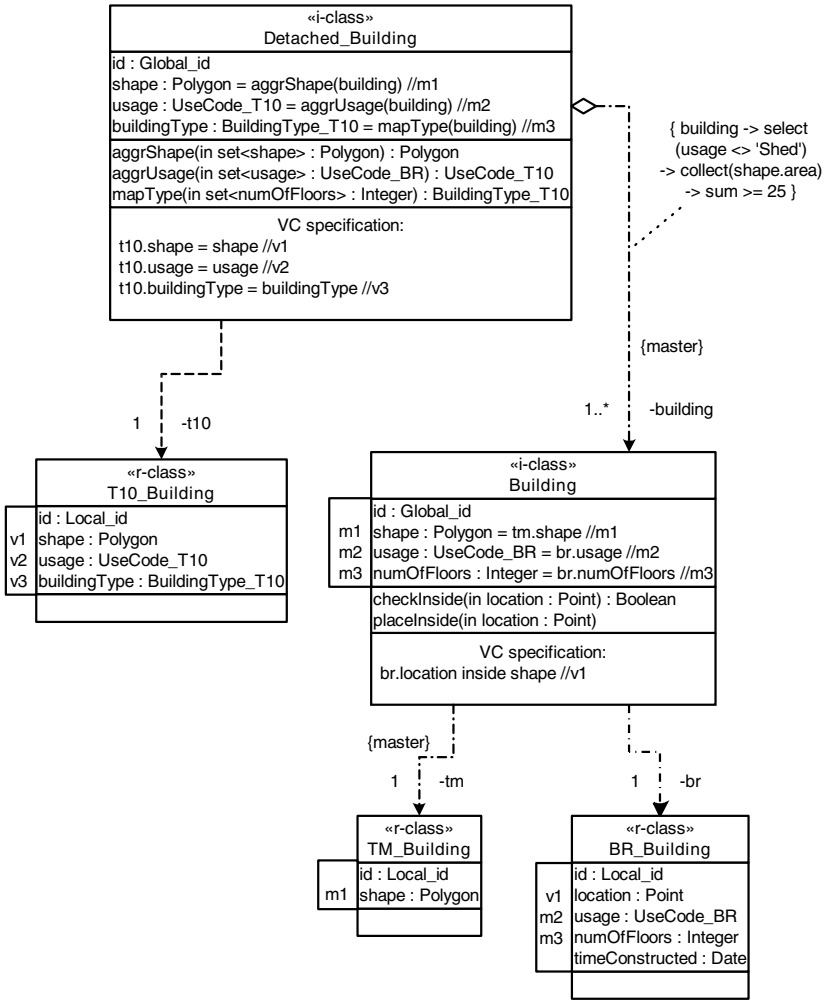


FIG. 2.9 *Exemple de schéma multi-représentations tiré de [Friis-Christensen, 2003].*

valeurs d'attributs. En principe, une autre clause indique comment sont déterminées les correspondances entre instances, c'est-à-dire comment on repère les objets de chacune des deux classes qui représentent la même entité réelle. Si cette clause peut souvent être relativement simple dans le cas des bases de données classiques, où la présence d'un attribut identifiant sans ambiguïté, tel qu'un numéro INSEE, est fréquente, ce n'est pas le cas dans les bases de données géographiques, où le seul critère permettant cette détermination est le plus souvent la géométrie de l'objet (forme et position). [Devogele, 1997] mentionne la possibilité de recourir à une clause spécifique, « appariement géométrique des données », décrivant une règle de correspondance pour chacune des classes, mais indique que la complexité du problème ne permet généralement pas de raisonner classe par classe. Il propose en conséquence le remplacement de toutes ces clauses formelles par un processus global d'appariement, réalisé par logiciel, pour lequel il fournit un certain nombre d'algorithmes. L'appariement géométrique est une étape importante de l'intégration

de bases de données géographiques et a été traité par de nombreux autres travaux, par exemple [Uitermark *et al.*, 1998 ; Zhang, 1998 ; Badard et Lemarié, 2002].

Si le problème de l'appariement de données est particulièrement complexe dans le domaine de l'information géographique, la détermination des correspondances entre les schémas n'est pas à négliger pour autant. [Lassoued *et al.*, 2004] étudie l'ajout d'une nouvelle source de données à un système par médiation de type *local as views* (voir 2.1.1). Il s'agit donc d'apparier un nouveau schéma à un schéma global préexistant. L'approche proposée consiste à analyser un jeu de données pour en déduire des correspondances au niveau des schémas, en s'appuyant sur l'hypothèse que les données représentant un même type d'entités présentent des similitudes : par exemple, une église est généralement en forme de croix et une route suit un parcours plus régulier qu'une rivière. Un système à base d'apprentissage permet, après avoir été entraîné sur des schémas déjà appariés manuellement au schéma global, de déterminer automatiquement, grâce à ces critères morphologiques, les correspondances probables entre ce schéma global et un nouveau schéma. Cette approche présente la particularité de ne faire appel à aucune information extérieure sur la sémantique des données, c'est-à-dire qu'elle n'utilise que les données elles-mêmes et aucune métadonnée. Cet aspect est utile si l'on ne dispose pas de métadonnées ou qu'elles ne sont pas facilement exploitables. Cependant, si les niveaux de détail des différentes bases de données diffèrent trop, il sera plus difficile de trouver des similitudes ; et d'autre part il ne s'agit ici que d'apparier des schémas à un schéma global prédéfini et non de relier deux schémas quelconques. Enfin, la mise en œuvre concrète de ce processus semble délicate et nécessite un travail préalable poussé, notamment d'analyse spatiale pour bien choisir les échantillons et assurer le succès de l'apprentissage.

Plus proche de notre problématique, [Sotnykova, 2003] s'intéresse à la fédération de bases de données spatio-temporelles et présente des outils interactifs permettant la gestion de plusieurs schémas exprimés dans le modèle MADS, la déclaration des ACI et la génération du schéma fédéré. [Park, 2001] propose un outil aux fins similaires mais utilisant un formalisme différent, USM*. Dans les deux cas, les correspondances inter-schémas doivent être déterminées par l'utilisateur. Celui-ci aura besoin pour ce faire d'informations détaillées sur la sémantique des données, informations dont seule une petite partie est contenue dans les schémas, comme le rappelle [Parent et Spaccapietra, 2000]. Il devra donc s'appuyer sur les connaissances des experts du domaine, qui sont intervenues de façon implicite dans la définition des schémas, et, surtout, sur la documentation des bases de données et en particulier leurs spécifications.

2.2 Nécessité d'utiliser la sémantique pour l'intégration

Selon [Sheth, 1998], on peut distinguer quatre niveaux d'interopérabilité entre les systèmes, suivant les types d'hétérogénéité plus ou moins complexes qu'ils gèrent. Le premier type d'hétérogénéité a été l'hétérogénéité *système* : dans les années 1970 se posaient des problèmes de communication entre matériels et systèmes d'exploitation différents ; ils ont été résolus par l'avènement des protocoles internet. Par la suite se

. L'étoile n'indique pas une note de bas de page ; le formalisme USM est une extension spatio-temporelle d'un formalisme appelé USM (pour Unifying Semantic Model) sans étoile.

sont posés les problèmes de l'hétérogénéité *structurelle*, c'est-à-dire les différences de modèles de données, et de l'hétérogénéité *syntaxique*, relative aux langages de manipulation de données. Des solutions ont été apportées par les formats standardisés d'échange de données. Le dernier type d'interopérabilité à obtenir est l'interopérabilité *sémantique*, décrite par l'auteur comme la possibilité de relier le contenu et la représentation des sources d'information à des entités et des concepts du monde réel. Idéalement, des systèmes sémantiquement interopérables pourraient répondre de façon unifiée à des requêtes de haut niveau portant sur les concepts et ne nécessitant pas la connaissance de la structure ou du schéma de la base de données.

2.2.1 Les ontologies

Définitions du terme « ontologie »

Ces concepts du monde réel seraient représentés par des *ontologies*. Le terme d'« ontologie » apparaît de plus en plus fréquemment dans la recherche sur les systèmes d'informations et sur les bases de connaissances mais les significations qu'on lui prête sont relativement diverses [Guarino et Giaretta, 1995]. Il est à l'origine issu de la philosophie, où il désigne l'étude de l'être (l'élément onto- étant formé sur le participe présent du verbe être grec). Mais il a été repris dans des domaines techniques avec des sens plus ou moins concrets. [Gruber, 1993] a ainsi défini une ontologie, dans le domaine de l'intelligence artificielle, comme « la spécification explicite d'une conceptualisation »[1]. [Partridge, 2002], citant *the Oxford companion to philosophy*, définit une ontologie comme « l'ensemble des choses dont l'existence est admise par une théorie ou un système de pensée donné »[2]. Cette définition est relativement proche de celle généralement admise dans le domaine des systèmes d'information si l'on considère que l'ensemble en question doit être défini en *intension* et non en *extension* : il s'agit le plus souvent de décrire un ensemble de *concepts* partagés au sein d'un domaine plus ou moins restreint plutôt que l'ensemble des *instances* de ces concepts, qui correspondrait à la définition fournie par Partridge.

[Guarino, 1998] propose de distinguer différents types d'ontologies selon leur niveau de généralité : ontologies de haut niveau (« toplevel »), de domaines, de tâches et d'applications. Les ontologies de plus haut niveau décrivent des concepts très généraux comme ceux d'espace, de temps, de matière, d'objet, d'événement, d'action, etc. Les ontologies de domaines ou, respectivement, de tâches décrivent l'ensemble des concepts partagés relatifs à un domaine d'étude, tel que les automobiles ou l'information géographique, ou, respectivement, à une tâche générique, telle que la vente ou la construction. Enfin, les ontologies d'application décrivent des concepts spécifiques simultanément à un domaine et à une tâche, comme par exemple « terrain constructible ».

Lorsqu'il est utilisé dans un sens concret, le terme d'ontologie peut désigner, entre autres, une liste de termes reliés entre eux par des liens sémantiques tels que synonymie, hyponymie ou hyperonymie, ce qui en fait pratiquement un synonyme de « thésaurus ». Mais il peut également désigner une structure complexe s'apparentant à un schéma conceptuel de base de données. [Cullot *et al.*, 2003] appelle « ontologie

1. « An *ontology* is an explicit specification of a conceptualization. »
2. « the set of things whose existence is acknowledged by a particular theory or system of thought »

descriptive » une telle structure, présente quelques systèmes permettant leur modélisation et leur gestion, et propose des pistes pour aller plus loin dans le cas particulier des ontologies géographiques. Les ontologies descriptives peuvent être représentées dans des langages à base de logiques de description [Baader *et al.*, 2003], dont il existe des extensions spatiales telles que celles proposées par [Haarslev *et al.*, 1998]. Elles peuvent également être représentées à l'aide d'un modèle originellement prévu pour les bases de données, tel que MADS. La ressemblance avec un schéma de base de données peut aller jusqu'à la possibilité de définir des instances des éléments de l'ontologie, ce qui correspondrait à peupler la base de données définie par le schéma. Cependant, il existe deux différences importantes entre cette structure et un schéma de bases de données :

— le schéma de l'ontologie peut évoluer sans arrêt, tandis que les évolutions du schéma sont limitées dans le cas d'une base de données ;

— le schéma de l'ontologie peut être très volumineux ; il faut donc disposer d'un langage de requêtes permettant d'interroger aussi bien le schéma que les instances, et présentant en outre des possibilités de raisonnement sur le schéma. Ce n'est généralement pas le cas des langages de requêtes pour bases de données.

Utilisation d'ontologies pour l'intégration de données

[Wiederhold, 1994] montre l'utilité des ontologies dans les architectures de médiation : elles permettent de s'affranchir des différences terminologiques et d'assurer la compatibilité entre les différentes modélisations du domaine : au sein du médiateur, au niveau de l'application et au niveau de chacune des sources de données. [Kashyap et Sheth, 1996] insiste également sur la nécessité de la description de la sémantique pour gérer la surabondance de plus en plus importante d'information disponible sur l'« infrastructure globale d'information » dont le web, par exemple, est une partie. Les outils essentiels pour se repérer dans cette infrastructure sont les *métadonnées*, c'est-à-dire au sens large toutes les données décrivant d'autres données, dont les ontologies sont un cas particulier important.

Pour [Partridge, 2002], tout schéma conceptuel de base de données est issu d'une certaine conceptualisation du monde, qu'il appelle « ontologie sous-jacente » du schéma. Un schéma dont les éléments correspondent exactement chacun à un concept de l'ontologie sous-jacente est qualifié de « schéma ontologique » ; mais de tels schémas se rencontrent rarement dans la pratique. On qualifie alors de « divergence sémantique » les relations plus complexes entre le schéma et son ontologie. L'auteur indique que les différences de divergence sémantique entre bases de données sont un des principaux facteurs de l'hétérogénéité sémantique, pour peu que les bases relèvent du même domaine d'étude et possèdent une ontologie sous-jacente commune. Il propose en conséquence, en vue de l'intégration, de commencer par expliciter la divergence sémantique des différentes bases et d'utiliser l'ontologie sous-jacente comme base pour la définition du schéma intégré (figure 2.10). Dans le domaine de l'information géographique, [Fonseca *et al.*, 2003] montre également l'intérêt de relier ontologies et schémas conceptuels pour l'intégration. Selon les auteurs, l'ontologie décrit les concepts d'un domaine tandis que le schéma conceptuel décrit la structure d'une base de données, et l'établissement de liens entre eux permet d'intégrer les données en s'appuyant sur leur signification et non sur leur structure.

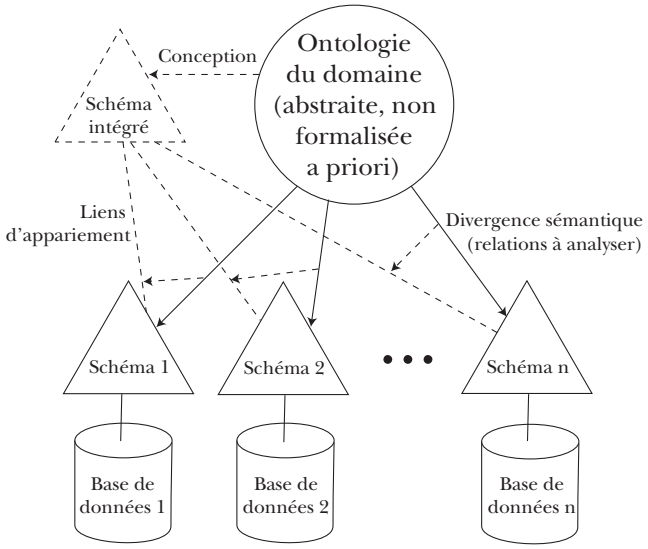


FIG. 2.10 *Méthode décrite par [Partridge, 2002]. On fait l'hypothèse que les bases de données à intégrer relèvent d'une même conceptualisation du monde, c'est-à-dire qu'il existe, de façon abstraite, une ontologie sous-jacente commune à toutes ces bases. On analyse alors cette ontologie et les déviations des schémas par rapport à elle (divergence sémantique). Le schéma intégré est ensuite conçu à partir de l'ontologie et l'analyse des divergences sémantiques permet de définir les liens d'appariement.*

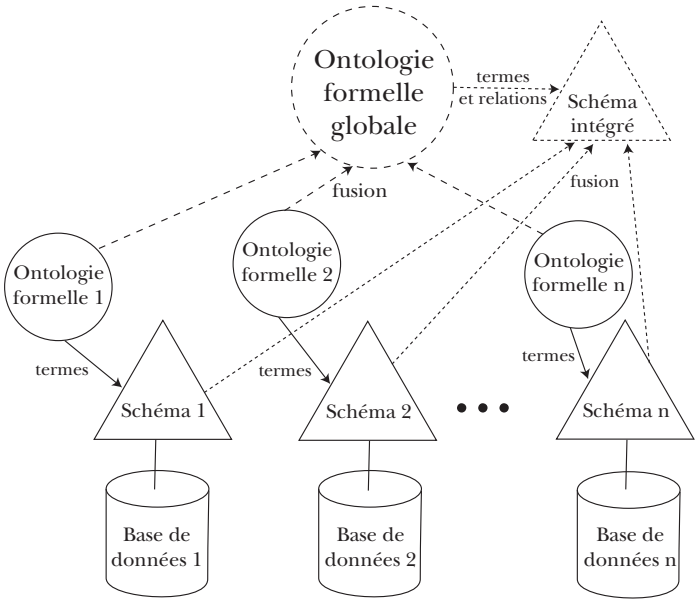


FIG. 2.11 *Méthode décrite par [Hakimpour et Geppert, 2002]. On suppose que chaque schéma local a été décrit avec des termes définis dans une ontologie formelle. Ces ontologies sont alors fusionnées pour constituer une ontologie formelle globale dans laquelle les relations sémantiques entre les différents termes sont déterminées. Le schéma intégré est ensuite le résultat de la fusion des schémas locaux à l'aide de ces relations.*

[Hakimpour et Geppert, 2002] propose plus concrètement une méthode de génération du schéma intégré faisant appel à des ontologies. Les ontologies sont ici des ensembles de termes reliés entre eux et munis de définitions formelles, et il n'est pas considéré a priori qu'il existe une ontologie sous-jacente commune à toutes les bases à intégrer. Les schémas locaux sont censés avoir été définis à l'aide de termes représentés dans une telle ontologie formelle spécifique à chacune des bases, et la première partie du processus consiste à fusionner les ontologies, c'est-à-dire à rassembler tous les termes et à calculer automatiquement les relations de similitude entre leurs définitions. L'article explique ensuite comment créer le schéma intégré à partir de l'ontologie globale ainsi obtenue. Dans le domaine plus particulier de l'information géographique, [Hakimpour et Timpf, 2001] se concentre également sur la fusion d'ontologies formelles en vue de l'intégration de bases de données appartenant à des communautés n'ayant pas la même conceptualisation du monde.

[Uitermark, 2001] présente une méthode d'intégration, spécifique aux bases de données géographiques, utilisant une ontologie (figure 2.12). L'auteur emploie les termes d'« ontologie de domaine » et d'« ontologie d'application » mais ce dernier correspond en fait à ce que nous appelons ici le « schéma conceptuel » d'une base de données. Les bases de données géographiques sont considérées comme des représentations du terrain *conceptualisé*, défini comme l'ensemble des objets issu du processus mental d'abstraction que nous opérons sur le terrain réel pour pouvoir le décrire. L'ontologie du domaine est l'ensemble des types d'objets qu'on peut rencontrer sur le terrain, autrement dit l'ensemble des concepts topographiques ; plus précisément, il s'agit ici d'une nomenclature néerlandaise standard appelée *Geo-Information Terrain Model*. Les règles permettant le passage du terrain conceptualisé à sa représentation dans une base de données sont décrites par les *spécifications* (« surveying rules ») de cette base de données ; l'auteur préconise en conséquence de les utiliser pour construire, à partir de l'ontologie du domaine, un « modèle de référence » (correspondant à un schéma intégré), et pour déterminer précisément les liens entre ce modèle et les « ontologies d'application », c'est-à-dire les schémas locaux. De ces liens peuvent alors être déduits des liens directs entre les schémas locaux puis, à l'aide d'un appariement géométrique, entre les instances. Ces liens doivent alors permettre la propagation des mises à jour, but principal de ce processus.

Cette démarche est très proche du processus envisagé dans le cadre de la présente thèse ; cependant l'auteur étudie essentiellement l'intégration de données très détaillées, c'est-à-dire relativement proches de la réalité terrain, et avec par conséquent des spécifications relativement simples. La formalisation de ces spécifications n'est que brièvement évoquée.

2.2.2 Rôle des spécifications dans l'intégration

Dans une base de données, comme le mentionne entre autres [Parent et Spaccapietra, 2000], les informations sur la signification précise des données, nécessaires à l'intégration, sont contenues pour une petite part dans le schéma conceptuel mais l'interprétation correcte de ce schéma repose largement sur l'expertise des administrateurs de la base et de ses concepteurs et sur la documentation rédigée par ces experts.

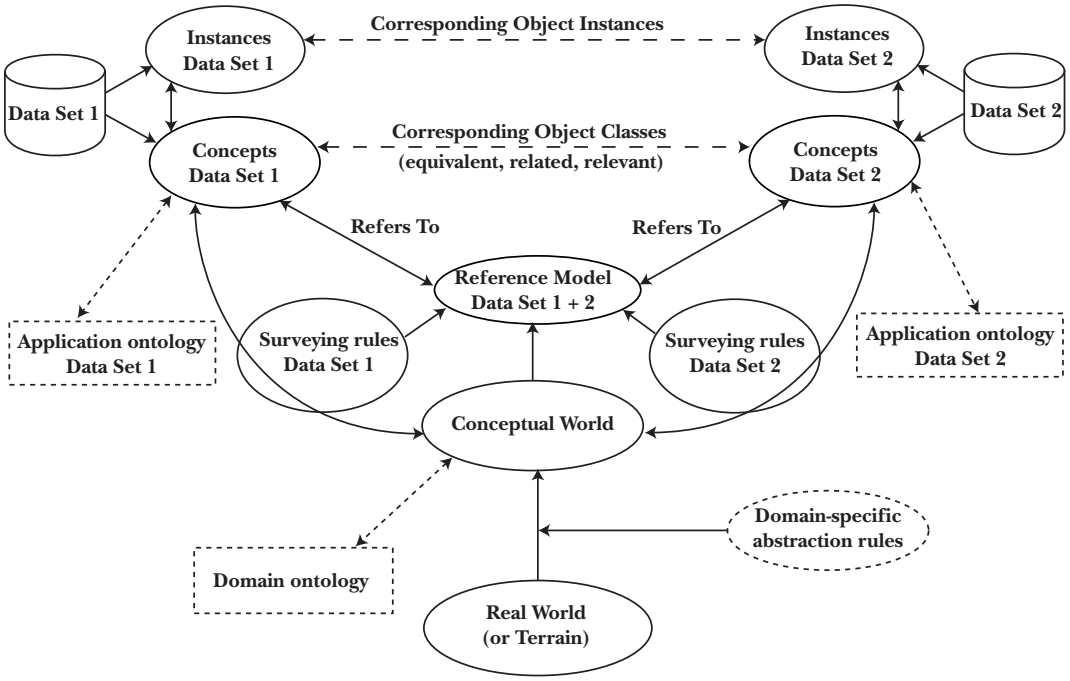


FIG. 2.12 *Schéma du processus d'intégration de [Uitermark, 2001], repris de ce document.*

Dans le domaine de l'information géographique, et plus précisément topographique, la conceptualisation de l'espace géographique, c'est-à-dire le processus mental qui permet de voir le terrain comme un ensemble d'entités géographiques, est liée à un ensemble de connaissances implicites. Mais les règles de correspondance entre cet espace géographique conceptualisé et le contenu théorique de la base de données (c'est-à-dire ce qu'elle contiendrait en l'absence de toute erreur de saisie, qu'on appelle également *terrain nominal* [David et Fasquel, 1997]) sont, elles, explicitées dans les spécifications des bases de données. Ces spécifications sont utilisées principalement par les opérateurs qui saisissent les données mais peuvent aussi l'être par les utilisateurs des données et doivent obligatoirement l'être pour l'intégration de plusieurs bases.

Ces spécifications sont également une source d'information précieuse pour interpréter les différences de représentations entre bases et savoir si celles-ci correspondent ou non à des incohérences [Sheeren *et al.*, 2004]. En effet, étant données deux bases en correspondance, on constate un certain nombre de différences telles par exemple que des objets ne possédant pas d'homologues dans l'autre base. Cette différence peut être normale et due à une différence d'exhaustivité prévue par les spécifications ; on parle alors d'équivalence. Dans le cas contraire, on parle d'incohérence. Les spécifications contiennent des critères de sélection qui peuvent par exemple indiquer que les bâtiments de surface supérieure à un certain seuil sont tous représentés dans l'une des bases. Si l'autre base contient un bâtiment non apparié et permet de calculer sa surface, on peut grâce à cette spécification savoir si cette absence d'appariement indique une incohérence. Les spécifications peuvent également ne pas indiquer de seuil précis mais se contenter d'un critère flou tel que « les

bâtiments suffisamment grands sont saisis ». [Sheeren *et al.*, 2004] propose d'utiliser des techniques d'apprentissage automatique pour déterminer de façon plus précise la signification que les experts du domaine qui saisissent les données accordent à ces critères flous, et également la marge de tolérance avec laquelle ils interprètent les critères précis. Le résultat de cet apprentissage permet tout à la fois de mieux détecter les incohérences et d'expliciter une partie de l'expertise du domaine nécessaire à l'interprétation des spécifications papier. Cependant, pour pouvoir enrichir les spécifications de cette façon, il faut préalablement les représenter de façon informatique. En effet, si la forme actuelle de ces spécifications, de gros documents textuels, est bien adaptée à la consultation par les opérateurs, elle ne l'est pas à la manipulation informatique de leur contenu.

L'une des étapes du processus d'intégration de bases de données est l'appariement de schémas. Cet appariement doit être réalisé en s'appuyant sur la sémantique des données, qui est décrite par la documentation des bases, rédigée par des experts du domaine [Parent et Spaccapietra, 2000]. Plusieurs auteurs mentionnent par ailleurs l'utilité, avant la mise en correspondance des schémas proprement dite, d'une étape de description des liens entre le contenu des bases de données et les concepts du monde réel, ou entre schéma conceptuel et ontologie [Partridge, 2002]. Nous pensons, rejoignant [Uitermark, 2001] sur ce point, que dans le cas des bases de données géographiques ces liens sont décrits par les *spécifications* des bases de données. L'analyse des spécifications est donc une étape cruciale de l'intégration ; leur description formelle pourrait permettre d'automatiser en bonne partie l'étape suivante, celle de la déclaration des correspondances entre schémas.

Le travail présenté dans ce mémoire consiste à étudier comment une telle description formelle pourrait être réalisée.

Chapitre 3

Présentation du modèle de représentation des spécifications

Où l'on cherchera à déterminer une structure de représentation des spécifications commune permettant de faire ressortir automatiquement les convergences et les divergences entre les représentations du terrain stockées dans les différentes bases de données.

Nous nous concentrerons tout d'abord, en 3.1, sur la structure générale des spécifications. Actuellement (3.1.1), celles-ci sont organisées selon le schéma de la base de données qu'elles spécifient ; chaque classe a sa spécification. Cette spécification comprend généralement deux parties : des critères de sélection visant à définir précisément l'ensemble des entités que la classe est censée représenter, puis des règles de modélisation indiquant comment on détermine, pour chacune de ces entités, la géométrie et les autres attributs des objets qui la représenteront.

La description de la représentation en fonction du terrain, à l'aide de règles de sélection et de modélisation, est commune à toutes nos spécifications ; elle est relativement intuitive : on dit ce qu'on représente et comment on le représente, et cette structure ne dépend pas de la représentation considérée, nous proposons donc de la conserver. En revanche, l'organisation des spécifications en fonction du schéma de la base pose plusieurs problèmes. Tout d'abord, puisque toutes les bases ont un schéma différent, elle signifie que toutes les spécifications ont une structure différente, ce qui va à l'encontre du but d'intégration. D'autre part, elle signifie que la façon *dont le terrain est représenté (le « comment ») interfère dans la description de* ce *qui est représenté (le « quoi ») ; si cette situation ne pose pas de problème particulier au lecteur humain, elle compliquerait beaucoup l'élaboration d'un langage formel de représentation des spécifications qui chercherait à conserver cette structure.*

Nous proposerons donc en 3.1.2 d'organiser les spécifications formalisées non en fonction des classes des bases, mais en fonction de concepts géographiques communs indépendants de la représentation, c'est-à-dire d'appuyer la structure du modèle non sur un schéma conceptuel de base de données mais sur une ontologie du domaine. Les spécifications sont alors représentées sous forme de liens entre cette ontologie et le schéma : elles décrivent comment les concepts géographiques sont représentés par les classes de la base.

Nous chercherons ensuite en 3.2 à déterminer les briques fondamentales nécessaires à la description de ces liens, c'est-à-dire les opérations élémentaires communes qui interviennent fréquemment dans la modélisation des entités géographiques.

Nous proposerons alors en 3.3 un langage permettant, en combinant ces opérations élémentaires, de décrire les règles de sélection et de modélisation associées à chaque type d'entité géographique pour chacune des bases.

Enfin, nous discuterons en 3.4 de l'instanciation du modèle, c'est-à-dire de la construction des spécifications formalisées à partir des spécifications papier.

Nous allons proposer dans ce chapitre un modèle formel de représentation des spécifications de bases de données géographiques vectorielles, telles que celles produites par l'Institut géographique national. Ce modèle visera à permettre le traitement par un ordinateur des spécifications formalisées et à faciliter l'analyse des différences de représentations entre les bases, en vue de leur intégration au sein d'une base multi-représentations.

Il s'agit donc de définir une structure commune à toutes les spécifications, qui permettra de distinguer clairement les éléments communs et les éléments spécifiques de chaque représentation. Par exemple, si la base A sélectionne tous les bâtiments de plus de 30 m^2 et la base B tous les bâtiments de plus de 50 m^2, on veut que le modèle mette en évidence qu'on a le même type de critère de sélection dans les deux bases (les bâtiments de superficie suffisamment importante sont retenus), avec simplement une différence de seuil. Le modèle doit également permettre à un logiciel de déduire facilement d'une telle règle que tous les bâtiments sélectionnés dans la base B le sont aussi (théoriquement, c'est-à-dire d'après les spécifications) dans la base A.

Nous nous appuierons pour cela sur la structure des spécifications telles qu'elles existent actuellement, en en conservant les aspects indépendants de la représentation et en adaptant les autres.

3.1 Structure générale

3.1.1 Situation actuelle

Nous utiliserons principalement dans cette partie des exemples tirés des spécifications de la BDTopo Pays version 1.2 [IGN, 2002] et de la BDCarto version 2.5 [IGN, 2004]. Nous ferons autant que possible référence aux versions externes, disponibles publiquement, de ces documents ; il en existe également des versions internes à l'IGN mais les différences, présentation mise à part, ne sont pas très importantes. Elles sont liées soit à la gestion interne de la base de données — ainsi l'attribut « identifiant BDNyme » du cours d'eau BDTopo Pays, qui fait référence à une autre base de données, est remplacé dans le produit externe par un attribut « nom » qui contient le toponyme — soit aux limitations ou variantes possibles liées aux formats d'export des données — ainsi les spécifications externes de la BDTopo Pays précisent que dans les formats d'export 2D, les tronçons de cours d'eau sont munis d'attributs « Z_initial » et « Z_final » pour restituer l'information altimétrique.

Normes ISO pour la structure des spécifications

Notons qu'il existera bientôt une norme internationale relative aux spécifications de bases de données géographiques (ISO 19131, à paraître en 2006 ou 2007) ; cette norme vise à unifier la structure générale des spécifications, et son utilisation pour leurs versions futures est actuellement étudiée à l'IGN. Si l'organisation actuelle des documents n'est pas encore normalisée, on y trouve cependant déjà toujours à peu près les éléments recommandés par l'ISO, plus d'autres plus spécifiques, dans une première partie correspondant à la présentation globale du produit : extension géographique, système de référence géodésique et type de projection utilisé, spécifications de qualité (précision géométrique, actualité des données...)

Ces généralités ne constituent qu'une toute petite partie des spécifications. L'essentiel de celles-ci est regroupé par la norme ISO 19131 dans une seule partie « data content and structure ». Pour les bases de données à objets, cas qui nous intéresse (« feature-based data product » dans la terminologie ISO[1], par opposition à « coverage-based » qui désigne les bases de données à champs continus), cette partie comprend le schéma de la base (« application schema ») et la description de tous les éléments de celui-ci (« feature catalogue »). Pour la représentation du schéma, la norme ISO 19109 propose un modèle de données spécifique, le « general feature model ». Il s'agit d'un modèle orienté objet complet, décrit en UML, avec associations et liens de généralisation-spécialisation ; notons toutefois qu'en pratique les bases de données que nous connaissons contiennent peu d'associations et aucun lien de généralisation-spécialisation : il s'agit essentiellement de classes indépendantes, qui par ailleurs n'ont que des attributs et pas d'opérations. La représentation formelle, graphique ou non, du schéma est en fait pour l'instant, contrairement à la norme, rarement présente dans les spécifications papier. Elle n'est en particulier pas dans les spécifications externes, car la structure précise de la base de données dépend du format de livraison et n'est donc pas toujours totalement conforme au schéma conceptuel interne, sur lequel s'appuient les spécifications ; la structure exacte des jeux de données livrés est décrite dans des documents séparés.

Presque tout le contenu des spécifications correspond donc à ce que l'ISO appelle « feature catalogue » : la description détaillée de la sémantique des classes de la base, de leurs attributs et des quelques relations entre elles[2]. Il existe également une norme pour la constitution de ce catalogue : ISO 19110 ; cette norme indique une liste de champs descriptifs à remplir pour chaque élément du schéma, permettant notamment de donner des noms alternatifs aux classes, ou encore de lister les valeurs possibles d'un attribut énuméré avec, pour chacune d'elles, un code et une description de sa signification. L'essentiel de la sémantique est contenu dans le champ « définition » et, dans une moindre mesure, dans le champ facultatif « contrainte », qui ne sont pas formalisés plus avant (il s'agit dans les deux cas de texte libre).

Par ailleurs, la norme précise que « les critères de saisie utilisés pour identifier individuellement les phénomènes du monde réel à représenter comme objets dans un jeu de données ne sont pas spécifiés par cette norme »[3] et qu'ils doivent être inclus « séparément » dans les spécifications. Le but est probablement de permettre le partage d'un même catalogue pour une série de produits à différents niveaux de détail qui possèdent des classes similaires, avec les mêmes attributs thématiques, et diffèrent uniquement par les critères de sélection et de modélisation géométrique. En effet, les spécifications allemandes d'ATKIS[4], par exemple, utilisent ce principe, et c'est aussi le cas des spécifications mexicaines de l'INEGI[5], que nous avons pu consulter.

1. L'ISO emploie le terme « feature » (ou « feature instance ») pour désigner les objets ou entités géographiques ; « feature type » désigne une classe d'objets géographiques.

2. « The feature catalogue defines the meaning of the feature types and their associated feature attributes, feature operations and feature associations contained in the application schema » (ISO 19110).

3. « The collection criteria used to identify individual real world phenomena and to represent them in a dataset are not specified in this International Standard. »

4. disponibles en ligne : `http://www.atkis.de/dstinfo`

5. également disponibles en ligne :
`http://mapserver.inegi.gob.mx/geografia/espanol/normatividad/diccio/`

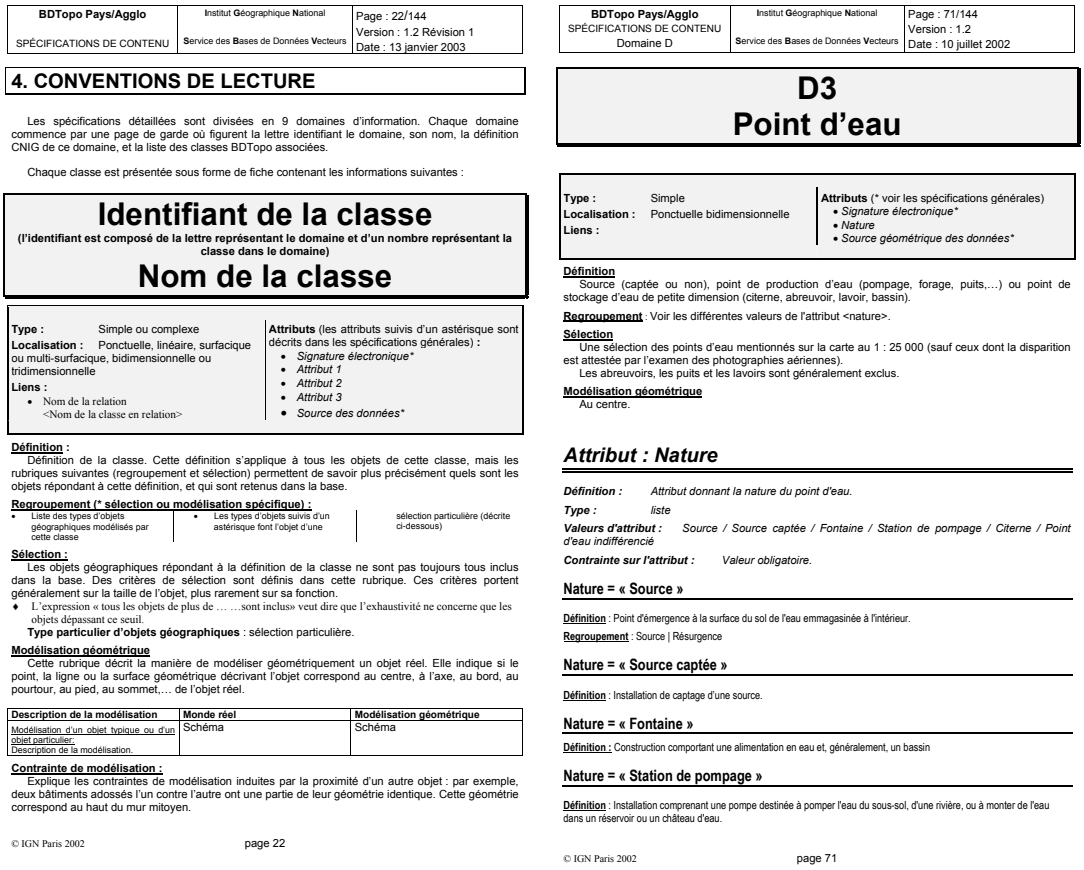
Toutes les classes n'existent pas dans toutes les bases de données, et tous les attributs des classes non plus, mais un bon nombre d'entre elles sont communes et certains de leurs attributs également, tandis que les critères de sélection et de modélisation géométrique diffèrent. Par exemple, sur le site web d'ATKIS, on peut consulter ce qui s'appelle « Objektartenkatalog » (OK) (cette dénomination correspond exactement à « feature catalogue ») ; pour chacune des classes (Objektarten), on peut consulter ses spécifications pour chacun des niveaux de détail (OK25, OK50, OK250, OK1000). Si l'on regarde la classe « cours d'eau » (Wasserlauf), on peut constater que la définition de la classe est toujours la même. Cependant, les critères de sélection indiquent pour OK1000 que les cours d'eau ne sont saisis qu'à partir de 2000 m de long, sauf s'ils sont importants pour la connexité du réseau ; pour OK250, ils sont saisis à partir de 1000 m ; pour OK50, ils sont tous saisis à quelques exceptions près (qui sont décrites). De la même façon, la modélisation diffère suivant le niveau de détail : dans OK1000 les cours d'eau ne sont représentés par des surfaces qu'à partir d'une largeur de 200 m, dans OK250 le seuil est 42 m et dans OK50 12 m.

Notons cependant que des critères de sélection et de modélisation différents signifient que les classes sont elles-mêmes différentes, quand bien même elles partagent le même nom et, en partie, les mêmes attributs : elles ne représentent pas le même ensemble d'entités du monde réel ni les mêmes caractéristiques des entités. Cela signifie que ce que l'ISO appelle définition de la classe doit être une définition très générale, les critères de saisie devant la compléter et la préciser ; la limite entre définition et critères de saisie est a priori laissée au choix du rédacteur. Ainsi, dans ATKIS, les trois niveaux les moins détaillés utilisent tous la même classe Wasserlauf avec des critères différents, mais le niveau le plus détaillé, OK25, utilise à la place et pour représenter les mêmes entités la classe « Strom, Fluß, Bach » (torrent, rivière, ruisseau).

Structure actuelle des spécifications de l'IGN

Les spécifications de l'IGN sont globalement organisées d'une façon similaire à ce que propose l'ISO : on a une partie générale, puis la description des classes du schéma, avec une fiche par classe. Ces fiches sont classées par thèmes. Chacune de ces fiches comprend le nom de la classe, sa définition générale, les éventuelles relations la concernant, les critères de sélection restreignant la définition générale pour décrire précisément l'ensemble des entités que la classe est censée représenter, puis les attributs et leurs descriptions (voir exemple figure 3.1). Les spécifications de l'INEGI comportent aussi une fiche par classe ; les spécifications d'ATKIS également, mais ces dernières comportent en plus une fiche pour chaque thème, avec des règles de modélisation applicables à tout le thème.

Si, contrairement à ce que dit l'ISO, les critères de sélection et de modélisation ne sont pas « séparés » du reste des spécifications et font partie intégrante de la description des classes, la distinction entre définition générale et critères de sélection est cependant systématique. L'idée est en fait que la définition définit non la classe de la base mais le concept géographique qu'elle est censée représenter. Les critères de sélection indiquent alors, parmi les entités géographiques correspondant à cette définition, lesquelles seront représentées par des objets de la base ; les règles de modélisation indiquent enfin comment on détermine la géométrie et les valeurs d'attributs de chacun de ces objets en fonction de l'entité qu'il représente. L'ensemble de ces deux opérations, sélection et modélisation, peut être considéré comme un unique



BDTopo Pays/Agglo	Institut Géographique National	Page : 22/144
SPÉCIFICATIONS DE CONTENU	Service des Bases de Données Vecteurs	Version : 1.2 Révision 1
		Date : 13 janvier 2003

4. CONVENTIONS DE LECTURE

Les spécifications détaillées sont divisées en 9 domaines d'information. Chaque domaine commence par une page de garde où figurent la lettre identifiant le domaine, son nom, la définition CNIG de ce domaine, et la liste des classes BDTopo associées.

Chaque classe est présentée sous forme de fiche contenant les informations suivantes :

Identifiant de la classe
(l'identifiant est composé de la lettre représentant le domaine et d'un nombre représentant la classe dans le domaine)
Nom de la classe

Type : Simple ou complexe
Localisation : Ponctuelle, linéaire, surfacique ou multi-surfacique, bidimensionnelle ou tridimensionnelle
Liens :
- Nom de la relation <Nom de la classe en relation>

Attributs (les attributs suivis d'un astérisque sont décrits dans les spécifications générales) **:**
- *Signature électronique**
- *Attribut 1*
- *Attribut 2*
- *Attribut 3*
- *Source des données**

Définition :
Définition de la classe. Cette définition s'applique à tous les objets de cette classe, mais les rubriques suivantes (regroupement et sélection) permettent de savoir plus précisément quels sont les objets répondant à cette définition, et qui sont retenus dans la base.

Regroupement (* sélection ou modélisation spécifique) :
- Liste des types d'objets géographiques modélisés par cette classe
- Les types d'objets suivis d'un astérisque font l'objet d'une sélection particulière (décrite ci-dessous)

Sélection :
Les objets géographiques répondant à la définition de la classe ne sont pas toujours tous inclus dans la base. Des critères de sélection sont définis dans cette rubrique. Ces critères portent généralement sur la taille de l'objet, plus rarement sur sa fonction.
- L'expression « tous les objets de plus de … …sont inclus» veut dire que l'exhaustivité ne concerne que les objets dépassant ce seuil.

Type particulier d'objets géographiques : sélection particulière.

Modélisation géométrique
Cette rubrique décrit la manière de modéliser géométriquement un objet réel. Elle indique si le point, la ligne ou la surface géométrique décrivant l'objet correspond au centre, à l'axe, au bord, au pourtour, au pied, au sommet,… de l'objet réel.

Description de la modélisation	Monde réel	Modélisation géométrique
Modélisation d'un objet typique ou d'un objet particulier: Description de la modélisation.	Schéma	Schéma

Contrainte de modélisation :
Explique les contraintes de modélisation induites par la proximité d'un autre objet : par exemple, deux bâtiments adossés l'un contre l'autre ont une partie de leur géométrie identique. Cette géométrie correspond au haut du mur mitoyen.

BDTopo Pays/Agglo	Institut Géographique National	Page : 71/144
SPÉCIFICATIONS DE CONTENU	Service des Bases de Données Vecteurs	Version : 1.2
Domaine D		Date : 10 juillet 2002

D3
Point d'eau

Type : Simple
Localisation : Ponctuelle bidimensionnelle
Liens :

Attributs (* voir les spécifications générales)
- *Signature électronique**
- *Nature*
- *Source géométrique des données**

Définition
Source (captée ou non), point de production d'eau (pompage, forage, puits,…) ou point de stockage d'eau de petite dimension (citerne, abreuvoir, lavoir, bassin).

Regroupement : Voir les différentes valeurs de l'attribut <nature>.

Sélection
Une sélection des points d'eau mentionnés sur la carte au 1 : 25 000 (sauf ceux dont la disparition est attestée par l'examen des photographies aériennes).
Les abreuvoirs, les puits et les lavoirs sont généralement exclus.

Modélisation géométrique
Au centre.

Attribut : Nature

Définition : *Attribut donnant la nature du point d'eau.*
Type : *liste*
Valeurs d'attribut : *Source / Source captée / Fontaine / Station de pompage / Citerne / Point d'eau indifférencié*
Contrainte sur l'attribut : *Valeur obligatoire.*

Nature = « Source »

Définition : Point d'émergence à la surface du sol de l'eau emmagasinée à l'intérieur.
Regroupement : Source | Résurgence

Nature = « Source captée »

Définition : Installation de captage d'une source.

Nature = « Fontaine »

Définition : Construction comportant une alimentation en eau et, généralement, un bassin

Nature = « Station de pompage »

Définition : Installation comprenant une pompe destinée à pomper l'eau du sous-sol, d'une rivière, ou à monter de l'eau dans un réservoir ou un château d'eau.

FIG. 3.1 *Conventions de lecture des spécifications de la BDTopo Pays, et exemple de la classe « point d'eau ».*

processus de *représentation* qui va du terrain vers la base de données (figure 3.2) : l'ensemble des entités géographiques correspondant à la définition est représenté par un ensemble d'objets de la base de données.

Problèmes posés par cette structure relativement à notre objectif

Le principe consistant, pour décrire le contenu de la base de données, à décrire le processus de représentation, est commun à toutes nos spécifications et nous semble relativement intuitif : le contenu de la base de données est fonction du contenu du terrain, et on décrit cette fonction[1], qui est le lien entre les deux.

En revanche, l'organisation générale des spécifications selon le schéma de la base de données, avec une fiche par classe, pose plusieurs problèmes relativement à notre

1. Il ne s'agit en fait pas d'une fonction à strictement parler, dans la mesure où il peut exister des choix de représentation arbitraires qui, à partir d'un même terrain, mènent à plusieurs représentations légèrement différentes mais toutes également valides et conformes aux spécifications ; mais l'idée est celle d'une fonction.

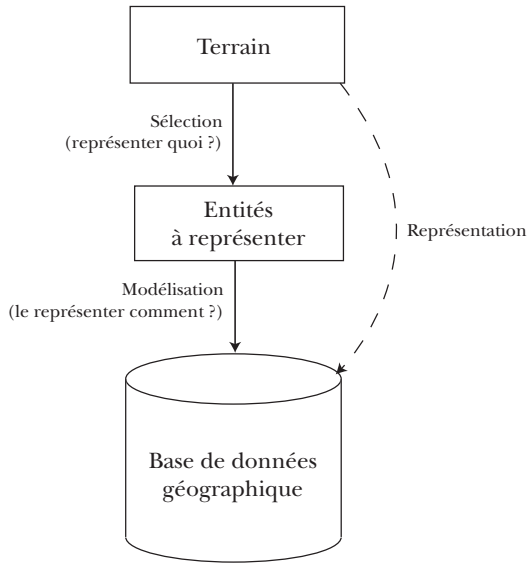


FIG. 3.2 *Processus schématisé de représentation du terrain dans une base de données géographique.*

objectif. Elle signifie que le terrain conceptualisé doit être classifié, dès le départ, selon ce schéma, puisque la fiche d'une classe décrit comment on représente les entités géographiques *de la classe*. Or les schémas des bases, même s'il s'agit de schémas conceptuels, sont réalisés pour structurer les données et non pour classifier les entités géographiques. Cela a pour conséquence, entre autres, que des regroupements plus ou moins arbitraires peuvent être faits, qui diffèrent suivant les bases. Ainsi, les aqueducs appartiennent pour la BDTopo Pays à la classe « canalisation », avec les oléoducs, et pour la BDCarto à la classe « tronçon hydrographique », avec les rivières. Il est gênant, pour notre but d'intégration, que ces différences de regroupement influent sur la structure même des spécifications dès le plus haut niveau.

D'autre part, certaines classes ne correspondent pas réellement à des entités bien définies a priori, en raison de critères de découpage notamment. Ainsi, dans la BDCarto, la classe « tronçon de route » a pour définition générale « portion connexe de route, de chemin, de piste cyclable, ou de sentier, homogène pour les relations la mettant en jeu et pour les attributs qu'elle porte. » Or cette définition se réfère aux attributs et aux relations concernant l'objet de la base de données, c'est-à-dire le résultat de la modélisation, pour définir l'entité qui est censée être le point de départ de cette modélisation. Ceci ne pose aucun problème à un lecteur humain des spécifications : on comprend que les attributs sont calculés tout le long de la route et que celle-ci est découpée en morceaux où ils sont homogènes; le découpage et le calcul des attributs sont ainsi faits simultanément. C'est-à-dire qu'on ne peut déterminer exactement ce qu'est, sur le terrain, un tronçon de route au sens défini dans les spécifications avant de l'avoir modélisé.

Mais cela signifie que les règles de représentation influent sur la conceptualisation du terrain. Si l'on conserve cette structure, cela complique la représentation formelle des règles de sélection et de modélisation puisqu'elles ne partent pas d'un ensemble

d'entités bien défini ; en effet, il peut exister plusieurs façons différentes de découper les routes en tronçons qui soient toutes conformes aux spécifications. Considérons par exemple cet extrait de spécification de la BDCarto : « Les tronçons retenus sont : [...] les tronçons de voies carrossables [...], à l'exception des culs-de-sac de moins de 1000 mètres de long [...]. » Soit maintenant la situation de la figure 3.3, où l'on a sur le terrain un cul-de-sac en forme de Y tel que le tronçon unique obtenu en ôtant l'une ou l'autre des deux branches dépasse 1000 mètres, mais qu'aucun des trois tronçons formant les branches du Y n'atteigne cette longueur. Les schémas 1, 2 et 3 correspondent à trois possibilités de saisie ; les représentations 2 et 3 sont a priori toutes deux autorisées (car le tronçon est suffisamment long) mais la représentation 1 est interdite puisqu'elle comprend des tronçons trop courts. Il est impossible d'exprimer simplement cette règle si l'on considère que l'on fait une sélection sur l'ensemble des tronçons de route : les tronçons des représentations 2 et 3 sont bien tous deux des tronçons vérifiant la définition, mais ils ne peuvent exister simultanément.

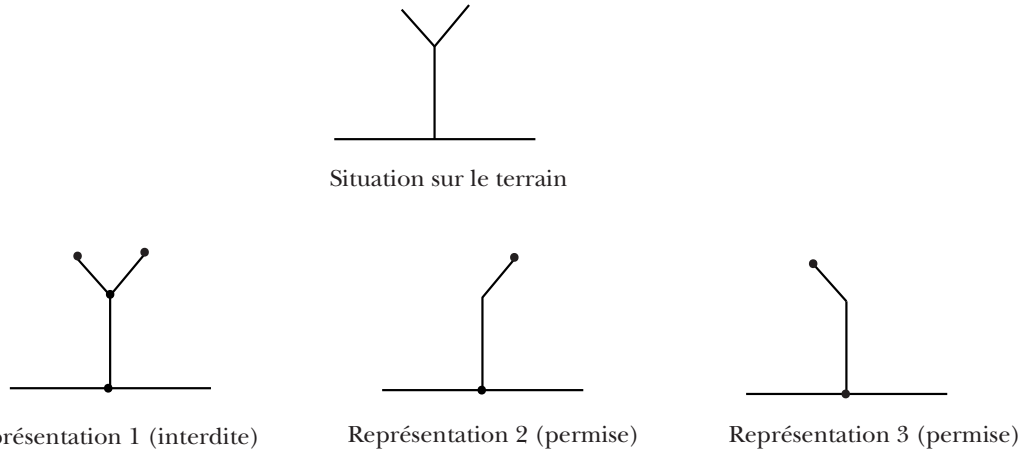


FIG. 3.3 *Exemple montrant la difficulté de définir* a priori *l'ensemble des entités géographiques correspondant à une classe donnée, en l'occurrence « tronçon de route ».*

Nous proposons donc, pour la représentation formelle des spécifications, de conserver l'idée qu'on décrit des règles de représentation indiquant comment on passe du terrain à la base de données mais d'utiliser pour les entités géographiques une classification indépendante des schémas des bases de données. Il s'agit, plutôt que de s'appuyer sur les classes du schéma conceptuel de données, de partir de concepts géographiques autant que possible objectifs et partagés par les différentes bases ; c'est-à-dire d'une ontologie du domaine. Les problèmes cités plus haut peuvent d'ailleurs s'exprimer en disant que les schémas des bases de données ne sont pas *ontologiques*.

3.1.2 Structure proposée

Comme nous l'avons vu au chapitre 1, l'esprit humain ne manipule pas directement le terrain dans toute sa complexité mais une représentation abstraite, conceptualisée, de celui-ci, composée d'entités géographiques que l'on sait reconnaître sans pour autant savoir décrire le processus mental permettant cette reconnaissance. Les spécifications, pour décrire le processus de représentation du terrain, s'appuient évidemment sur cette conceptualisation. Nous appelons terrain *conceptualisé* l'ensemble

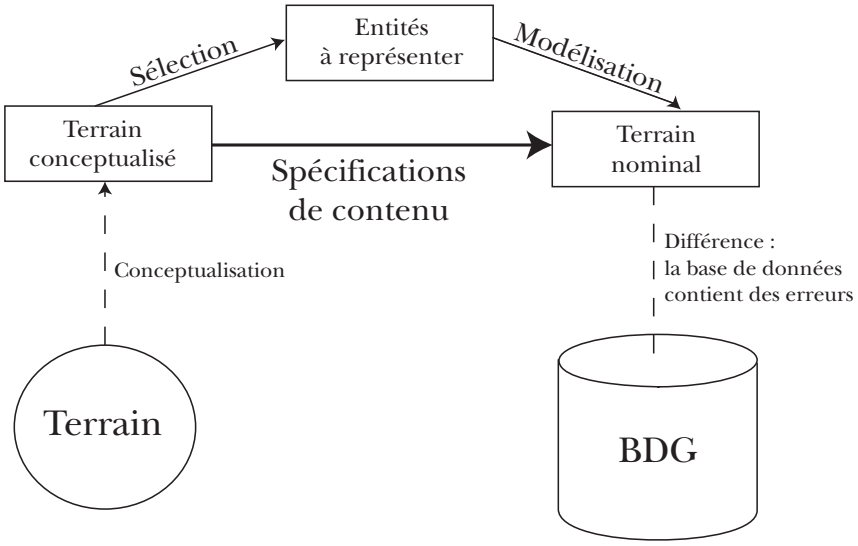


FIG. 3.4 *Les spécifications de contenu décrivent le passage du terrain conceptualisé au terrain nominal.*

des entités géographiques qu'un observateur humain sait reconnaître sur le terrain ; ces entités géographiques sont des instances de concepts géographiques, que nous appellerons aussi types d'entités géographiques. Ainsi, Paris, la Seine, le rocher du zoo de Vincennes sont des entités géographiques ; bâtiment, pré, clôture sont des types d'entités géographiques. Par ailleurs, rappelons que les spécifications de contenu ne décrivent pas le contenu effectif de la base de données mais son contenu théorique, appelé *terrain nominal* ; les données ne correspondent au terrain nominal que dans la mesure où il n'y a pas eu d'erreur de saisie. Les spécifications de qualité visent à décrire le lien entre terrain nominal et base de données ; les spécifications de contenu, celles auxquelles nous nous intéressons ici, peuvent être considérées comme décrivant le lien entre terrain conceptuel et terrain nominal (figure 3.4).

Notre modèle de représentation des spécifications s'appuie sur une ontologie du domaine pour la description de la structure du terrain conceptualisé, et sur le schéma conceptuel de données pour la description de la structure du terrain nominal ; les spécifications elles-mêmes sont représentées au sein de règles de représentation qui font le lien entre des types d'entités géographiques, éléments de l'ontologie, et des classes de la base, éléments du schéma conceptuel. Nous représentons les types d'entités géographiques comme des classes UML munies du stéréotype « entité géographique » et les classes du schéma de la base comme des classes UML munies du stéréotype « objet de la base ». La figure 3.5 montre un exemple où l'on a en bleu l'ontologie et en rouge le schéma de la base de données ; les traits pointillés verts indiquent les liens de représentation entre classes de l'ontologie et classes de la base[1]. Pour plus de clarté, nous utiliserons désormais les conventions typographiques

1. Les liens verts que nous avons indiqués sur ce diagramme ne tiennent pas compte des dépendances entre les représentations, liées aux relations entre classes internes aux bases. Ainsi, dans la BDTopo Pays, un [cours d'eau nommé] est composé de [tronçons de cours d'eau], mais nous avons néanmoins relié

suivantes au sein du texte : les noms d'éléments de l'ontologie (entités géographiques ou propriétés) seront écrits en PETITES CAPITALES et les noms d'éléments du schéma d'une base entre crochets [comme ceci].

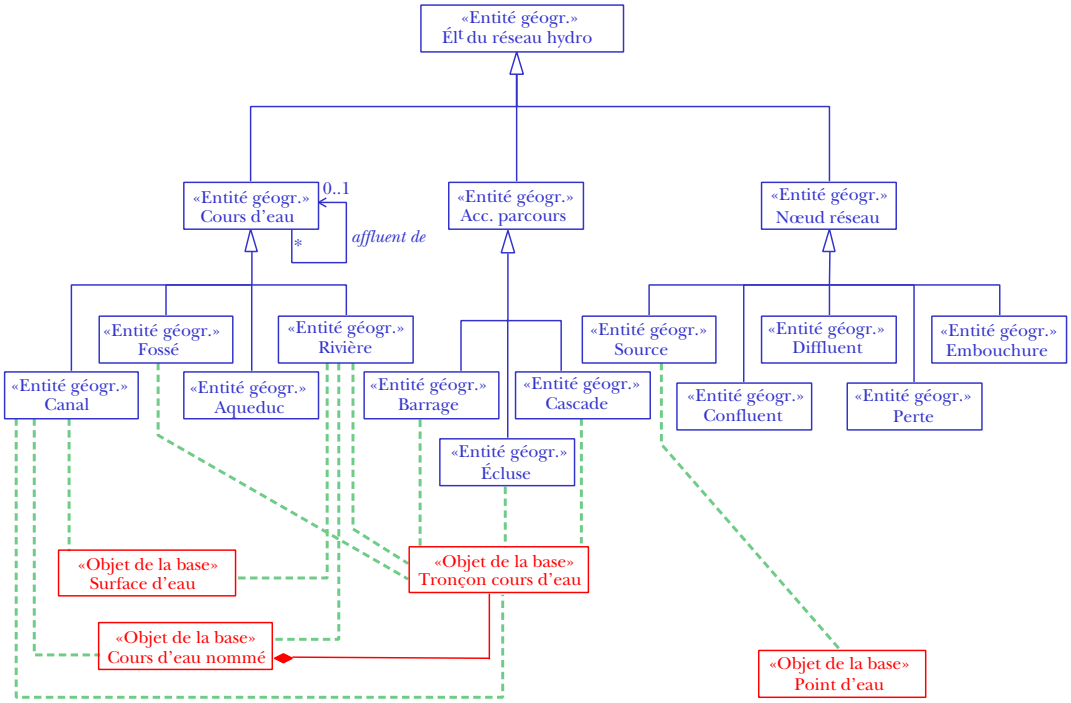


FIG. 3.5 *Le réseau hydrographique dans la BDTopo Pays.*

La sémantique de ces liens est représentée par des règles de représentation. Une règle de représentation part d'une ou plusieurs classes de l'ontologie et les relie à une ou plusieurs classes du schéma de la base ; elle indique comment à partir d'une entité géographique instance d'une des classes de l'ontologie concernées on crée dans la base des objets, instances des classes du schéma concernées. Cette règle doit donc indiquer, d'une part, sous quelle condition des instances sont créées dans la base, autrement dit quelles entités sont retenues (sélection) ; et, d'autre part, comment on affecte à ces instances leurs valeurs d'attributs, y compris la géométrie (modélisation). Pour ce faire, elle doit s'appuyer sur les propriétés de l'entité ; chaque type d'entité géographique possède un certain nombre de propriétés qu'on définit dans l'ontologie. Par exemple, un COURS D'EAU peut être NAVIGABLE ou non, il peut être PERMANENT ou INTERMITTENT, posséder un TOPONYME, il a une certaine LARGEUR, etc. Notons que ces propriétés ne possèdent pas nécessairement de valeur exacte prédéterminée : leur valeur peut être dépendante du niveau de détail. Par exemple, le contour d'une forêt ne peut pas être défini de façon exacte, mais il peut l'être avec une résolution donnée. Cet aspect est important car il est possible que des attributs dans deux bases de données différentes soient définis de la même façon et à partir des mêmes propriétés sans que cela implique que ces attributs doivent avoir

des types d'entités comme RIVIÈRE à chacune de ces deux classes car il s'agit bien de deux représentations distinctes, même si l'une utilise l'autre : il est nécessaire pour comprendre la représentation des rivières de lire les fiches de spécification correspondant à chacune des deux classes.

toujours la même valeur : il suffit pour cela que la valeur de la propriété considérée ne soit pas déterminée pour le même niveau de détail dans les deux cas. D'autre part, ces propriétés peuvent avoir une valeur variable, qui dépend de la position à l'intérieur de l'entité ; par exemple la LARGEUR d'un COURS D'EAU varie tout au long de son cours, et il peut également être NAVIGABLE à certains endroits et pas à d'autres. Cet aspect n'existe en général pas dans les attributs des objets de la base de données, même si certains modèles conceptuels, comme MADS [Parent *et al.*, 1997], le proposent.

3.2 Structure détaillée : règles de représentation

Nous avons proposé au paragraphe précédent de représenter les spécifications sous forme de règles de représentation reliant des types d'entités géographiques et des classes du schéma de la base de données. Voyons maintenant plus en détail les différents éléments qui devront intervenir dans ces règles.

Sélection

Comme nous l'avons vu, la représentation d'un type d'entité géographique commence toujours par une étape de sélection. Cette sélection s'exprime généralement sous la forme d'une condition plus ou moins complexe que doit vérifier une entité du type considéré pour être représentée dans la base. Ces conditions portent sur divers critères permettant de mesurer l'importance géographique de l'entité ; il s'agit donc d'une combinaison de critères intrinsèques (pour mesurer l'importance absolue de l'objet) et de critères contextuels (pour mesurer son importance dans le paysage). Les critères intrinsèques les plus rencontrés portent sur les dimensions de l'entité : sa superficie, sa largeur, sa longueur, ou sa hauteur par exemple, et donnent des seuils que ces dimensions doivent dépasser. Ce type de critère n'est pas difficile à formaliser. Plus généralement, beaucoup de critères de sélection peuvent être considérés comme des contraintes sur les propriétés de l'entité, en leur imposant ou interdisant une valeur, ou en leur demandant d'être supérieures ou inférieures à un seuil donné. Les quelques critères intrinsèques qui ne peuvent pas être formalisés de cette façon étant a priori soit très spécifiques (comme « l'entité est saisie si elle apparaît sur la carte au 1:25 000 en cours ») soit subjectifs ou flous (« d'aspect remarquable »), il est difficile de définir un cadre exhaustif permettant de tous les formaliser ; l'intérêt en serait par ailleurs limité dans la mesure où de tels critères ont relativement peu de chances d'être comparables entre plusieurs bases de données différentes. Nous proposons donc de les conserver sous forme de texte libre.

Les critères contextuels peuvent être considérés comme des contraintes sur les relations que l'entité doit entretenir avec d'autres entités : ainsi, une ROUTE est plus importante géographiquement si elle mène à un BÂTIMENT que si elle ne mène nulle part ; un BÂTIMENT est plus remarquable s'il est isolé, ce qui peut s'exprimer en disant qu'il est loin de tout autre BÂTIMENT. Ces relations peuvent être très variées, néanmoins il s'agit souvent de relations topologiques, telles que être dans, toucher, traverser, etc., ou de relations métriques, par exemple être plus près ou plus loin qu'un certain seuil d'une entité d'un type donné. Il arrive aussi fréquemment que des critères portent sur l'entité en relation plus que sur l'existence de la relation elle-même, par exemple un des critères de sélection pour les BARRAGES dans la BDCarto

dit que la RETENUE D'EAU générée par le barrage doit avoir une superficie supérieure à 50 hectares. On rencontre également souvent un critère très particulier qui ne porte pas directement sur le terrain mais sur la base de données elle-même : le fait que l'entité en relation soit représentée dans la base en tant que telle, ce qui revient à dire qu'elle vérifie sa propre contrainte de sélection.

Pour la plupart des règles de représentation, les critères de sélection peuvent être représentés comme une contrainte complexe, combinaison de contraintes simples des différents types mentionnés ci-dessus (intrinsèques et contextuelles) reliées entre elles par des opérateurs logiques (« et » et « ou »). Cependant, en ce qui concerne les réseaux, routier, hydrographique, ferré ou autre, il est difficile de découper a priori le réseau en entités bien distinctes ; il existe bien une sélection, comme pour les autres entités, dans la mesure où seule une partie du réseau est retenue, mais cette sélection ne peut s'exprimer que comme une sélection d'ensemble et non entité par entité. Cela ne change pas grand-chose, sauf que les critères de sélection sont en général variables suivant les parties du réseau considérées : il existe généralement une règle de sélection spécifique pour les culs-de-sac, et une autre pour les parcours secondaires, qui sont a priori moins importants géographiquement. Ces deux règles ne s'intègrent pas très bien à une contrainte de sélection générale telle que définie plus haut, car cela nécessiterait de définir à l'intérieur de la contrainte ce que sont un parcours secondaire et un cul-de-sac, pour pouvoir distinguer les cas. Or si l'on a plusieurs parcours d'égale importance, le choix de celui qui est « principal » et de ceux qui sont « secondaires » est arbitraire, et le fait d'être un parcours secondaire ou principal ne s'exprime de façon satisfaisante ni comme un critère intrinsèque ni comme un critère contextuel. Les culs-de-sac posent également problème, comme on l'a montré figure 3.3. Nous proposons donc de représenter séparément la contrainte de sélection des culs-de-sac et celle des parcours secondaires, si elles existent.

Agrégation et découpage

Les contraintes de sélection présentées ci-dessus ne suffisent pas toujours à déterminer exactement ce qui sera représenté dans la base de données. Il arrive en effet fréquemment qu'on ne représente pas directement des entités individuelles mais des morceaux ou des agrégats d'entités. En général, les entités sont découpées en morceaux pour des raisons topologiques (intersections dans un réseau) ou pour refléter les changements de valeur d'un attribut dont la valeur dépend de la position, sachant que les systèmes actuels ne permettent pas à un attribut d'avoir une valeur variable au sein d'un même objet. Lorsqu'on coupe aux intersections, il s'agit en général uniquement des intersections du réseau déjà filtré par la contrainte de sélection, puisque ce découpage est fait pour des raisons de représentation des relations topologiques dans la base. Lorsqu'on coupe aux changements de valeur d'attribut, ce découpage est fait relativement à un certain niveau de détail, c'est-à-dire que les changements ne sont pas pris en compte pour des sections trop petites. Ce niveau de détail peut varier suivant l'attribut considéré, comme on le voit dans cet extrait de spécification de la BDCarto :

Les routes, sentiers et chemins retenus sont découpés en portions connexes portant les mêmes valeurs d'attributs et les mêmes relations. Le changement de valeur des attributs « Nombre de chaussées », « Nombre total de voies », « État physique de la route », « Nombre de voies chaussée montante, chaussée descendante », « Toponyme » et « Date de mise en service » n'entraîne la

création d'un nœud et d'un tronçon supplémentaires que si la nouvelle valeur reste constante sur une longueur d'au moins 1000 mètres ; sinon, il n'est pas tenu compte de ce changement de valeur.

Pour les valeurs 1 et 2 de l'attribut « Position par rapport au sol » (pont, viaduc), ce seuil est fixé à 500 mètres ; pour la valeur 3 de ce même attribut (tunnel, souterrain, couvert), il est fixé à 200 mètres. Il n'y a pas de seuil minimal pour les autres attributs.

À l'opposé, l'agrégation regroupe des entités voisines ayant des propriétés similaires de façon à éviter une représentation inutilement détaillée. Contrairement à ce qui se passe pour le découpage, la sélection est généralement faite *après* l'agrégation, c'est-à-dire qu'elle est faite sur les agrégats et non sur les entités individuelles. Les critères qui entrent en jeu pour déterminer si l'on agrège ou non peuvent être par exemple la distance entre les entités individuelles, les différences de valeur entre leurs propriétés, leur superficie, leur densité, la superficie de l'agrégat. Par exemple, dans la BDTopo Pays, deux BÂTIMENTS sont agrégés si : ils se touchent, ils possèdent les mêmes valeurs des attributs [nature] et [fonction], leur différence de HAUTEUR est inférieure à 10 m et l'un d'eux au moins a une SUPERFICIE inférieure à 400 m^2. Dans la BDCarto, deux FORÊTS sont agrégées si la distance entre les deux est inférieure à 100 m ; des PLANS D'EAU sont agrégés s'ils font moins de 4 ha et que la surface de l'ensemble dépasse ce seuil, les plans d'eau devant recouvrir les trois quarts de cette surface.

Instanciation

Une fois qu'on a clairement déterminé ce qu'il fallait représenter, on décrit dans des règles d'instanciation la façon de créer les objets et de calculer leurs attributs en fonction de l'entité qu'on représente et de ses propriétés.

3.3 Langage de description des règles de représentation

Nous allons maintenant définir, en nous appuyant sur les différents types d'éléments de spécification que nous venons de décrire, un langage formel pour la description des règles de représentation des différents types d'entités géographiques.

Nous appellerons « procédure de représentation » l'ensemble des règles formalisées décrivant complètement comment un type d'entités donné est représenté dans la base. Il peut être utile ou nécessaire de distinguer dans l'ontologie différents types d'entités que les spécifications traitent de la même façon, aussi une même procédure de représentation pourra-t-elle être reliée à plusieurs classes de l'ontologie. D'autre part, il est fréquent que plusieurs types d'entités soient représentés de façon identique à quelques détails près, typiquement une valeur d'attribut. Par exemple, dans la BDTopo Pays, les IMMEUBLES D'HABITATION et les MONUMENTS sont représentés par la même classe [bâtiment] et selon les mêmes règles, la seule différence étant que l'attribut [nature] ne prend pas la même valeur dans les deux cas. Nous proposons dans une telle situation de définir une seule procédure de représentation et d'utiliser une condition sur le type de l'entité (« contrainte de nature », voir 3.3.3) pour faire la distinction là où elle est nécessaire.

L'ensemble des classes de la base concernées par une procédure de représentation donnée pourra, quant à lui, se déduire de l'ensemble des règles d'instanciation contenues dans ladite procédure.

Nous utiliserons pour présenter notre langage une notation de type BNF [Backus *et al.*, 1963], où la barre verticale indique une alternative, avec les extensions proposées par [Wirth, 1977] : les crochets indiquent que l'élément qu'ils entourent est optionnel, les accolades indiquent que l'élément qu'elles entourent peut être répété de zéro à plusieurs fois. La grammaire BNF complète de ce langage est jointe en annexe.

3.3.1 Structure générale

Nous distinguons quatre types de règles de modélisation, correspondant aux différents éléments décrits en 3.2 : règles de sélection, règles de découpage, règles d'agrégation et enfin règles d'instanciation.

Par ailleurs, il peut être nécessaire de définir des « attributs » hors des règles d'instanciation proprement dites ; c'est notamment utile pour les règles de découpage et d'agrégation. Nous utilisons le terme « attribut » de façon générale pour indiquer qu'ils sont définis dans la procédure de représentation, par opposition aux « propriétés » qui appartiennent aux entités géographiques indépendamment de leur représentation. Ces attributs ne correspondent pas nécessairement aux attributs des classes de la base qui interviendront dans les règles d'instanciation, même si c'est généralement le cas ; il s'agit en quelque sorte de variables locales pouvant être utilisées dans les différentes règles de notre procédure de représentation. On considère que ces attributs peuvent, comme les propriétés de l'entité, avoir une valeur qui varie suivant la position. Ils sont définis à partir des propriétés de l'entité ou d'autres attributs ; ces définitions d'attributs constituent un cinquième type de règles. Nous utiliserons dans le texte, pour désigner de tels attributs locaux, une `police à chasse fixe`, pour les distinguer à la fois des PROPRIÉTÉS et des [attributs de la base].

Nous avons donc cinq types de sections possibles dans une procédure de représentation, correspondant aux différents types de règles. Chacune de ces sections constitue un élément de la procédure de représentation.

```
élément_représentation ::= attributs | sélection | découpage
   | agrégation | instanciation
```

Une procédure de représentation est composée de plusieurs de ces sections, dans un ordre qui peut éventuellement varier. Une nouvelle section est introduite par le nom du type de section suivi de deux points (par exemple « attributs : » ou « instanciation : »).

```
procédure_représentation ::= { élément_représentation }
```

Nous allons maintenant décrire les différents types de règles définis dans notre langage, type de section par type de section. Deux types importants d'éléments de ce langage ne sont pas liés à un type de section en particulier : les expressions et les contraintes. Les expressions servent soit à définir un attribut local soit à définir un attribut de la base de données, dans une règle d'instanciation ; nous les décrirons en 3.3.2. Les contraintes servent à exprimer des conditions portant sur une entité géographique ; nous les décrirons en 3.3.3.

3.3.2 Attributs et expressions

Une section *attributs* est composée de définitions d'attributs.

```
attributs ::=
   "attributs :" définition_attribut { ";" définition_attribut }
définition_attribut ::= identifiant_attribut "=" expression
```

Un attribut est défini par une expression. A priori, les expressions mathématiques ne sont pas utiles pour les spécifications de bases de données géographiques ; nous avons donc uniquement des expressions simples, qui sont soit des valeurs littérales soit des références à d'autres attributs ou propriétés, et des expressions conditionnelles. Les expressions conditionnelles servent à affecter à l'attribut des valeurs différentes suivant que l'entité vérifie ou non une certaine contrainte (voir la définition des contraintes au paragraphe suivant). Une référence à une propriété ou un autre attribut se fait grâce à son identifiant (c'est-à-dire son nom[1]) ; nous désignons dans la description BNF du langage un tel identifiant par « identifiant_attribut » mais il peut également désigner une propriété. Cet identifiant doit avoir été préalablement défini, soit dans l'ontologie, s'il s'agit d'une propriété, soit dans une section *attributs* de la procédure de représentation en cours s'il s'agit déjà d'un attribut calculé. Dans le cas d'une propriété, il doit s'agir d'une propriété que possèdent tous les types d'entités géographiques concernés par la procédure de représentation en cours.

```
expression ::=
   identifiant_attribut ["(" précisions ")"]
   | valeur_littérale
   | expression_conditionnelle
   | "(" expression ")"

expression_conditionnelle ::=
   "Si" contrainte "Alors" expression "Sinon" expression
```

La référence à un attribut ou une propriété peut inclure entre parenthèses des précisions sur la façon dont on détermine sa valeur, par exemple à quelle résolution on la calcule. Nous n'avons pour l'instant pas formalisé ces précisions, qui sont cependant indispensables pour assurer une bonne interprétation des spécifications : il s'agit d'une chaîne de caractères.

Les valeurs littérales peuvent être des chaînes de caractères, des valeurs booléennes (vrai ou faux) ou des valeurs numériques, correspondant ou non à des grandeurs physiques, éventuellement munies d'une unité (qui est une chaîne de caractères) si c'est le cas.

```
valeur_littérale ::= chaîne | valeur_numérique | valeur_booléenne
valeur_numérique ::= nombre [unité]
valeur_booléenne ::= "Vrai" | "Faux"
unité ::= chaîne
```

1. Dans notre langage, un identifiant peut être n'importe quelle suite de lettres minuscules non accentuées et de blancs soulignés (_) ; les mots réservés, quant à eux, commencent tous par des majuscules.

Exemple : On veut représenter la définition des attributs [largeur] et [navigabilité] du [tronçon de cours d'eau] dans la BDCarto. Ces attributs sont utilisés dans les règles de découpage des COURS D'EAU et doivent donc être définis, dans la procédure de représentation, avant lesdites règles. Ces attributs sont ainsi définis dans les spécifications actuelles :

Largeur :

1. *entre 0 et 15 m*
2. *entre 15 et 50 m*
3. *plus de 50 m*

Navigabilité :

1. *navigable : inscrite à la nomenclature des voies navigables*
2. *non navigable*
3. *déclassé : voies anciennement navigables non entretenues par l'état*

On suppose que LARGEUR et NAVIGABLE sont des propriétés des COURS D'EAU définies dans l'ontologie, dont les valeurs sont respectivement numérique et booléenne. On peut alors écrire dans la procédure de représentation, avec des expressions conditionnelles[1] :

```
attributs :
   largeur_discrete = Si largeur < 15 "m" Alors 1
      Sinon Si largeur < 50 "m" Alors 2
      Sinon 3;
   navigabilite = Si navigable = Vrai Alors 1
      Sinon Si Vérifie "anciennement navigable" Alors 3
      Sinon 2
```

3.3.3 Contraintes

Les contraintes ne font pas l'objet d'une section spécifique dans la procédure de représentation mais peuvent être utilisées dans tous les types de règles, aussi nous leur consacrons un paragraphe spécifique avant de décrire les quatre types de sections restants. Nous appelons « contrainte » une condition qui peut être vérifiée ou non par une entité géographique.

Nous distinguons plusieurs types de contraintes élémentaires, correspondant aux différents types de critères mentionnés en 3.2 : des contraintes portant sur l'entité elle-même et des contraintes sur son contexte, que nous appellerons contraintes de relation. Plus précisément, nous allons définir dans ce paragraphe les types de contraintes suivants :

— contrainte sur propriété, simple ou complexe,

— contrainte de nature,

— contrainte descriptive,

— contrainte de relation,

1. Un attribut local ne peut porter le même nom qu'une propriété, on doit donc utiliser pour représenter l'attribut [largeur] un nom comme `largeur_discrete`, par exemple, plutôt que `largeur`.

— contrainte « base de données ».

Parmi les contraintes sur l'entité, on a les contraintes portant sur des propriétés de l'entité ou des attributs définis dans une section *attributs* de la procédure de représentation en cours. Ces contraintes sur propriété peuvent porter sur des propriétés ou attributs simples, dont la valeur est numérique ou booléenne ou encore textuelle (par exemple un toponyme). Dans ce cas, la contrainte consiste simplement à comparer la valeur de la propriété ou de l'attribut à une valeur seuil.

Notons que les opérateurs de comparaison > et < n'ont de sens que dans le cas d'un attribut (ou propriété) numérique. D'autre part, nous ne proposons pas les opérateurs $\leq$ et $\geq$ car s'il s'agit d'une valeur continue (a priori une grandeur physique), étant donné que le point de départ est le monde réel, la distinction entre inégalité large ou stricte n'a pas de sens (on ne peut pas mesurer avec une précision infinie) ; et s'il s'agit d'une valeur discrète, telle qu'un entier, l'inégalité large peut s'exprimer comme une inégalité stricte avec une autre valeur seuil.

```
contrainte_sur_propriete_simple ::=
   identifiant_attribut [ "(" précisions ")" ]
       opérateur_comparaison valeur_littérale
opérateur_comparaison ::= ">" | "<" | "=" | "=/="
```

Les contraintes sur propriété peuvent également porter sur des propriétés plus complexes représentant par exemple des parties de l'entité, comme le TOIT d'un BÂTIMENT ou le MUR D'ENCEINTE d'une PROPRIÉTÉ. En ce cas, nous proposons de considérer la valeur de cette propriété comme une entité géographique et d'appliquer une contrainte à cette entité. Par exemple, si l'on veut classer d'une façon particulière les PROPRIÉTÉS qui possèdent un MUR D'ENCEINTE de hauteur supérieure à 5 m, on utilisera : `mur_d_enceinte.hauteur > 5 "m"`.

```
contrainte_sur_propriété_complexe ::=
   identifiant_attribut"."contrainte_élémentaire
```

Nous avons également les contraintes de nature, qui imposent à l'entité d'appartenir à un certain type ; elles sont nécessaires car on permet à une même procédure de représentation de s'appliquer à plusieurs types différents, voir 3.3. La référence à un type d'entité géographique se fait par son identifiant, défini dans l'ontologie.

```
contrainte_nature ::= "Est" identifiant_type_eg
```

On y ajoute les contraintes descriptives, qui permettent d'exprimer sous forme de caractères ce qu'on ne sait pas formaliser autrement.

```
contrainte_descriptive ::= "Vérifie" chaîne
```

Les contraintes de relation, quant à elles, portent sur le contexte de l'entité : elles indiquent que l'entité doit entretenir une certaine relation avec une autre entité dont on précise le type et à laquelle on peut éventuellement demander de vérifier elle-même une certaine contrainte. Par exemple, si l'on veut qu'une entité soit située dans une AGGLOMÉRATION de plus d'un million d'habitants, on pourra utiliser : `Relation (agglomeration nombre_d_habitants > 1000000, Inclusion)`.

```
contrainte_relation ::=
   "Relation (" identifiant_type_eg [contrainte]
       "," type_relation ")"
```

La contrainte de relation est considérée comme vérifiée par une entité E si *il existe* une entité F du type spécifié, différente de E, vérifiant la contrainte éventuelle, telle que la relation ait lieu entre E et F. Par conséquent, si une contrainte de relation est niée (E ne doit pas vérifier la contrainte), cela signifie que *quelle que soit* l'entité F du type indiqué vérifiant la contrainte éventuelle, différente de E, la relation ne doit pas avoir lieu entre E et F. Ainsi, pour exprimer qu'une entité doit être à au moins 100 mètres de toute HABITATION, on écrira `Non Relation (habitation, Distance < 100 "m")` : l'entité ne doit pas être à moins de 100 m d'une HABITATION.

Parmi les contraintes de relation possibles, on distingue les relations topologiques et métriques. Les relations métriques consistent à comparer la valeur d'un certain critère métrique, mesurée entre les deux entités, à une valeur seuil. Les autres types de relation sont décrits par des chaînes de caractères.

Pour déterminer la liste des relations topologiques possibles, on peut s'appuyer sur la méthode des neuf intersections proposée par [Egenhofer et Herring, 1990], qui permet de distinguer huit types de relations. Cependant, nous n'avons pas inclus ici tous ces types de relations car certains nous semblent peu utiles dans un contexte de spécifications s'appuyant sur le monde réel[1]. Nous définissons ainsi dans notre langage les relations d'adjacence (partage de frontière sans intérieur commun), superposition (égalité topologique des surfaces couvertes par chacune des deux entités), inclusion dans chacun des deux sens (l'entité qu'on traite est incluse dans l'entité en relation, ou contient l'entité en relation), et enfin intersection (les deux entités se rencontrent mais on ne précise pas de quelle façon)[2].

Pour les relations métriques, nous n'avons pas pour le moment défini de liste exhaustive des critères métriques utilisables; nous définissons « distance » et « dénivelée » à titre d'exemple, mais d'autres critères pourraient probablement être ajoutés à cette liste. Notons toutefois que nous n'avons pas rencontré dans les spécifications actuelles de contrainte de relation métrique faisant appel à un autre critère que la distance.

```
type_relation ::=
   relation_topologique | relation_métrique | relation_autre
relation_topologique ::= "Adjacence" | "Superposition"
   | "Intersection" | "Inclus" | "Contient"
```

1. Plus précisément, nous ne faisons pas la distinction entre inclusion dans l'intérieur d'une région et inclusion avec partage de frontière, car la frontière d'une entité au sens topologique n'est pas déterminable dans la réalité. Cependant, nous conservons la relation d'adjacence (partage de frontière sans intérieur commun), car l'*existence* d'une frontière entre deux entités disjointes peut-être déterminée sans devoir pour cela déterminer la frontière elle-même.

2. Notons que les relations définies par la méthode des neuf intersections sont mutuellement exclusives, il n'existe donc pas de relation d'*intersection* à proprement parler, mais une relation de *chevauchement* (overlap) qui n'est pas considérée comme vérifiée si l'une des régions est complètement incluse dans l'autre. Notre but ici n'étant pas de classer les relations topologiques en catégories disjointes mais d'exprimer des contraintes, nous pensons plus utile de définir simplement la relation d'intersection, en considérant que l'inclusion est un cas particulier d'intersection (il reste de toute façon possible d'exprimer par une contrainte complexe la relation « intersection sans inclusion »).

```
relation_métrique ::=
   critère_métrique opérateur_comparaison valeur_littérale
critère_métrique ::= "Distance" | "Dénivelée"
relation_autre ::= chaîne
```

Enfin, la contrainte « base de données » permet d'indiquer que l'entité doit être représentée dans la base comme objet d'une certaine classe ; on peut la considérer comme une référence à une autre procédure de représentation. La classe de la base concernée est indiquée par son identifiant, qui doit être défini dans le schéma de la base. Cette contrainte est typiquement utilisée comme contrainte sur l'entité en relation, à l'intérieur d'une contrainte de relation : par exemple, dans la BDCarto, les CHEMINS sont sélectionnés entre autres s'ils « suivent le tracé d'une voie antique (dont les critères de sélection sont donnés dans la description de l'objet complexe « itinéraire routier ») ». On écrira pour représenter ceci : `Relation (voie_antique Sélectionné comme itineraire_routier, Inclusion)`.

```
contrainte_bd ::= "Sélectionné comme" identifiant_classe_base
```

Une contrainte est composée de contraintes élémentaires reliées par des opérateurs logiques.

```
contrainte ::=
   ["Non"] contrainte_élémentaire
   { ("Et" | "Ou") contrainte_élémentaire }
contrainte_élémentaire ::=
   "(" contrainte ")"
   | contrainte_sur_propriété_simple
   | contrainte_sur_propriété_complexe
   | contrainte_nature
   | contrainte_descriptive
   | contrainte_relation
   | contrainte_bd
```

Exemple : On veut représenter la contrainte de sélection des BÂTIMENTS dans la BDTopo Pays, qui est ainsi libellée :

Tous les bâtiments de plus de 50 m^2 sont inclus. Les bâtiments faisant entre 20 et 50 m^2 sont sélectionnés en fonction de leur environnement (les petits bâtiments [situés à] plus de 100 m d'une habitation sont inclus, alors que les petits bâtiments situés en ville ne le sont pas [...]) et de leur aspect (les petits bâtiments d'aspect précaire [...] sont exclus). Les bâtiments de moins de 20 m^2, [...] s'ils sont spécifiquement désignés sur la carte au 1:25 000 en cours, sont représentés [...]. Toutes les constructions de plus de 50 m de haut [sont incluses].

On suppose pour l'exemple que « petit bâtiment d'aspect précaire » est exactement recouvert par le type d'entité géographique CABANE, défini dans l'ontologie. On suppose d'autre part que SUPERFICIE et HAUTEUR sont des propriétés définies dans l'ontologie. Cette contrainte peut alors s'exprimer comme contrainte complexe de la façon suivante :

```
surface > 50 "m2"
Ou (superficie > 20 "m2"
   Et (Non Relation (habitation, Distance < 100 "m"))
```

```
   Et (Non Est cabane))
Ou Vérifie "spécifiquement désigné sur la carte au 1:25000 en cours"
Ou hauteur > 50 "m"
```

3.3.4 Sélection

Une section *sélection* contient une contrainte générale optionnelle, plus éventuellement des contraintes spécifiques pour les culs-de-sac et les parcours secondaires. Rappelons que, comme nous l'avons dit au 3.2, nous introduisons ces contraintes spécifiques parce que la contrainte générale ne permet pas d'exprimer de façon satisfaisante les critères de sélection pour les entités formant un réseau. Il est possible que d'autres types d'entités, que nous n'avons pas spécifiquement étudiés dans cette thèse, nécessitent d'autres types de critères spécifiques, qu'il faudrait alors ajouter à cette liste.

```
sélection ::= "sélection :"
   [ contrainte ]
   [ "culs-de-sac :" contrainte ]
   [ "parcours secondaires :" contrainte ]
```

Exemple : Dans la BDCarto, les [tronçons hydrographiques] (portions de RIVIÈRE, de RUISSEAU ou de CANAL) retenus sont :

— *tous les axes principaux, y compris dans la zone d'estran et dans les zones de marais, à l'exception des « culs-de-sac » d'une longueur inférieure à 1000 mètres [...];*

— *outre l'axe principal, les axes des bras secondaires d'une longueur supérieure à 1000 mètres ou qui délimitent une île d'une superficie supérieure à 10 hectares quand un cours d'eau se subdivise en plusieurs.*

On pourra exprimer ceci de la sorte :

```
sélection :
   culs-de-sac : longueur > 1 "km"
   parcours secondaires : longueur > 1 "km"
      Ou Relation (ile superficie > 10 "ha", Adjacence)
```

3.3.5 Découpage

On distingue trois critères de découpage : on peut couper aux intersections du réseau (ce n'est pas toujours le cas), couper lorsque l'entité rencontre un certain obstacle, représenté par une entité géographique d'un type donné devant éventuellement vérifier une certaine contrainte, et enfin couper aux changements de certains attributs avec éventuellement une longueur minimale (ou éventuellement une surface) sur laquelle le changement doit avoir lieu pour être pris en compte. Les critères de découpage sont souvent compliqués, aussi nous laissons la possibilité d'ajouter des précisions sous forme de texte libre entre crochets. Il est possible de définir un découpage par plusieurs règles, séparées par des points-virgules, pour dire par exemple qu'on coupe aux intersections et aussi aux changements d'attributs.

```
règle_découpage ::=
   "Intersections"
   | "Rencontre" identifiant_type_eg [ contrainte ]
   | "Changement" [ ">" valeur_numérique ] [ "[" précisions "]" ]
       "(" identifiant_attribut { "," identifiant_attribut } ")"
```

Le découpage permet de passer de l'entité géographique de départ à un ensemble de *sections* et de *limites de sections* (en général des « tronçons » et des « nœuds », car le découpage se fait sur des objets de forme générale linéaire, mais il pourrait éventuellement s'agir de zones et de frontières). On peut alors donner un ensemble de règles de représentation pour les sections et un autre ensemble de règles pour les limites de sections, comme s'il s'agissait d'entités géographiques définies dans l'ontologie. En effet, il est fréquent qu'on ait dans la base de données d'une part une représentation des sections et d'autre part une représentation des limites. Notons que les sections obtenues par découpage aux changements de la valeur d'un attribut donné doivent être considérées comme possédant désormais une valeur homogène de cet attribut, même si ce n'est pas le cas en réalité à cause du seuil de prise en compte des changements de valeur. Après la mention « Fin découpage », on considère de nouveau les entités de départ. Cette mention permet de décrire dans une même procédure de représentation plusieurs représentations d'une même entité utilisant des découpages différents.

```
découpage ::= "découpage :" règle_découpage { ";" règle_découpage }
   [ "sections :" { élément_représentation } ]
   [ "limites :" { élément_représentation } ]
   "Fin découpage"
```

Exemple : Dans la BDCarto, les COURS D'EAU sont représentés par des objets surfaciques là où ils sont larges, ce qui correspond à un découpage uniquement selon la largeur : seules les sections de plus de 50 mètres de large sont représentées ainsi (ce qui correspond à la valeur 3 de l'attribut `largeur_discrète` défini dans l'exemple du 3.3.2), et seulement si leur SUPERFICIE est au moins de 4 hectares. Ils sont d'autre part représentés par un ensemble de tronçons et de nœuds, avec un découpage aux intersections du réseau, aux changements d'attributs de plus d'un kilomètre et lors de la rencontre de cascades[1]. On suppose que CASCADE est un type d'entité géographique. On aura donc :

```
découpage : Changement (largeur_discrete)
   sections :
      sélection : largeur_discrete = 3 Et superficie > 4 "ha"
      [...]
Fin découpage
découpage :
   Changement > 1 "km" (largeur_discrete, navigabilite);
   Intersections;
   Rencontre cascade;
   sections : [...]
   limites : [...]
Fin découpage
```

1. Nous avons simplifié ce dernier critère pour la clarté de l'exemple.

3.3.6 Agrégation

Les critères employés pour décider si l'on agrège plusieurs entités sont similaires aux contraintes utilisées ailleurs en ce qu'ils peuvent être décomposés en critères élémentaires reliés par des opérateurs logiques. La différence est qu'ils ne portent pas sur une unique entité mais sur un ensemble d'entités du même type.

Ces critères peuvent porter sur les différences entre les entités, en leur imposant d'avoir la même valeur d'un attribut ou d'une propriété donné, ou une différence de valeur inférieure à un certain seuil. Il peut également s'agir de contraintes que chacune des entités doit vérifier, ou que l'ensemble des entités à agréger doit vérifier. Typiquement, les entités individuelles doivent être petites et l'ensemble doit être suffisamment grand. Enfin, il peut s'agir d'une contrainte de relation métrique entre les entités ; typiquement, elles doivent être suffisamment proches les unes des autres.

```
contrainte_agrégation ::=
   ["Non"] contrainte_agr_élém { ("Et" | "Ou") contrainte_agr_élém }

contrainte_agr_élém ::=
   "(" contrainte_agrégation ")"
   | "Chacun" contrainte_élémentaire
   | "Ensemble" contrainte_élémentaire
   | "Même" identifiant_attribut
   | "Différence" identifiant_attribut
       opérateur_comparaison valeur_numérique
   | relation_métrique
```

Exemple : Pour les BÂTIMENTS dans la BDTopo (critère décrit au 3.2), on aura :

```
agrégation : Même nature Et Même fonction
   Et Différence hauteur < 10 "m"
   Et (Non Chacun superficie > 400 "m2")
   Et Distance = 0
```

L'agrégation permet de passer de l'ensemble d'entités de départ à un ensemble d'agrégats ; comme pour le découpage, on peut alors donner des règles de représentation devant s'appliquer aux agrégats comme s'il s'agissait d'entités individuelles. Après la mention « Fin agrégation », on considère de nouveau les entités de départ.

```
agrégation ::= "agrégation :" contrainte_agrégation
       { élément_représentation } "Fin agrégation"
```

3.3.7 Instanciation

Les règles d'instanciation permettent de décrire à quelles conditions et comment des objets de différentes classes sont créés pour représenter l'entité qu'on considère, et comment les attributs de ces objets éventuels sont calculés. Les conditions qui interviennent dans les règles d'instanciation ont un but différent des contraintes de sélection : en effet, une entité qui a été sélectionnée peut encore être représentée de différentes façons suivant les cas. Typiquement, elle sera représentée par un objet ponctuel ou linéaire si elle est petite ou peu large, et par

un objet surfacique si elle est plus étendue. Des règles d'instanciation conditionnelles sont donc nécessaires. Par exemple, pour un BÂTIMENT dans la BDTopo Pays : `Si superficie > 20 "m2" Alors Instancie batiment [...] Sinon Instancie construction_ponctuelle [...]`

```
instanciation_conditionnelle ::=
   "Si" contrainte "Alors" règle_instanciation
       [ "Sinon" règle_instanciation ]
```

L'entité peut également être représentée par plusieurs objets simultanément, auquel cas on écrit les différentes règles d'instanciation les unes après les autres en les séparant par des points-virgules. Par exemple, dans la BDCarto, les FORÊTS sont représentées par des [zones d'occupation du sol], et également par des objets de classe [massif boisé] si elles sont très grandes. On aura ainsi : `Instancie zone_d_occupation_du_sol [...] ; Si superficie > 500 ha Instancie massif_boise [...]`

```
liste_règles_instanciation ::=
   règle_instanciation { ";" règle_instanciation }
```

Une règle d'instanciation élémentaire indique quelle classe on instancie et comment on détermine ses valeurs d'attributs; à chaque attribut est associée une expression (voir 3.3.2). Les attributs sont désignés par leurs identifiants, qui doivent être définis dans le schéma de la base de données et appartenir à la classe correspondante; il ne s'agit pas d'attributs locaux à la procédure de modélisation définis dans une section *attributs*, ni de propriétés de l'entité. Les attributs peuvent bien sûr *correspondre* à des attributs locaux préalablement définis; dans ce cas, on l'écrit explicitement. Par exemple : `Instancie troncon_hydrographique (largeur = largeur_discrete, navigabilite = navigabilite, geometrie = axe)` où `largeur_discrete` et `navigabilite` sont des attributs locaux à la procédure de représentation tandis qu'AXE est une propriété provenant de l'ontologie.

```
instanciation_attribut ::= identifiant_attr_bd "=" expression
règle_instanciation ::=
   "Instancie" identifiant_classe_base
       "(" instanciation_attribut {"," instanciation_attribut} ")"
   | instanciation_conditionnelle
   | "(" liste_règles_instanciation ")"
```

Une section *instanciation* contient une liste de règles d'instanciation.

```
instanciation ::= "instanciation :" liste_règles_instanciation
```

3.4 Instanciation du modèle

3.4.1 Constitution de l'ontologie

Stratégies

Différentes stratégies sont envisageables pour la constitution de l'ontologie [Jones *et al.*, 1998; Noy et McGuinness, 2001]. L'une d'elles consiste à partir d'une nomenclature existante; c'est la démarche proposée par [Uitermark, 2001], qui s'appuie sur

un standard néerlandais, le « Geo-information Terrain Model » (GTM). Une autre consiste à définir une ontologie *ad hoc* pour les spécifications, par exemple à partir de mots-clefs comme nous l'avons fait dans l'exemple.

D'autre part, plusieurs stratégies sont envisageables pour l'intégration [Wache *et al.*, 2001] : on peut chercher à obtenir une ontologie unique pour les spécifications de toutes les bases à intégrer, ou bien établir une ontologie par base puis faire des liens entre elles. Notre modèle est utilisable dans les deux cas. Notons que les langages permettant la description d'ontologies peuvent être plus ou moins riches, notamment en ce qui concerne la définition des concepts et les relations entre eux. La plupart de ces langages sont des variantes des *logiques de description* [Baader *et al.*, 2003]. L'expressivité plus ou moins grande du langage et la façon de définir les concepts peuvent rendre plus ou moins facile l'établissement de liens entre ontologies pour la stratégie des ontologies multiples. Notre modèle, quant à lui, ne nécessite rien de plus que des concepts possédant des propriétés et des liens is-a (liens de généralisation-spécialisation) entre ces concepts, et peut donc être utilisé avec pratiquement n'importe quel type d'ontologie.

La stratégie de l'ontologie unique nous semble *a priori* possible pour l'ensemble des bases de données vecteur de l'IGN : les spécifications de ces différentes bases ont été rédigées par des personnes ayant reçu une formation similaire et possédant une culture d'entreprise commune. Cette stratégie est probablement plus délicate à mettre en œuvre pour des bases de données provenant d'organismes différents, surtout si ces organismes ont des cultures cartographiques différentes ; de telles différences pourraient entre autres provenir de la taille du pays et des types de paysages qu'on y rencontre. On pourrait, si l'on souhaite intégrer beaucoup de bases de données, envisager une stratégie mixte : l'utilisation de plusieurs ontologies, mais une par organisme plutôt qu'une par base, par exemple. Cette stratégie pourrait éventuellement mener à différents niveaux d'intégration : intégration forte (bases de données fédérées) au sein d'un organisme, intégration plus faible entre les différents organismes.

Dans notre exemple, nous avons cherché à appliquer la stratégie de l'ontologie unique ; nous allons discuter les principaux problèmes rencontrés.

Méthode utilisée pour notre exemple

L'ontologie utilisée dans l'exemple donné figure 3.5, que nous décrirons plus avant en 3.4.3, a été constituée manuellement à partir de mots-clefs relevés dans les spécifications. En effet, l'ontologie du domaine doit regrouper les concepts partagés par les différents experts du domaine. Ces concepts constituent un univers du discours commun, donc correspondent au vocabulaire utilisé pour rédiger des documents techniques devant être compréhensibles par tous, tels que les spécifications. Les spécifications que nous avons utilisées pour retrouver ce vocabulaire provenant toutes de l'IGN, il était de plus peu probable de rencontrer des désaccords sur le vocabulaire, puisque les experts les ayant rédigées ont une culture d'entreprise commune.

Nous avons donc simplement parcouru l'ensemble des spécifications du thème hydrographie des deux bases de données en notant tous les mots-clefs. Ces mots-clefs se trouvent un peu partout dans le texte (voir exemple figure 3.6). Nous avons ensuite organisé les termes ainsi obtenus en différentes catégories, pour obtenir une structure hiérarchique plus facile à manipuler : éléments du réseau hydrographique

2. SEMANTIQUE

2.1. CLASSES D'OBJETS SIMPLES

D-s-1 Tronçon hydrographique

a Définition - sélection

Un tronçon hydrographique correspond à l'axe du lit d'une rivière, d'un ruisseau ou d'un canal.

La BDCarto contient :

1. sur le territoire national

   tous les axes principaux, y compris dans la zone d'estran et dans les zones de marais, à l'exception des "culs-de-sac" d'une longueur inférieure à un kilomètre sauf s'ils appartiennent à un cours d'eau d'une longueur supérieure à un kilomètre ;

   outre l'axe principal, les axes des bras secondaires d'une longueur supérieure à un kilomètre ou qui délimitent une île d'une superficie supérieure à dix hectares quand un cours d'eau se subdivise en plusieurs.

2. à l'étranger

   tous les tronçons hydrographiques qui assurent la continuité, vers l'amont ou vers l'aval, du réseau du territoire national ;

   tous les tronçons de canaux navigables ;

   les tronçons des cours d'eau importants.

La continuité du réseau est assurée lors de la traversée de plans d'eau, de zones de marais, de drainage, d'agglomérations.

b Géométrie - construction

Les tronçons hydrographiques sont localisés par des arcs géométriques (relation **[1,n][0,1]**).

Les éléments du réseau d'hydrographie sont découpés en portions ayant les mêmes attributs. Le changement de valeur d'un attribut n'entraîne la création d'un tronçon que si la nouvelle valeur reste la même sur une longueur d'au moins un kilomètre ; sinon, le tronçon précédent est prolongé.

Couche géométrique : hydrographie linéaire.

c Attributs

[1] Etat

- **0-** inconnu : l'existence d'un écoulement est certaine, mais le tracé n'est pas connu avec précision.
- **1-** continu
- **2-** intermittent
- **3-** fictif : assure la continuité de l'écoulement à l'intérieur des zones d'hydrographie (poste 51 des zones d'occupation du sol O-s-1), lorsque le tracé n'est pas connu avec précision.
- **4-** abandonné, à sec

[2] Largeur

- **S-** sans objet (seulement si l'état est inconnu ou fictif)
- **1-** entre 0 et 15 m
- **2-** entre 15 et 50m
- **3-** plus de 50 m

FIG. 3.6 *Extrait des spécifications de la classe [tronçon hydrographique] de la BDCarto, où nous avons entouré les mots-clefs du domaine.*

(cours d'eau, source, etc.), équipements hydrographiques (château d'eau, station de traitement...), tout ceci définissant des types d'entités géographiques. D'autres termes correspondent plutôt à des propriétés, comme artificiel, navigable, intermittent, touristique. D'autres encore décrivent des relations, comme affluent.

Remarquons qu'il n'y a pas a priori de raison pour interdire à un même concept d'appartenir à plusieurs catégories différentes ; en d'autres termes, l'héritage multiple est possible dans le schéma de l'ontologie. Par exemple, un aqueduc peut être à la fois un cours d'eau, un équipement hydrographique et une canalisation. L'avantage de permettre ceci est que la hiérarchie de classification et de sous-classification des concepts géographiques dépend du point de vue adopté, et qu'en privilégier une serait un choix arbitraire. Par ailleurs, si nous avons établi notre liste de concepts en relevant les mots-clefs sans nous préoccuper a priori de la représentation desdits concepts, l'établissement des liens is-a entre concepts peut gagner à prendre en compte l'étape suivante, celle des liens entre ontologie et schéma. En effet, les classes de la base regroupent souvent un certain nombre de concepts, et ces regroupements sont rarement complètement arbitraires : ils correspondent au point de vue adopté par les concepteurs de la base de données, et il nous semble donc cohérent d'en tenir compte pour la réalisation de l'ontologie. Certains regroupements peuvent néanmoins être dus uniquement à des critères d'implémentation, c'est ainsi qu'on a dans la BDTopo Pays des classes [construction ponctuelle], [construction linéaire] et [construction surfacique] ; dans ce cas, il n'est pas souhaitable d'en tenir compte dans l'ontologie.

Évolution de l'ontologie

Même dans le cas où l'on souhaite réaliser une ontologie unique, comme dans notre exemple, on peut construire d'abord une ontologie pour une base et ensuite étendre ou raffiner cette ontologie pour l'autre base. Cela peut permettre également d'ajouter une base de données qui n'était pas prévue dès le départ ; en effet, il est peu probable qu'une ontologie constituée pour une ou deux bases contienne absolument tous les concepts nécessaires à une troisième. Si l'on modifie l'ontologie après avoir modélisé des spécifications, il peut être nécessaire d'adapter certains des liens ; en principe, les procédures de représentation elles-mêmes ne devraient pas avoir à être modifiées, sauf éventuellement les critères de sélection. L'ajout de concepts précédemment absents de l'ontologie et sans rapport avec les concepts présents ne devrait pas poser de problème. En revanche, la définition de nouveaux sous-concepts de concepts existants peut en poser. Si l'on est amené à raffiner un concept qui est relié à une procédure de représentation, il faudrait vérifier que la procédure de représentation s'applique bien à tous les nouveaux sous-concepts. Ce n'est pas nécessairement le cas puisque les définitions peuvent être relativement floues ; et l'ajout de sous-concepts est une façon de modifier la définition, en la précisant.

3.4.2 Liens entre ontologie et schéma

Une fois l'ontologie constituée, il reste à déterminer les liens entre les classes de la base et les concepts de l'ontologie. Ces liens ne peuvent être établis qu'en partant du schéma de la base, en raison de la structure des spécifications papier ; c'est la

raison pour laquelle cette étape est nécessaire avant de passer à la rédaction des procédures de représentation qui, elles, partent de l'ontologie. Cette étape ne pose pas a priori de problème majeur, puisque la liste des types d'entités du monde réel qu'une classe de la base est susceptible de représenter est normalement indiquée dans les spécifications de cette classe. Cependant différents choix sont parfois possibles.

Il arrive fréquemment — au moins dans les spécifications de la BDTopo Pays — qu'une classe soit associée à un concept plutôt vague, tel que BÂTIMENT, mais que la définition de la classe soit ensuite précisée par une liste de concepts plus précis, qui peut être ou non limitative. Cette liste correspond souvent aux différentes valeurs possibles d'un attribut énuméré. Si la liste est limitative, on peut soit relier la procédure de représentation individuellement à tous les sous-concepts, mais pas au concept général, soit la relier au concept général mais avec une contrainte de sélection énumérant la liste des sous-concepts représentés. En effet, la relier au concept général sous-entend a priori qu'elle s'applique à tous les sous-concepts (puisqu'un lien is-a indique qu'une entité appartenant au sous-type appartient toujours aussi au type général). Rappelons que nous avons introduit une contrainte de nature pour permettre la distinction entre plusieurs types d'entités à l'intérieur d'une procédure de représentation (voir l'introduction du paragraphe 3.3) ; la deuxième solution (relier la classe au concept général) consiste à utiliser ces contraintes de nature dans les critères de sélection.

La première solution (ne relier la classe qu'aux sous-concepts) a l'avantage qu'on n'a pas besoin de regarder la contrainte de sélection pour savoir si un sous-concept est ou non concerné. Dans le cas où le sous-concept est aussi un sous-concept d'une autre hiérarchie — par exemple, un CHÂTEAU D'EAU est à la fois un BÂTIMENT, un RÉSERVOIR et un POINT DE REPÈRE — elle permet également de rendre plus directement visible le lien (figure 3.7). Ainsi, si l'on regarde les sous-concepts de BÂTIMENT, on pourra voir que certains bâtiments sont représentés dans la BDTopo Pays par des objets de classe [réservoir]. La deuxième solution obligerait non seulement à regarder les représentations de tous les sous-concepts de BÂTIMENT mais également à regarder celles de tous les autres super-concepts de chacun de ces sous-concepts, puis à regarder pour chacun d'eux les contraintes de sélection afin de vérifier que le concept qui nous intéresse n'est pas en fait exclu de cette représentation. Par exemple, un CHÂTEAU D'EAU n'est jamais représenté dans la BDTopo Pays par un objet de classe [bâtiment]. Or si on s'intéresse à la représentation des RÉSERVOIRS, le fait que CHÂTEAU D'EAU soit un sous-type de BÂTIMENT laissera supposer a priori qu'il est concerné, et donc que certains RÉSERVOIRS sont concernés, par la procédure de représentation relative à la classe [bâtiment], et seuls les critères de sélection montreront que ce n'est en fait pas le cas.

Notons qu'il est possible de transformer automatiquement la solution utilisant les contraintes de nature en celle reliant la procédure de représentation seulement à chacun des sous-concepts : il suffit pour cela d'interpréter les contraintes de nature. Il est d'ailleurs également possible, au moins théoriquement, de supprimer complètement les contraintes de nature (c'est-à-dire même dans les règles de modélisation) en transformant une procédure de représentation en autant de procédures qu'il y a de types d'entités géographiques concernés. Ainsi, même si la solution de relier la procédure de représentation au concept général présente des inconvénients, elle représente bien la même information que l'autre, mais de manière moins explicite.

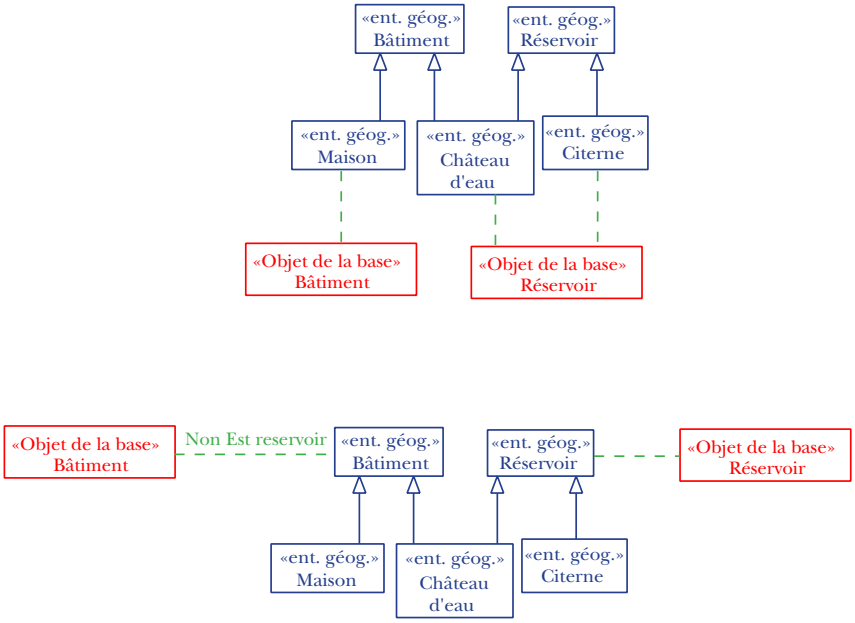


FIG. 3.7 *Deux possibilités pour l'établissement des liens schéma-ontologie : relier les classes de la base à des concepts précis (en haut) ou les relier à des concepts plus généraux en utilisant des contraintes de nature dans les critères de sélection.*

Relier la procédure de représentation aux sous-concepts nous semble donc généralement préférable, même s'il y a probablement des exceptions, dans le cas où la liste des sous-concepts fournie par les spécifications est limitative. Le problème est plus complexe quand elle ne l'est pas. Prenons l'exemple de la classe [bâtiment] dans la BDTopo Pays. On trouve dans les spécifications une liste des types de bâtiments concernés par cette classe de la base ; cette liste peut servir pour créer dans l'ontologie des sous-types du type d'entité géographique BÂTIMENT, chacun de ces sous-types étant reliés à la même procédure de représentation, et les autres sous-types éventuels ne l'étant pas. Un des sous-types non représentés est CHÂTEAU D'EAU. Cependant, la liste des types représentés contient « constructions diverses ». La définition d'un tel concept est évidemment très floue ; mais on peut l'interpréter comme signifiant « constructions n'entrant dans aucune autre catégorie », ainsi les châteaux d'eau, entrant dans une autre catégorie, n'en feraient pas partie, ce qui correspond à la signification voulue dans les spécifications. Plus exactement, ce type de mention sert à indiquer que la liste n'est pas nécessairement exhaustive : il ne s'agit vraisemblablement pas de prétendre que *tout* ce qui pourrait être regroupé sous l'étiquette « constructions diverses » doit être considéré comme relevant de la classe [bâtiments] de la BDTopo Pays, mais simplement d'indiquer que des entités qui n'entrent dans aucune autre des sous-catégories de bâtiments mentionnées *peuvent* néanmoins être représentées par cette classe. Il ne nous semble cependant pas très souhaitable de définir dans l'ontologie un concept tel que « constructions diverses » ou « autres bâtiments » : la définition d'un tel concept serait modifiée par l'ajout d'un nouveau type de bâtiments, et compliquerait donc l'évolution de l'ontologie.

Dans un tel cas, une bonne solution peut être, si peu de sous-concepts sont exclus, de relier la procédure de représentation au concept général, ici BÂTIMENT, en excluant dans la contrainte de sélection les sous-concepts qui ne sont pas représentés. Dans notre exemple, cela s'écrirait `Non Est chateau_d_eau`. En l'occurrence, ce choix semble bon puisque la seule raison pour laquelle les CHÂTEAUX D'EAU sont exclus est qu'ils sont représentés par une autre classe[1] : on peut donc supposer qu'un éventuel raffinement a posteriori de la sous-classification des BÂTIMENTS n'introduira pas de nouveaux concepts non représentés, puisque si de tels concepts étaient représentés par ailleurs dans la base de données ils seraient déjà présents dans l'ontologie. En fait, la meilleure solution dans un tel cas serait probablement de ne pas exclure directement le sous-concept (ici CHÂTEAU D'EAU) mais le parent concurrent de ce sous-concept (ici RÉSERVOIR) : `Non Est reservoir`. Cela indique explicitement que si une entité géographique appartient à la fois aux types RÉSERVOIR et BÂTIMENT, la représentation en tant que RÉSERVOIR est privilégiée ; et cela permet l'ajout d'un type d'entité géographique qui serait à la fois un sous-type de BÂTIMENT et de RÉSERVOIR sans être un sous-type de CHÂTEAU D'EAU. Une autre possibilité, pour bien montrer qu'il s'agit d'une contrainte de mono-représentation, serait d'utiliser `Non Sélectionné comme reservoir`, faisant ainsi référence à la classe de la base [réservoir] et non plus au type d'entité géographique RÉSERVOIR.

L'inconvénient de cette solution est que la liste de sous-concepts présente dans les spécifications papier n'est pas reportée dans les spécifications formalisées. Plus exactement, les sous-concepts seront définis dans l'ontologie, mais ne seront pas directement reliés à la procédure de représentation. Or si un sous-concept est mentionné dans les spécifications papier, on est sûr qu'il est concerné par cette procédure (sauf erreur dans les spécifications), tandis que s'il n'est pas mentionné, il peut y avoir un doute. Il pourrait donc être intéressant par exemple d'établir des liens redondants, un avec le concept général et d'autres avec chacun des sous-concepts explicitement mentionnés ; une autre solution pourrait être d'écrire la liste sous forme d'annotation dans la procédure de représentation.

3.4.3 Exemple

Nous allons présenter dans cette partie plusieurs diagrammes indiquant, pour le thème *hydrographie*, une possibilité d'ontologie, les classes de la base concernées et les liens entre ontologie et schéma, sur le modèle de la figure 3.5 : les types d'entités géographiques en bleu, les classes de la base en rouge et les liens en pointillés verts.

Les diagrammes présentés visent à indiquer des choix de modélisation possibles, pour l'exemple, mais ne prétendent pas donner la meilleure solution possible.

Rappelons que les thèmes ne sont pas toujours parfaitement cloisonnés et qu'il est donc pratiquement nécessaire de traiter la totalité des spécifications même si l'on ne

1. En réalité, il n'est écrit nulle part dans les spécifications de la BDTopo Pays que les CHÂTEAUX D'EAU ne sont pas concernés par la classe [bâtiment]. Ce qui le laissait supposer était le fait que les CHÂTEAUX D'EAU soient représentés par des objets de classe [réservoir], car les représenter en plus par des objets de classe [bâtiment] aurait signifié qu'on avait deux objets superposés possédant une géométrie identique, ce qui semblait curieux. Nous avons vérifié sur un jeu de données qu'il n'y avait effectivement pas d'objet [bâtiment].

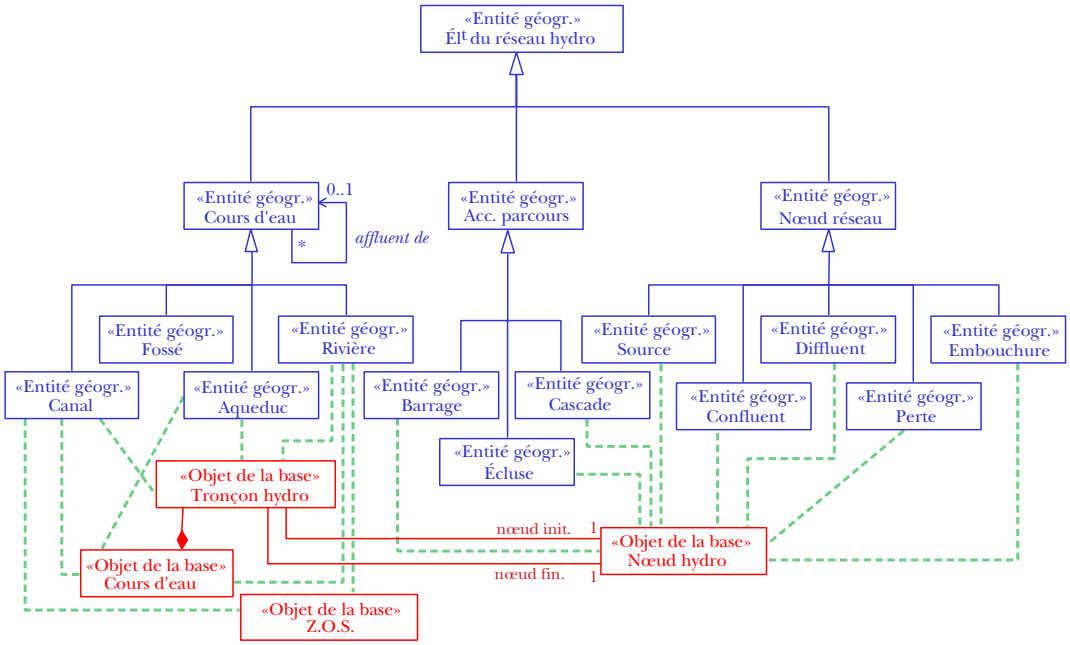


FIG. 3.8 *Représentation du réseau hydrographique dans la BDCarto.*

s'intéresse qu'à un domaine particulier, comme ici l'hydrographie. Ainsi, nous avons réalisé notre ontologie partielle en ne nous appuyant que sur les mots-clefs relevés dans les spécifications des thèmes *hydrographie* de chacune des deux bases (BDCarto et BDTopo Pays) ; mais, même avec seulement ces termes hydrographiques, de nombreuses classes d'autres thèmes interviennent dans la représentation. Ainsi la figure 3.9 ajoute à la figure 3.5 les classes d'autres thèmes de la BDTopo Pays qui sont en relation avec des éléments du réseau hydrographique : [construction linéaire] et [construction surfacique], du thème *bâti*, et [hydronyme], du thème *objets divers*. Dans la suite de l'exemple, nous avons indiqué les classes provenant d'autres thèmes mais intervenant dans la représentation de l'hydrographie. Cependant, lors de la formalisation complète des spécifications d'une base de données, il ne serait pas nécessaire de rechercher ainsi *a priori* les relations entre thèmes pour l'établissement des liens ontologie-schéma : il suffit que ces liens soient pris en compte dans l'ontologie, par exemple par une classification multiple. Les liens de [construction linéaire] et [construction surfacique] vers BARRAGE et ÉCLUSE seraient établis lors du traitement du thème *bâti*, et seraient visibles dans le thème *hydrographie* du fait de la classification de BARRAGE et ÉCLUSE dans les ÉLÉMENTS DU RÉSEAU HYDROGRAPHIQUE. Des outils logiciels, tels que ceux que nous présenterons au chapitre suivant, peuvent réaliser ceci automatiquement.

En général, tous les types d'entités géographiques apparaissant dans notre exemple d'ontologie sont mentionnés quelque part dans les spécifications, à l'exception de certaines catégories générales telles que ÉQUIPEMENT HYDROGRAPHIQUE OU PLAN D'EAU DOUCE que nous avons ajoutées.

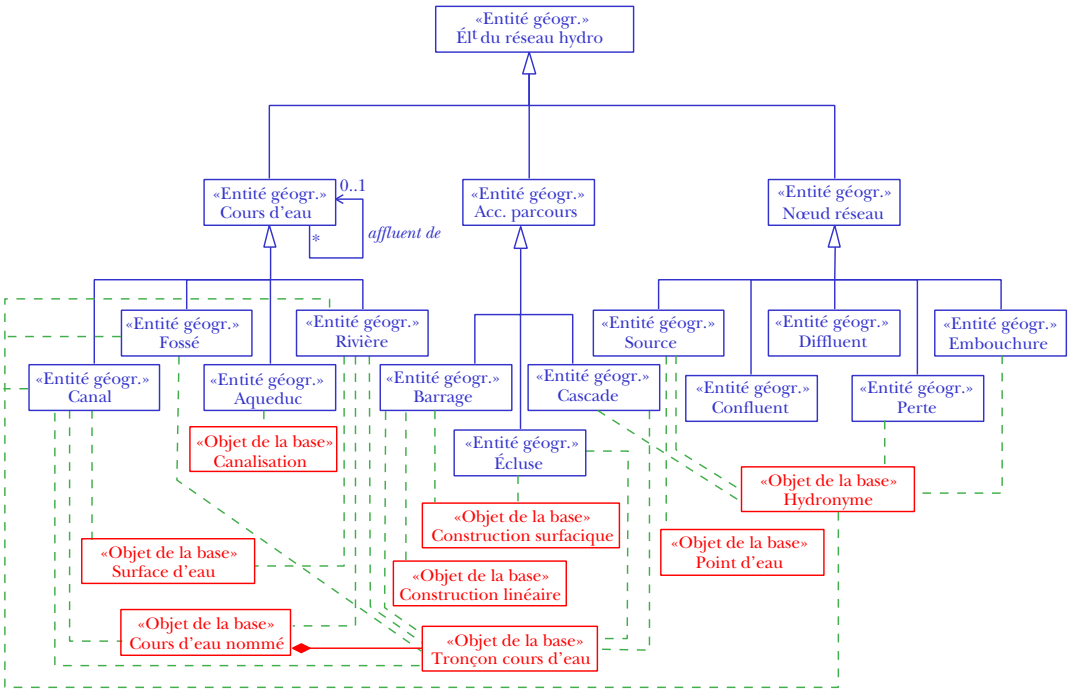


FIG. 3.9 *Représentation du réseau hydrographique dans la BDTopo Pays.*

Le réseau hydrographique

Pour chacune des deux bases, le réseau hydrographique est composé de différents types de COURS D'EAU, la BDTopo Pays ne mentionnant pas les aqueducs et la BDCarto ne mentionnant pas les fossés. Nous avons omis pour ne pas alourdir les diagrammes le type d'entité géographique RUISSEAU, qui n'est distingué de RIVIÈRE par aucune des deux bases. La BDCarto possède de plus une classe [nœud hydrographique] qui « correspond à une modification de l'écoulement de l'eau » ; tous les types d'entités géographiques que nous avons classés dans NŒUD DU RÉSEAU et ACCIDENT DE PARCOURS sont mentionnés dans les spécifications de cette classe, c'est pourquoi ils y sont tous reliés sur la figure 3.8. Les trois concepts classés dans ACCIDENT DE PARCOURS sont ceux qui sont qualifiés d'« obstacles » dans les spécifications de la BDTopo Pays.

Cette représentation des liens entre ontologie et schémas permet de mettre en évidence plusieurs choses qui l'étaient moins dans les spécifications papier. Tout d'abord, la multi-représentation des cours d'eau à l'intérieur d'une même base : dans la BDCarto (figure 3.8), chacun des trois types de cours d'eau représentés (CANAL, RIVIÈRE, AQUEDUC) l'est par trois classes différentes. La représentation en [tronçons hydrographiques] et celle en [cours d'eau] sont liées mais chacune possède ses propres spécifications ; la représentation par les [zones d'occupation du sol] (Z.O.S.) est pratiquement indépendante des deux autres[1] Dans la BDTopo Pays (figure 3.9), on trouve trois classes similaires, et la comparaison des deux diagrammes permet de

1. un attribut du [tronçon hydrographique] indique s'il est superposé à une [zone d'occupation du sol] de type 51, « eau libre », mais il n'y a pas d'autre lien.

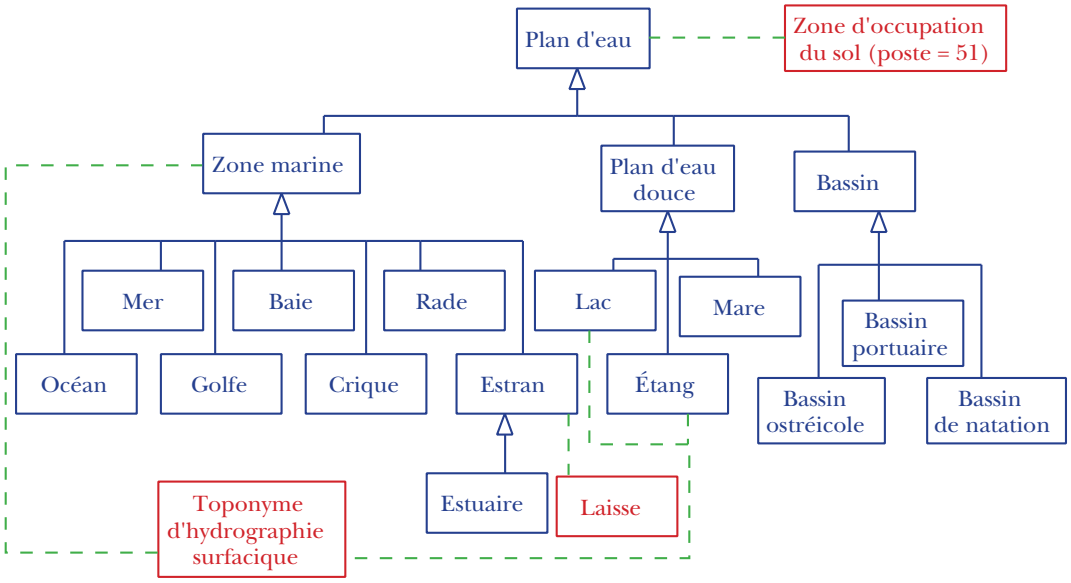


FIG. 3.10 *Représentation des plans d'eau dans la BDCarto.*

voir que la différence principale est la représentation des AQUEDUCS. L'absence de représentation des FOSSÉS dans la BDCarto n'est pas surprenante compte tenu du niveau de détail.

On peut ensuite remarquer que la BDTopo Pays ne possède aucune classe correspondant au [nœud hydrographique] de la BDCarto : les NŒUDS DU RÉSEAU n'y sont représentés que par des toponymes éventuels, sauf les SOURCES qui peuvent être représentées par des [points d'eau] ; mais ces [points d'eau] ne sont pas reliés à la représentation des COURS D'EAU. D'autre part, les ÉCLUSES et BARRAGES sont représentés à la fois par des [tronçons de cours d'eau][1] et par des classes appartenant au thème *bâti* ; une telle information n'est pas directement apparente dans les spécifications telles qu'elles existent actuellement mais est cruciale pour la vérification de la cohérence des données, par exemple.

Plans d'eau

Les PLANS D'EAU nous fournissent un exemple où il est intéressant de relier les classes de la base à des types généraux plutôt qu'à des types spécifiques, en raison de la forme des critères de sélection. Ainsi, figure 3.10, nous avons relié la [zone d'occupation du sol] de poste 51[2] au concept le plus général, car elle représente tout plan d'eau de plus de 4 hectares. Ce critère de sélection suffit à annuler toute ambiguïté sur la définition du concept ; il n'est par exemple pas utile, pour savoir exactement ce que représente la classe, de déterminer la limite entre une MARE et un ÉTANG. De la même façon, les spécifications du [toponyme d'hydrographie surfacique] men-

1. les barrages et écluses sont en fait distingués par des valeurs d'attributs particulières, ce qui ne se voit pas sur le diagramme puisqu'il ne montre pas les procédures de représentation.

2. Il est pratique de considérer la [zone d'occupation du sol] comme plusieurs classes différentes suivant la valeur de l'attribut [poste], en effet cette classe définit une partition de l'espace géographique en zones de différents types, distingués par cet attribut. 51 correspond à « eau libre ».

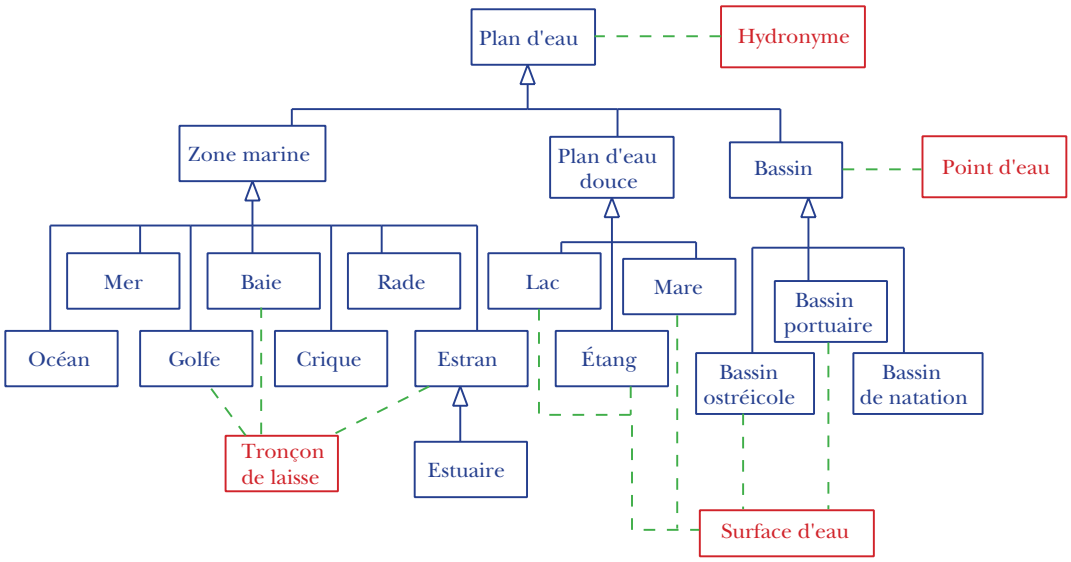


FIG. 3.11 *Représentation des plans d'eau dans la BDTopo Pays.*

tionnent une liste de types de zones marines[1] mais y ajoutent « espace marin divers », avec par ailleurs un seuil de sélection de 25 hectares. Le seuil semble ici aussi plus important que la classification précise pour savoir si une entité géographique est ou non concernée par cette classe. Pour les plans d'eau douce, nous l'avons reliée dans notre exemple à LAC et ÉTANG parce que ces termes sont explicitement mentionnés dans les spécifications. Il n'est d'ailleurs pas évident que cette classe puisse représenter tout PLAN D'EAU DOUCE dans la mesure où les BASSINS ne sont pas cités et où certaines entités peuvent être considérées à la fois comme des BASSINS et des PLANS D'EAU DOUCE.

En ce qui concerne la BDTopo Pays, figure 3.11, on a également une classe très générale, [hydronyme] : ses spécifications mentionnent notamment comme types d'entités géographiques concernés ÉTANG, LAC, MARE, ESPACE MARITIME et BASSIN, c'est-à-dire qu'elle concerne bien a priori tous les types d'entités géographiques présents sur la figure, le critère de sélection étant cette fois-ci que l'entité possède un toponyme présent sur la carte au 1:25 000 en service. On peut cependant douter du fait que les BASSINS DE NATATION soient concernés. Le terme de BASSIN DE NATATION provient des spécifications de la classe [surface d'eau], qui le mentionne uniquement pour l'exclure. Cette classe a été reliée, dans notre exemple, aux sous-concepts parce qu'une liste de sous-concepts est donnée ; cependant elle aurait également pu être reliée aux concepts généraux, avec un critère de sélection excluant les bassins de natation. Nous n'avons pas représenté ici tous les sous-types de BASSIN mentionnés par les spécifications.

Nous avons relié la classe [tronçon de laisse] à GOLFE et BAIE parce que ces termes sont explicitement indiqués, ce qui n'est pas le cas pour la classe [laisse] de la BD-Carto. La comparaison des deux diagrammes pourrait permettre d'homogénéiser

1. que nous n'avons pas toutes indiquées sur le diagramme : elle comprend également ANSE et CALANQUE.

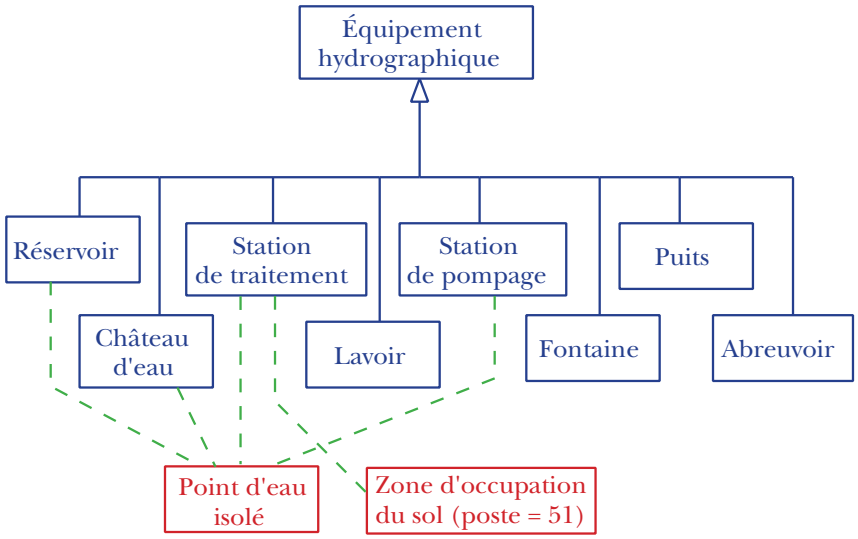


FIG. 3.12 *Représentation des équipements hydrographiques dans la BDCarto.*

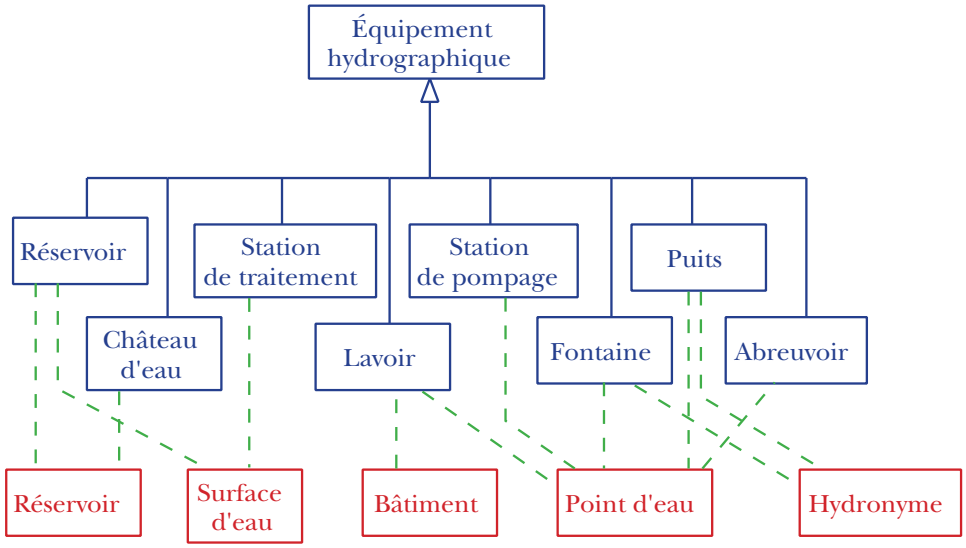


FIG. 3.13 *Représentation des équipements hydrographiques dans la BDTopo Pays.*

ceci : en effet, le contour terrestre d'une BAIE est bien représenté par une [laisse] dans la BDCarto, bien que les spécifications ne le mentionnent pas explicitement.

Équipements hydrographiques divers

Le type d'entité géographique ÉQUIPEMENT HYDROGRAPHIQUE n'est mentionné dans aucune des deux bases : nous l'avons introduit par commodité. CHÂTEAU D'EAU est considéré comme un sous-type de RÉSERVOIR dans la BDTopo Pays mais ce n'est pas le cas dans la BDCarto ; nous aurions pu mettre ce lien is-a dans l'ontologie.

Les figures 3.12 et 3.13 nous permettent notamment de remarquer que, malgré leurs noms similaires, les classes [point d'eau isolé] de la BDCarto et [point d'eau] de la BDTopo Pays n'ont en commun que la représentation des STATIONS DE POMPAGE. L'absence de représentation de LAVOIR, FONTAINE, PUITS et ABREUVOIR dans la BDCarto s'explique par le niveau de détail. Sur la figure 3.13, les liens entre LAVOIR et [point d'eau] et entre LAVOIR et [bâtiment] n'indiquent pas une multi-représentation mais des représentations alternatives, suivant que le lavoir est couvert ou à ciel ouvert ; ceci n'est pas apparent sur le diagramme mais le serait dans les procédures de représentation.

Groupes d'entités

Les figures 3.14 et 3.15 montrent des exemples de classification multiple, ainsi que des types d'entités géographiques représentant des groupes. Ces groupes ne sont mentionnés qu'à titre d'exemple : on pourrait, en l'occurrence, s'en passer. Cependant, définir un type d'entité géographique ENSEMBLE DE BRAS D'EAU en le reliant à [tronçon hydrographique] peut fournir une représentation alternative intéressante de la contrainte de sélection des parcours secondaires. Il s'agit en fait de considérer

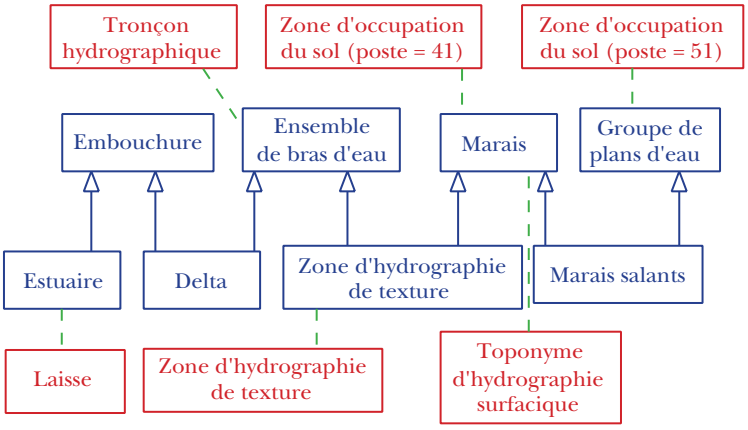


FIG. 3.14 *Représentation de certains groupes d'entités dans la BDCarto.*

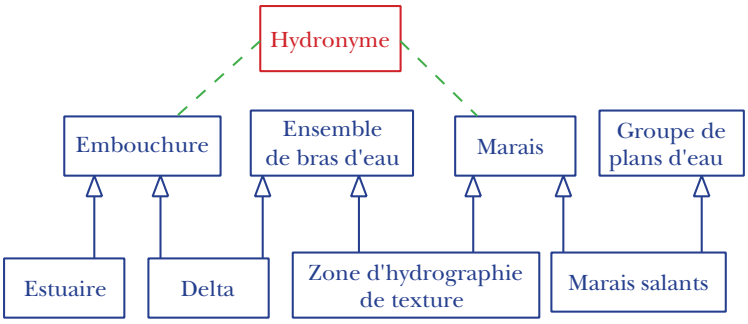


FIG. 3.15 *Représentation de certains groupes d'entités dans la BDTopo Pays.*

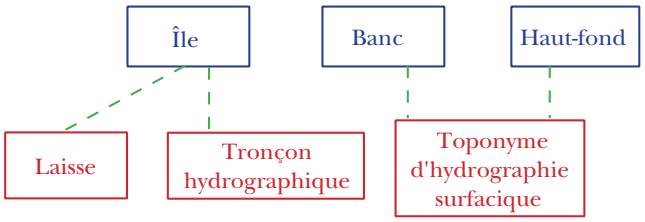


FIG. 3.16 *Représentation des îles, bancs et hauts-fonds dans la BDCarto.*

cette contrainte d'un point de vue différent : au lieu de dire qu'un seul des bras, choisi plus ou moins arbitrairement, est sélectionné, on dit que l'ensemble des bras est représenté par un seul tronçon. En l'occurrence, le tronçon saisi correspond en principe à l'axe d'un des bras, donc dire qu'il représente en fait un groupe de bras peut sembler artificiel. Mais il peut exister d'autres cas où, par exemple pour des raisons de généralisation cartographique, un groupe d'entités se trouve représenté par un objet, ou un groupe d'objets, sans qu'on puisse considérer chaque objet comme représentant précisément l'une des entités. Par exemple, un alignement de BÂTIMENTS peut être représenté en cartographie par un alignement d'objets [bâtiment], représentant la structure du groupe, sans que le nombre exact de bâtiments dans l'alignement soit respecté. Ce type de représentation est a priori moins courant dans des bases de données à but non directement cartographique, mais il peut néanmoins exister. Pour en revenir à notre exemple, dans certains cas, comme par exemple dans une zone de marais, les tronçons présents dans la base peuvent ne l'être qu'à titre indicatif, et les considérer comme représentant non directement des cours d'eau mais plutôt des ensembles de bras d'eau nous semble alors pertinent[1].

De façon similaire, la [zone d'occupation du sol] de poste 51 peut représenter des ensembles de plans d'eau : « ensemble de petits plans d'eau inférieurs à 4 hectares, la zone fermée comporte les 3/4 de surface en eau, la superficie de l'ensemble est supérieure à 4 hectares ». Ceci pourrait en fait s'exprimer à l'aide d'une contrainte d'agrégation, dans la procédure de représentation des plans d'eau individuels ; cependant, dans les spécifications actuelles, c'est bien exprimé comme une représentation de groupe, contrairement, par exemple, au critère de représentation des FORÊTS, qui est vraiment considéré comme une agrégation : « les zones distantes de moins de 100 mètres sont regroupées ». Ce dernier critère est d'ailleurs mentionné en tant que règle de généralisation, tandis que l'« ensemble de petits plans d'eau » est mentionné dans la *définition* des zones d'eau.

La ZONE D'HYDROGRAPHIE DE TEXTURE est définie dans les spécifications de la BDCarto comme « une zone plate au drainage complexe dans laquelle circule un ensemble de portions de cours d'eau formant un entrelacs de bras d'égale importance ».

1. Les spécifications de la BDTopo Pays indiquent que « la continuité du réseau hydrographique n'est [...] pas toujours assurée dans [...] les zones de marais, où les connexions et interruptions du réseau restent indicatives. » À l'inverse, les spécifications de la BDCarto indiquent que « la continuité du réseau est assurée lors de la traversée [...] de zones de marais ». Cette différence de spécifications est difficile à exprimer dans notre modèle ; l'utilisation de groupes de bras d'eau pourrait être une piste intéressante.

Îles

Les spécifications de la BDCarto ne mentionnent les îles que dans les critères de sélection des [laisses] et des [tronçons hydrographiques], et non en tant qu'entités représentées dans la base de données. Cependant il s'agit bien dans les deux cas d'indiquer une superficie minimale pour la représentation des îles ; nous pensons donc logique de relier ces classes au type d'entité géographique île. La BDTopo Pays ne mentionne île que dans les spécifications du [tronçon de laisse], et cette fois en tant qu'entité représentée.

Propriétés

Nous n'avons pas, dans notre exemple, établi la liste des propriétés de chaque type d'entité géographique. Ces propriétés ne sont pas nécessaires, du moins dans l'état actuel de notre modèle, pour l'établissement des liens entre ontologie et schémas ; elles servent pour l'expression des contraintes de sélection et des définitions d'attributs. En conséquence, nous avons défini les propriétés nécessaires seulement lors de la rédaction des procédures de représentation, dont nous présenterons des exemples au chapitre suivant. Il était cependant tout à fait envisageable de définir les propriétés en même temps que les types d'entités géographiques, dans la mesure où les noms des propriétés peuvent, comme les noms des types d'entités géographiques, être fournis par des mots-clefs relevés dans les spécifications.

Si l'on cherche à définir une ontologie munie d'une structure plus complexe, avec par exemple des définitions formelles ou tout au moins des connaissances sur les concepts, définir les propriétés en même temps que les concepts peut même être nécessaire. Ainsi, dans la BDTopo Pays, un bassin est défini comme une « *construction* non couverte destinée à recevoir de l'eau [...] », et ce terme est opposé par les spécifications à « surface hydrographique naturelle ». On pourrait donc dire qu'un bassin est un plan d'eau dont la propriété artificiel est vraie. De la même façon, un canal peut peut-être être défini comme un cours d'eau dont la propriété artificiel est vraie. De telles connaissances pourraient être utilisées pour raisonner sur les spécifications formalisées : leur utilisation pourrait être une extension intéressante de notre modèle.

3.4.4 Procédures de représentation

Une fois les liens entre schéma et ontologie déterminés, on peut passer à la définition des procédures de représentation. Il faut pour cela pour chaque type d'entité — ou ensemble de types d'entités représentés par la même procédure — prendre les spécifications de l'ensemble des classes qui le concernent. Les critères de sélection de l'entité sont normalement la disjonction (« ou » logique) des critères donnés pour ce type d'entité dans chacune des classes : une entité de ce type est prise en compte si et seulement si elle est prise en compte par au moins une des classes concernées. Si une classe donne des critères plus stricts, ces critères seront transformés en conditions d'instanciation par cette classe, dans la section *instanciation* de la procédure. Les règles de découpage peuvent concerner plusieurs classes différentes ; généralement, le découpage pour changement d'attribut est décrit dans la spécification du tronçon, car dans ce cas, le nœud éventuel n'a aucun rôle sémantique : il s'agit d'un artifice d'implémentation. En revanche, le découpage pour la rencontre d'un obstacle est

plutôt décrit dans la spécification du nœud, puisque dans ce cas, le nœud représente l'obstacle. Les règles d'agrégation que nous avons pu voir ne concernaient quant à elles toujours qu'une seule classe. Les règles d'instanciation sont bien sûr toujours entièrement décrites dans les spécifications de la classe concernée.

Avant la rédaction de ces procédures de représentation, il est utile de pouvoir saisir et stocker dans une base de données l'ontologie, les schémas et les liens entre eux, ainsi que de pouvoir explorer ces schémas et ces liens. Cela permet entre autres de mettre en évidence des liens indirects qui n'étaient pas visibles à la lecture des spécifications : ainsi un lien peut avoir été établi vers un concept lors du traitement d'une certaine hiérarchie et être rendu automatiquement accessible via une autre hiérarchie. Nous avons donc développé un ensemble d'outils logiciels à cet effet, que nous présenterons dans le chapitre suivant. Nous discuterons la rédaction des procédures de représentation plus avant à la fin dudit chapitre, sur l'exemple des COURS D'EAU.

Chapitre 4

Mise en œuvre pratique

Où l'on présentera, en 4.1, des outils logiciels réalisés pour étudier la mise en œuvre pratique de notre modèle. Ces outils s'intègrent dans la plate-forme de développement du laboratoire COGIT, qui utilise le langage Java. Ils comprennent notamment une interface de manipulation des divers schémas qui entrent en jeu et un parseur pour le langage de description des procédures de représentation décrit au chapitre précédent.

Ce parseur permet de transformer la procédure de représentation exprimée dans ce langage en une structure d'objets (4.1.4). Ceci permet d'une part de la rendre plus facilement manipulable par logiciel et, d'autre part, de transformer en relations explicites les références faites, dans la procédure de représentation, à des éléments des schémas ou de l'ontologie en utilisant leurs identifiants. Cela permet également, au passage, de vérifier la syntaxe de la procédure.

On montrera ensuite, en 4.2, deux exemples de rédaction de procédures de représentation dans le langage défini au chapitre précédent, correspondant à la représentation des mêmes types d'entités, les COURS D'EAU, *dans deux bases de données différentes. On verra ainsi une partie des possibilités et des limites de ce langage pour l'expression des spécifications.*

Afin d'étudier la mise en œuvre pratique du modèle de représentation des spécifications défini au chapitre précédent, nous avons développé un ensemble d'outils logiciels en java permettant :

— de saisir et de manipuler les éléments du schéma et de l'ontologie, et également de les stocker ;

— de saisir et de stocker des procédures de représentation associées à un ensemble de types d'entités géographiques ;

— d'interpréter ces procédures de représentation pour créer automatiquement une structure d'objets représentant les spécifications ;

— de naviguer dans cette structure.

Nous décrirons dans un premier temps ces outils logiciels, puis, dans une deuxième partie, nous étudierons l'utilisation de notre modèle sur deux exemples concrets, la représentation des cours d'eau dans la BDCarto et dans la BDTopo Pays.

4.1 Description des outils logiciels

4.1.1 Structure générale

Nous avons utilisé pour développer nos outils la plate-forme GeOxygene du laboratoire COGIT[1]. Cette plate-forme est essentiellement conçue pour la manipulation de données géographiques, alors que nous souhaitons manipuler des *méta*données ; nous n'utilisons donc que peu de ses possibilités. L'intérêt d'intégrer notre travail à la plate-forme du laboratoire est double : cela permet de faciliter son utilisation par les autres membres du laboratoire, et d'autre part de stocker les données et les métadonnées ensemble, ce qui facilitera leur utilisation simultanée. En effet, les spécifications peuvent être enrichies par des connaissances issues de l'analyse des données elles-mêmes, et, d'autre part, ces spécifications peuvent être utilisées pour étudier la cohérence des données [Sheeren, 2005].

La partie de la plate-forme GeOxygene que nos outils utilisent est la gestion de la *persistance* des objets java dans une base de données relationnelle, gestion qui s'appuie sur la bibliothèque OJB[2]. Le stockage des schémas, de l'ontologie et des procédures de représentation est réalisé par ce moyen.

Nous avons dans un premier temps défini un certain nombre de classes génériques, regroupées dans un package java, outilsSchema, permettant de gérer l'affichage et l'édition, dans des boîtes de dialogue, de divers éléments de schéma, ainsi que leur stockage dans la plate-forme GeOxygene. Les éléments de schéma gérés par ce package sont des objets java de classe ObjetSchema. Ces objets peuvent posséder un certain nombre d'attributs, dont les valeurs seront visibles et éditables dans l'interface, et également participer à des relations, qui seront visibles et navigables dans l'interface. Nous décrirons brièvement le fonctionnement de ce package en 4.1.2.

Le reste de nos outils logiciels s'appuie sur ce package. Comme nous l'avons vu au chapitre précédent, notre modèle consiste à représenter d'une part une ontologie et d'autre part des schémas de données, puis à les relier par des procédures de

1. `http://oxygene-project.sourceforge.net/`
2. `http://db.apache.org/ojb/`

représentation. Nous pensons qu'il serait utile de pouvoir définir les schémas des bases et les ontologies à l'aide d'outils indépendants de notre prototype, car il existe déjà des outils dédiés à cela. Par ailleurs, les schémas des bases peuvent déjà exister dans un format informatique quelconque avant la formalisation des spécifications[1]. Nous ne nous sommes donc pas concentré sur la représentation de ces deux parties : nous avons simplement défini des interfaces java pour les décrire, interfaces dont nous avons réalisé, pour les besoins de nos exemples, des implémentations limitées au minimum. Nous détaillerons ceci en 4.1.3.

Pour la représentation des procédures de représentation, nous avons défini une structure de classes java qui n'est autre qu'une traduction, dans un modèle orienté-objet, de la structure décrite au chapitre précédent à l'aide d'une grammaire formelle. Nous décrirons cette structure en 4.1.4.

Enfin, nous expliquerons en 4.1.5 comment cette structure est générée à partir d'une procédure de représentation exprimée dans notre langage.

4.1.2 Le package outilsSchema

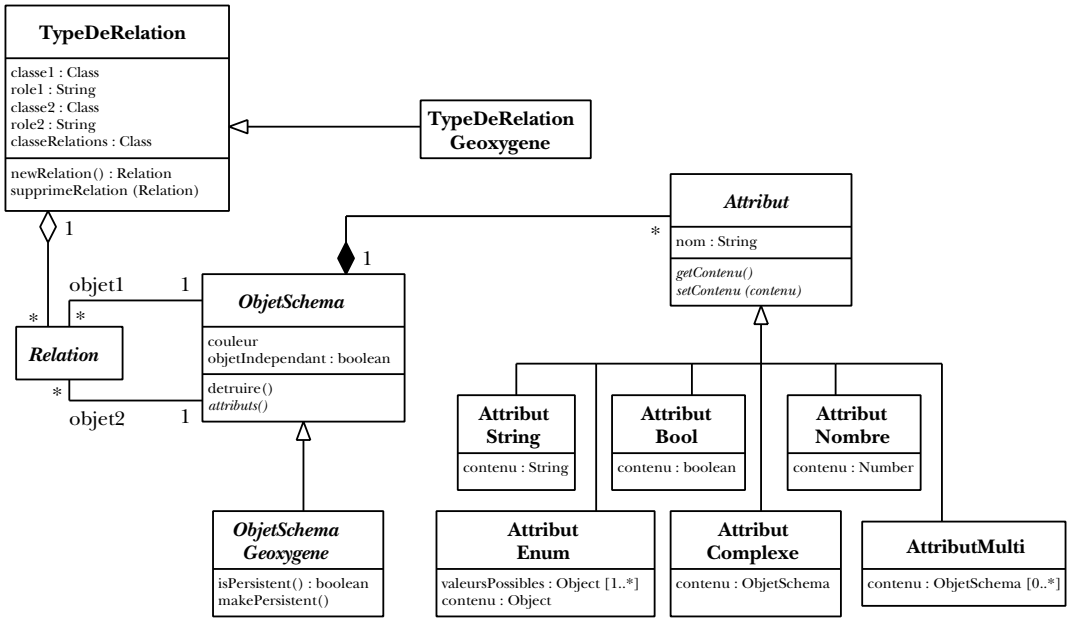


FIG. 4.1 *Diagramme de classes UML montrant la structure générale du package outilsSchema.*

Pour faciliter la compréhension de la suite, nous allons décrire brièvement les principaux éléments du package outilsSchema, sur lequel s'appuie le reste de nos outils logiciels.

Pour pouvoir être exploré et modifié par notre interface, un schéma doit être composé d'objets de classe ObjetSchema ; leurs attributs visibles dans l'interface doivent appartenir à la classe Attribut.

1. Notons toutefois qu'il n'est pas absolument évident que le schéma *conceptuel*, celui sur lequel s'appuient les spécifications, soit reconstituable à partir des données ou soit disponible facilement sous forme autre que textuelle ou graphique.

La classe ObjetSchema

Pour que les instances d'une classe soient éditables à l'aide de notre interface, cette classe doit être une sous-classe d'ObjetSchema. Elle doit également définir une méthode attributs() renvoyant la liste des attributs que l'interface doit montrer, attributs qui doivent appartenir à la classe Attribut.

Afin de permettre l'utilisation de fenêtres de différentes couleurs pour des objets appartenant à différentes parties du schéma, par exemple des objets correspondant aux schémas de deux bases de données différentes, la classe ObjetSchema possède un attribut couleur qui indique avec quelle couleur l'objet doit être affiché. Elle possède également un attribut objetIndependant qui indique si l'objet est indépendant ou s'il n'existe qu'en tant qu'attribut d'un autre objet. Ce dernier attribut permet d'interdire à l'interface certaines opérations, comme la destruction de l'objet ou sa sauvegarde dans la base de données : ces opérations ne sont permises qu'à l'objet propriétaire.

La classe abstraite ObjetSchema possède une sous-classe prédéfinie, ObjetSchemaGeoxygene, qui ajoute à ObjetSchema la gestion de la persistance des objets par le moteur de GeOxygene. Il serait possible, si on voulait utiliser un autre système que GeOxygene, de définir une sous-classe équivalente gérant la persistance d'une autre façon.

La classe Attribut

La classe abstraite Attribut regroupe tous les types d'attributs que peut posséder un ObjetSchema. Ces attributs possèdent tous un nom et une valeur ; leur valeur est accessible par la méthode getContenu() et modifiable par la méthode setContenu(). Leur nom est fixe. Différentes classes d'attributs sont définies suivant le type de la valeur de l'attribut :

— les classes AttributString, AttributBool et AttributNombre, qui peuvent contenir ce que leur nom indique ;

— la classe AttributEnum, qui possède une liste de valeurs autorisées (de type Object) et peut contenir uniquement une valeur de la liste ;

— la classe AttributComplexe, dont la valeur est un objet de classe ObjetSchema[1] ;

— la classe AttributMulti, qui peut contenir plusieurs valeurs d'une sous-classe donnée de ObjetSchema.

Les classes Relation et TypeDeRelation

La classe TypeDeRelation permet de définir un type de relation binaire entre deux sous-classes d'ObjetSchema. Un objet de cette classe comprend les différents éléments décrivant le type de relation : les deux classes concernées et les noms des deux rôles. D'autre part, cet objet de classe TypeDeRelation est associé à un ensemble d'objets de classe Relation, qui correspondent aux instances de la relation. Il possède des méthodes permettant de créer et de supprimer de telles instances[2].

1. Cette classe a la particularité que la valeur de l'attribut est invariable, ou du moins ne peut pas être changée par l'interface. L'objet qui constitue la valeur de l'attribut peut être modifié, mais il s'agit toujours du même objet.

2. Pour faciliter la gestion de la persistance, chaque objet de classe TypeDeRelation est associé à une certaine sous-classe de Relation, et l'ensemble des relations qui correspondent au type est l'ensemble des instances de cette sous-classe.

TypeDeRelation possède une sous-classe, TypeDeRelationGeoxygene, qui lui ajoute la gestion de la persistance des relations via la plate-forme GeOxygene.

L'interface graphique

Enfin, notre package outilsSchema permet d'afficher une boîte de dialogue correspondant à un objet donné de classe ObjetSchema. Cette boîte de dialogue permettra de visualiser et d'éditer les différents attributs de l'objet ainsi que les relations auxquelles il participe. Plus précisément, pour chacun des attributs sont affichés son nom et un élément d'interface graphique permettant d'éditer sa valeur, élément qui dépend de la classe de l'attribut : il s'agira d'une zone de texte pour un AttributString, d'une case à cocher pour un AttributBool, etc. Pour chaque type de relation concernant l'objet, une liste des objets en relation est affichée, et il est possible de créer de nouvelles instances de la relation par glisser-déposer entre les fenêtres correspondant aux deux objets à relier. Cette partie graphique utilise la bibliothèque standard java Swing.

Utilisation

Le reste de nos outils logiciels utilise ce package, pour permettre la navigation dans les schémas via l'interface graphique et pour gérer la persistance des données. Cependant il s'agit là de détails d'implémentation que nous omettrons pour faciliter la lecture. Indiquons simplement que dans les paragraphes suivants :

— tous les éléments que nous représentons comme des classes sont implémentés comme des sous-classes d'ObjetSchema ;

— à quelques exceptions près, leurs attributs appartiennent à la classe Attribut, et sont déclarés par leurs méthodes attributs() ;

— la plupart des éléments que nous représentons comme des associations sont implémentés comme des sous-classes de Relation associées à des objets de classe TypeDeRelation. Toutefois, les relations de composition correspondent à des AttributMulti et certaines associations 1-1 ou 1-0..1 sont implémentées par des AttributComplexe (voir les différentes sous-classes d'Attribut figure 4.1).

À titre d'exemple, nous avons indiqué en gris l'utilisation du package outilsSchema sur la figure 4.2.

4.1.3 Le package modeleSpecs : structure générale

Comme nous l'avons dit au chapitre précédent, notre modèle de représentation des spécifications comprend trois parties : une ontologie, des schémas de bases de données, et des procédures de représentation. À cette structure correspondent dans notre prototype trois classes, TypeEntiteGeographique pour l'ontologie, ClasseBase pour les schémas des bases de données et Modelisation pour les procédures de représentation (figure 4.2). Notre modèle n'utilise que le minimum d'informations sur les parties ontologie et schémas, de façon à laisser ouvert le choix de leur implémentation et de leur structure. Nous avons défini des sous-classes très simples de ObjetSchema et TypeEntiteGeographique pour les besoins de nos exemples, mais il serait possible d'utiliser des structures plus complexes.

Nous allons maintenant détailler ces trois parties.

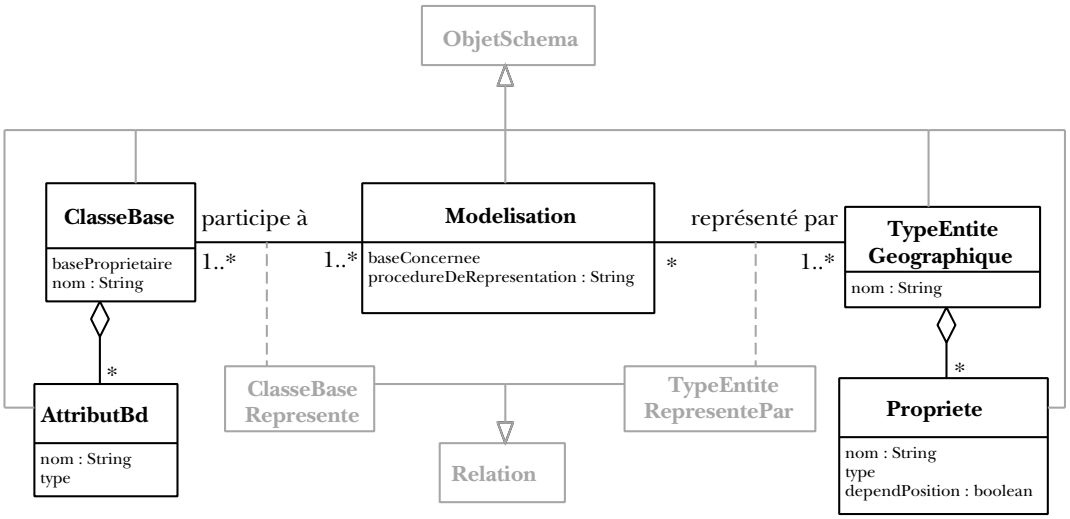


FIG. 4.2 *Diagramme UML indiquant la structure générale du package modeleSpecs. En gris, le détail des liens avec le package outilsSchema.*

Les classes ClasseBase et AttributBd

Pour les besoins du modèle de représentation des spécifications, les seules informations qu'on a besoin de connaître sur les classes de la base sont leur nom et la liste de leurs attributs. Pour pouvoir gérer simultanément les schémas de plusieurs bases de données, nous avons également besoin d'un attribut indiquant à quelle base la classe appartient. Nous n'imposons donc à la classe ClasseBase que deux attributs et une relation, en plus de la relation « participe à » avec la classe Modelisation.

Les attributs des classes, de classe AttributBd, doivent quant à eux posséder deux attributs, nom et type. L'attribut type n'est pas actuellement utilisé par notre prototype, mais il pourrait l'être afin de vérifier la validité des affectations d'attributs, dans les règles d'instanciation (voir plus loin, figure 4.3).

Pour permettre le stockage dans la plate-forme GeOxygene des schémas que nous voulions traiter, nous avons défini des classes ClasseBaseSimple et AttributBdSimple qui héritent de ObjetSchemaGeoxygene (voir 4.1.2)[1].

Ces classes très simples suffisent aux besoins de notre modèle. Cependant, il pourrait être utile de stocker davantage d'informations sur ce schéma ou d'importer un schéma déjà existant; il suffit pour cela de définir une autre sous-classe de ClasseBase. À titre d'exemple, nous avons ainsi défini une classe ClasseBase_GFM permettant l'importation d'un schéma écrit dans le *General Feature Model* (GFM) défini par la norme ISO 19109.

Nous avons mentionné dans le chapitre précédent que la structure précise des bases de données, leur *schéma logique*, dépend du format de livraison des bases de données. Un tel schéma est associé à un certain *jeu de données*. Les spécifications auxquelles nous nous intéressons sont, au contraire, des documents génériques qui s'appuient sur le schéma *conceptuel* de la base de données et concernent l'ensemble

1. En java, les classes ClasseBase et AttributBd sont définies comme des interfaces.

de la base, indépendamment du format de livraison. « La » base de données peut même éventuellement regrouper en fait plusieurs bases de données physiquement distinctes en raison d'un découpage géographique, comme dans le cas de la BDTopo de l'IGN. [Balley, 2005] propose d'appeler *produit* l'ensemble des données produites conformément à un jeu de spécifications donné, indépendamment du format; le produit est donc une entité abstraite puisque les données sont toujours stockées dans un certain format. Un *jeu de données* est alors une réalisation du produit dans un format donné, qui peut être totale ou partielle (seulement certaines zones ou certains thèmes). Par exemple, « la BDTopo Pays version 1.2 » désigne un produit; « les thèmes routier et hydrographique de la BDTopo Pays de 2003 sur le département du Loiret au format Shapefile » désigne un jeu de données.

La plate-forme GeOxygene intègre le General Feature Model sous forme d'interfaces java. Les éléments du catalogue de données (*feature catalogue*, voir 3.1.1), qui est la description du schéma conceptuel d'un *produit*, sont définis par des classes implémentant ces interfaces, ce qui est conforme aux recommandations de l'ISO 19110. Il sera à terme possible de gérer différents schémas dérivés correspondant à des jeux de données : schémas adaptés aux jeux de données tels qu'ils sont vendus, schémas modifiés par l'utilisateur... Les éléments de ces différents schémas appartiendront à différentes implémentations des interfaces correspondant au GFM, et conserveront par ailleurs un lien vers l'élément du catalogue de données dont ils sont issus.

Le GFM est par ailleurs utilisé pour représenter les schémas de données dans le format d'échange GML 3 (Geography Markup Language, ISO 19136).

L'importation des classes de la base à partir de l'interface GF_FeatureType du GFM permettrait donc d'utiliser tous ces schémas avec notre modèle. La formalisation des spécifications pourrait être faite en s'appuyant sur le schéma du catalogue de données et les spécifications, une fois formalisées, être reliées aux schémas plus spécifiques grâce aux liens définis entre les schémas.

La classe ClasseBase_GFM que nous avons définie sert en fait d'intermédiaire entre le schéma GFM et notre modèle : un objet de cette classe est créé à partir d'un objet de classe GF_FeatureType, dont il reprend les valeurs d'attributs en les plaçant dans des objets de classe Attribut, ce qui les rend éditables dans notre interface. Des liens peuvent ensuite être établis avec cet objet ClasseBase_GFM lors de la formalisation des spécifications (voir 4.1.4). Il conserve un lien vers l'objet d'origine (de classe GF_FeatureType), il est donc possible par la suite de transformer les relations établies avec cet objet intermédiaire en liens avec le schéma GFM. D'autre part, on pourrait, si on le souhaite, permettre l'édition des attributs de l'objet dans l'interface en écrivant une méthode permettant de reporter les modifications dans l'objet d'origine; ceci n'est cependant pas forcément souhaitable. Comme il s'agit d'un objet intermédiaire, il n'est pas nécessaire de le stocker en tant que tel dans la base de données; nous n'avons donc pas fait hériter la classe ClasseBase_GFM de ObjetSchemaGeoxygene.

Nos exemples ont néanmoins utilisé la classe ClasseBaseSimple, qui nous permettait de saisir nos propres schémas (ou extraits de schémas).

Les classes TypeEntiteGeographique et Propriete

De la même façon que pour les classes de la base, notre modèle n'a besoin que de deux informations sur un type d'entité géographique : son nom et l'ensemble de ses propriétés. La classe TypeEntiteGeographique ne comprend donc rien de plus.

La classe Propriete, quant à elle, comprend les attributs nom, type et également dependPosition. Ce dernier attribut est un booléen indiquant si la valeur de la propriété peut varier en fonction de la position à l'intérieur de l'entité ou si, au contraire, la propriété possède une valeur unique pour l'ensemble de l'entité. Notre modèle n'utilise actuellement que l'attribut nom, mais nous pensons que les deux autres attributs sont utiles et pourraient être utilisés à des fins de vérification. Ainsi le type des propriétés pourrait restreindre les possibilités autorisées dans les contraintes sur propriétés, et permettre de vérifier la cohérence des affectations de valeurs aux attributs. La variabilité selon la position permettrait de vérifier deux choses. Tout d'abord, seules des propriétés variables selon la position devraient être utilisées dans les règles de découpage pour changement d'attribut. Ensuite, l'affectation d'une valeur à un attribut à partir d'une propriété variable devrait obligatoirement être accompagnée d'une précision indiquant comment la valeur est choisie (par exemple valeur maximale ou valeur moyenne), c'est-à-dire comment la valeur variable est transformée en valeur unique.

Nous avons utilisé pour nos exemples une sous-classe de TypeEntiteGeographique, TypeEntiteGeographiqueSimple, qui hérite également de ObjetSchemaGeoxygene, de façon similaire à ce que nous avons fait pour les classes de la base. Nous avons de plus défini un type de relation, HeritageEntites, pour permettre l'établissement de liens is-a entre types d'entités géographiques simples[1]. Nous avons enfin défini une méthode toutesProprietes() qui permet de considérer automatiquement que les propriétés des types parents appartiennent également aux types enfants.

De la même façon que ClasseBase peut être étendue pour importer des schémas de bases de données préexistants, il serait possible de définir une sous-classe de TypeEntiteGeographique permettant l'importation d'une ontologie externe, par exemple au format OWL. Cependant nous n'avons pour l'instant pas beaucoup étudié cette possibilité dans la mesure où, pour nos exemples, nous avons choisi de définir une ontologie *ad hoc*, à partir des spécifications.

La classe Modelisation

La classe Modelisation est au centre de notre modèle : c'est elle qui relie entités géographiques et classes de la base et qui contient la procédure de représentation. Cette classe possède comme attributs uniquement la procédure de représentation, qui est une chaîne de caractères, et un numéro identifiant la base de données à laquelle cette procédure se rapporte.

Deux types de relations sont définis pour relier cette classe aux deux schémas (base de données et ontologie) : TypeEntiteRepresentePar et ClasseBaseRepresente (voir figure 4.2). Les instances de ces relations peuvent être saisies interactivement (par glisser-déposer, voir 4.1.2).

Le troisième type de relation concernant la classe Modelisation la relie à la classe BlocModelisation, qui est un des éléments de la représentation structurée de la procédure de représentation générée par l'interprétation du langage correspondant, que nous allons maintenant décrire.

1. Ce type de relation n'apparaît pas sur la figure 4.2, car il n'appartient pas à la structure de notre modèle proprement dit mais à l'implémentation de la partie ontologie que nous avons utilisée pour nos exemples.

4.1.4 Représentation orientée-objet des procédures de représentation

Nous avons défini au chapitre précédent un langage formel permettant de décrire les procédures de représentation reliant les types d'entités géographiques aux classes de la base. Afin de permettre l'exploration et l'utilisation des spécifications ainsi formalisées par un logiciel, nous avons défini une structure de données orientée-objet reprenant les divers éléments de ces procédures de représentation. Le passage de la description textuelle de la procédure, dans le langage prévu à cet effet, à sa représentation structurée en objets est effectué grâce à un parseur (voir 4.1.5). Il ne s'agit que d'une manière différente de représenter la même information ; cependant, la structure d'objets présente plusieurs avantages, en plus d'être plus directement utilisable par un logiciel. En effet, nous avons vu que les spécifications décrivent comment on passe du terrain réel à la base de données (ou plus précisément du terrain conceptualisé au terrain nominal, voir figure 3.4). Mais l'utilisation des bases de données consiste essentiellement, à l'inverse, à partir des données pour en déduire des informations sur le monde réel. Il est donc utile voire indispensable de pouvoir remonter les procédures de représentation depuis les objets de la base vers les entités géographiques. Une possibilité serait de simplement regarder à quelles procédures de représentation la classe de la base à laquelle on s'intéresse est liée, puis de parcourir ces procédures pour retrouver l'information ; cela correspond d'ailleurs à la situation actuelle avec les spécifications non formalisées. Mais créer une structure d'objets pour représenter les procédures de représentation permettra de relier directement les classes de la base aux règles d'instanciation qui les concernent. On pourra alors remonter de ces règles jusqu'aux types d'entités géographiques sans devoir lire la totalité de la procédure, qui peut par ailleurs contenir les spécifications de plusieurs autres représentations, sans rapport avec la classe qui nous intéresse, des mêmes types d'entités. Plus généralement, il est possible de créer des relations bidirectionnelles avec tous les éléments du schéma et de l'ontologie qui interviennent dans la procédure de représentation, par exemple les types d'entités géographiques qui interviennent dans des contraintes de relation ; il peut être intéressant d'avoir directement accès à l'information que AGGLOMÉRATION, par exemple, sans être représenté en tant que tel, influe sur la représentation de tel et tel type d'entité : cela montre le rôle structurant de ce type d'entité.

Pour résumer, la structure que nous allons décrire ici permet de représenter la même information que le langage proposé au chapitre 3 mais pour en faire une utilisation différente. Notre prototype permet simplement la navigation dans la structure via l'interface graphique, mais un logiciel voulant raisonner sur les spécifications formalisées pourra la parcourir pour retrouver les informations qui l'intéressent.

Le point de départ de la représentation structurée des spécifications est la classe Modelisation. Il y a exactement un objet de cette classe par procédure de représentation ; il est relié aux types d'entités géographiques concernés par cette procédure par la relation « représenté par », et aux classes de la base concernées par la relaion « participe à ». C'est la seule classe persistante de la structure, et elle contient la représentation textuelle, dans le langage décrit au chapitre 3, de la procédure de représentation. Le reste de la structure peut être régénéré à la demande à partir de ce texte.

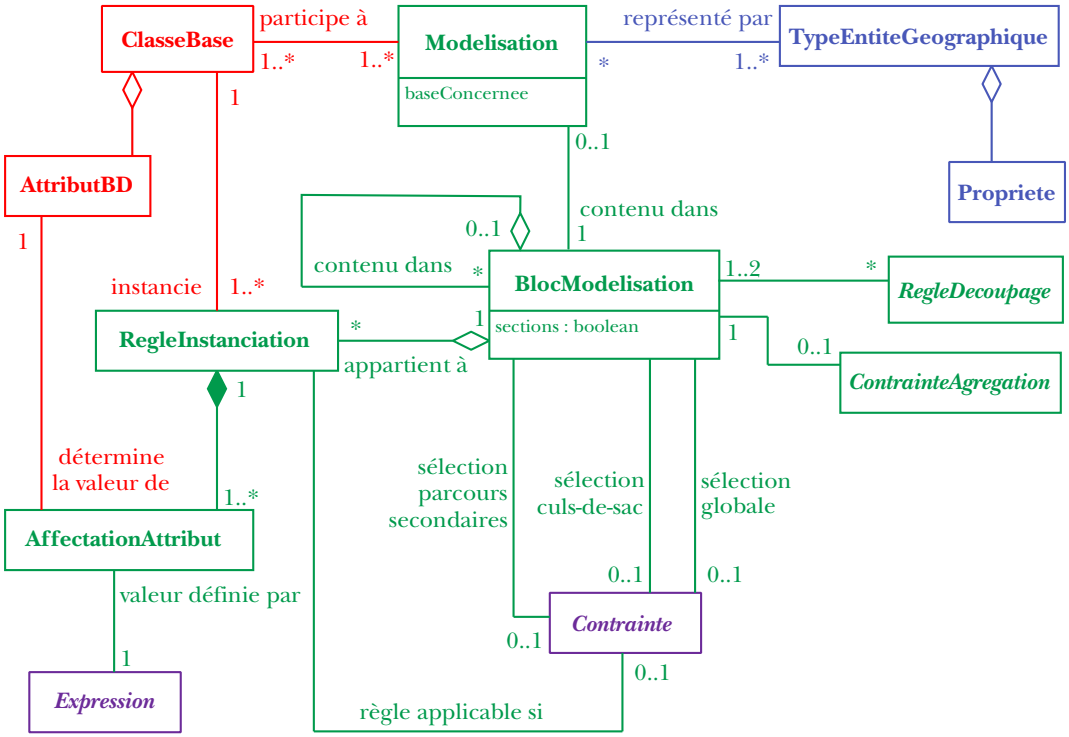


FIG. 4.3 *Diagramme de classes UML de la structure représentant une procédure de représentation. La partie bleue correspond aux éléments provenant de l'ontologie et aux liens avec eux, et la partie rouge à ceux provenant du schéma de la base. Les classes violettes sont décrites plus avant figure 4.4.*

La figure 4.3 représente les éléments constitutifs de cette structure. Pour plus de lisibilité, nous avons indiqué en bleu les éléments provenant de l'ontologie et en rouge ceux provenant du schéma de la base. Les éléments représentés en vert correspondent aux différentes parties de la procédure de représentation. Ceux représentés en violet sont les éléments non liés à une partie précise de cette procédure mais pouvant intervenir à différents endroits : contraintes et expressions.

Blocs modélisation

La procédure de représentation est structurée en blocs, de classe BlocModelisation. Un objet BlocModelisation représente un ensemble de règles de représentation s'appliquant à un même ensemble d'éléments. Ces éléments peuvent être soit des entités géographiques individuelles, soit des agrégats, soit des sections résultant d'un découpage, soit des limites de sections résultant également d'un découpage. Un bloc modélisation est toujours contenu soit dans un autre bloc modélisation soit directement dans l'objet Modelisation correspondant à la procédure de représentation. Il peut lui-même contenir un nombre quelconque de sous-blocs. Nous avons appelé cette relation « contenu dans » par similitude avec la structure du langage décrit au chapitre 3 (un bloc peut être défini à l'intérieur d'un autre bloc). La sémantique de cette relation « contenu dans » est que les éléments concernés par le sous-bloc, c'est-à-

dire les éléments auxquels s'appliquent les règles de représentation définies dans ce bloc, sont déterminés à partir des éléments concernés par le bloc parent, d'une façon que nous allons maintenant détailler. Lorsqu'un bloc est directement contenu dans l'objet Modelisation, les éléments concernés par ce bloc sont déterminés à partir de l'ensemble des entités géographiques appartenant à l'un des types reliés à cet objet Modelisation.

Un bloc modélisation peut posséder soit un ensemble de règles de découpage, soit une[1] contrainte d'agrégation, soit aucun des deux. Il peut par ailleurs posséder jusqu'à trois contraintes de sélection : sélection globale, sélection des culs-de-sac et sélection des parcours secondaires. Si un bloc ne possède ni règle de découpage ni contrainte d'agrégation, l'ensemble des éléments du terrain auxquels il s'applique est simplement un[2] résultat de l'application des contraintes de sélection à l'ensemble des éléments correspondant au bloc parent.

Si un bloc possède des règles de découpage, il s'applique soit aux sections soit aux limites obtenues en appliquant les règles de découpage à l'ensemble des éléments correspondant au bloc parent et filtrées ensuite par les éventuelles contraintes de sélection. Un attribut indique si le bloc s'applique aux sections ou aux limites ; une même règle de découpage peut donc être liée à deux blocs modélisation, un pour les sections et un pour les limites[3].

Enfin, si un bloc possède une contrainte d'agrégation, il s'applique aux agrégats qui vérifient les contraintes de sélection.

Règles d'instanciation

En plus des divers attributs et relations qui permettent de décrire l'ensemble des éléments auquel il s'applique, un bloc modélisation peut contenir un certain nombre de règles d'instanciation[4]. Une règle d'instanciation est associée à une classe de la base et s'applique à un certain ensemble d'éléments ; elle indique qu'une instance de cette classe doit être créée pour chacun de ces éléments. La règle peut être conditionnelle ou non. Si elle ne l'est pas, elle s'applique à tous les éléments du terrain correspondant au bloc modélisation auquel elle appartient. Si elle l'est, elle s'applique à ceux de ces éléments qui vérifient une certaine contrainte.

Une règle d'instanciation contient un objet AffectationAttribut pour chaque attribut de la classe de la base correspondante ; un objet AffectationAttribut associe à un attribut une expression qui détermine sa valeur en fonction de l'élément du terrain considéré.

1. La contrainte d'agrégation peut être complexe, donc constituée d'un ensemble de contraintes élémentaires reliées par des opérateurs logiques. Pour les règles de découpage, il n'y a pas a priori d'opérateur logique (quand il y a plusieurs critères, ils doivent tous être appliqués), donc il n'est pas nécessaire de regrouper l'ensemble des règles dans un seul objet.

2. Rappelons que ce filtrage n'est pas déterministe car les contraintes de sélection peuvent laisser des choix ouverts. L'ensemble des éléments concernés par un bloc de modélisation n'est donc pas totalement déterminé par les spécifications, c'est pourquoi nous disons *un* résultat et non *le* résultat (voir figure 3.3).

3. Les règles de découpage ne sont pas non plus déterministes, mais il est bien entendu que l'ensemble des sections et l'ensemble des limites issus d'une règle doivent correspondre à une seule et même application de la règle.

4. En fait, un bloc modélisation *doit* logiquement contenir soit au moins une règle d'instanciation soit au moins un sous-bloc (même si actuellement rien n'empêche de définir des blocs vides).

Le lien « participe à » entre la classe ClasseBase et la classe Modelisation n'ajoute aucune information supplémentaire : normalement, un objet Modelisation est lié par cette relation à une classe de la base si et seulement si il contient au moins une règle d'instanciation concernant cette classe. Pour l'instant, notre prototype se contente de créer des instances de cette relation, si elles n'existent pas déjà, lors de la lecture des règles d'instanciation, mais d'autres comportements sont envisageables. Sachant que la liste des classes de la base concernées par une certaine procédure de représentation doit de toute façon avoir été établie avant la rédaction de cette procédure, il pourrait notamment être utile, à des fins de vérification, de déclencher une erreur si l'on rencontre une règle d'instanciation concernant une classe qui n'a pas été reliée préalablement à l'objet Modelisation. Inversement, on pourrait vérifier que toutes les classes déclarées comme « participant à » la modélisation sont effectivement instanciées à au moins un endroit de la procédure.

Contraintes et expressions

La structure utilisée pour représenter les contraintes et les expressions (figure 4.4) s'appuie sur la structure du langage défini au chapitre 3. La classe abstraite *Contrainte* se spécialise en plusieurs sous-classes. L'une de ces sous-classes est la contrainte complexe, qui est composée de plusieurs sous-contraintes reliées par un opérateur logique. Les autres correspondent aux différents types de contraintes élémentaires décrits en 3.3.3.

La contrainte descriptive n'a qu'un attribut qui est la description, sous forme de texte, de la contrainte. La contrainte « base de données » (ContrainteBD) est en relation avec une classe de la base de données. Une entité est considérée comme vérifiant la contrainte s'il existe dans la base une instance de cette classe qui représente l'entité.

Les contraintes de nature et de relation ont été regroupées dans une classe abstraite ContrainteNatureOuRelation car elles partagent le fait d'être en relation avec un type d'entité géographique. La contrainte de relation peut de plus posséder une contrainte sur l'entité en relation : cela signifie que la relation doit avoir lieu avec une entité (du type indiqué par la relation « relation avec ») qui vérifie cette contrainte. S'il n'y a pas de telle contrainte, la relation doit simplement avoir lieu avec n'importe quelle entité du type indiqué. Les sous-classes de ContrainteRelation correspondent aux différents types de contraintes contextuelles décrits en 3.3.3.

Les contraintes sur propriété sont de deux sortes. Les contraintes sur propriété simple comparent simplement la valeur de la propriété à une valeur seuil qui est un attribut de la contrainte, à l'aide d'un opérateur de comparaison qui est lui aussi un attribut. Les contraintes sur propriété complexe sont reliées à un autre objet Contrainte, qui indique ce que doit vérifier la valeur de la propriété ; cette valeur est supposée être une entité géographique. Il serait possible de vérifier que c'est le cas en utilisant l'attribut type de la classe Propriete. Par exemple, « mur_d_enceinte.hauteur > 5 "m" » constitue une contrainte sur propriété complexe, où « mur_d_enceinte » est la propriété et « hauteur > 5 "m" » la contrainte que doit vérifier la valeur de la propriété. Cette dernière contrainte est elle-même une contrainte sur propriété simple, où « hauteur » est la propriété, « 5 "m" » le seuil et « > » l'opérateur de comparaison.

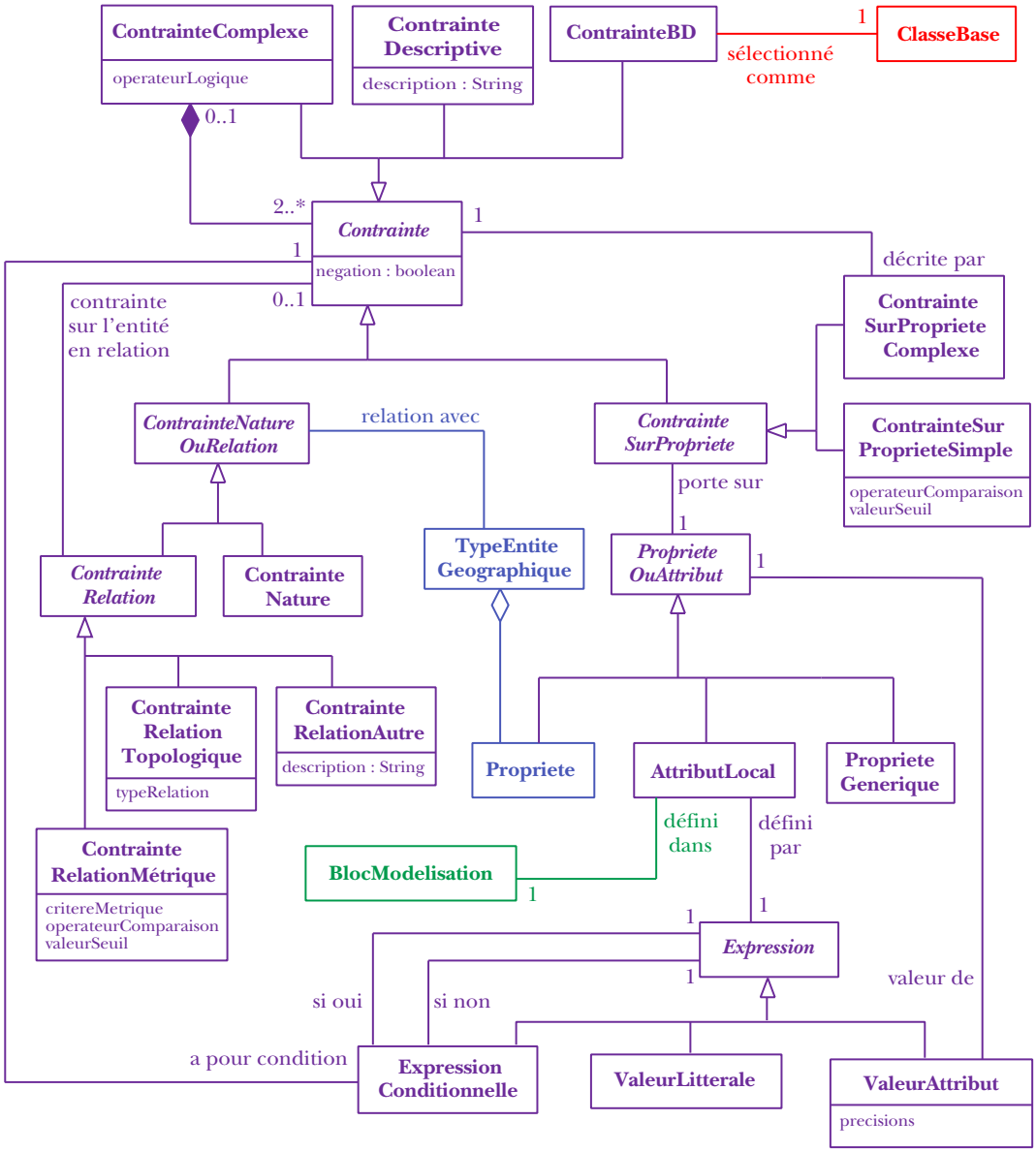


FIG. 4.4 *Diagramme UML représentant les différents types de contraintes et d'expressions.*

Les contraintes sur propriété peuvent porter sur une propriété ou sur un attribut local, défini dans une section *attributs* de la procédure de représentation (3.3.2). En principe, une contrainte sur propriété ne peut porter que sur une propriété explicitement déclarée comme appartenant à tous les types d'entités géographique concernés par la contrainte. Cependant, il existe un certain nombre de propriétés génériques que toute entité géographique ou presque est susceptible de posséder, et sur lesquelles les contraintes portent fréquemment. Il s'agit notamment des propriétés géométriques : longueur, largeur, hauteur, centre, sommet, etc. Pour alléger le processus de formalisation des spécifications, ces propriétés peuvent être référen-

cées sans être déclarées auparavant. Elles seront alors représentées, dans la structure générée à partir de notre langage, par des objets de classe ProprieteGenerique. Dans le cas où la contrainte ne porte pas sur une propriété mais sur un attribut local, cet attribut doit bien entendu appartenir à la même procédure de représentation que la contrainte.

Les attributs locaux sont définis par des expressions (3.3.2). Rappelons que les expressions sont également utilisées, dans les règles d'instanciation, pour l'affectation des valeurs aux attributs de la base de données (figure 4.3, en bas à gauche). Une expression peut être une expression conditionnelle, qui comprend une contrainte ainsi que deux sous-expressions indiquant les valeurs à retourner suivant que la contrainte est ou non vérifiée par l'entité. Elle peut être une simple valeur littérale, qui pourrait être munie d'un type afin de permettre des vérifications. Dans notre prototype, une valeur littérale est soit un booléen, soit une chaîne de caractères, soit un nombre éventuellement accompagné d'une unité, mais aucune vérification n'est faite. Elle peut, enfin, être une référence à une propriété ou un attribut local. Dans ce cas, des précisions indiquent comment déterminer la valeur de cette propriété ou de cet attribut. Dans notre prototype, ces précisions ne sont qu'une chaîne de caractères, cependant il serait intéressant d'en formaliser le contenu plus avant.

Les attributs locaux possèdent également une référence à un bloc modélisation, qui indique le contexte dans lequel ils ont été définis.

Rappelons que, en plus des contraintes sur propriété et des expressions, les propriétés et attributs locaux peuvent être référencés par les règles de découpage et les contraintes d'agrégation, dont nous n'avons pas développé la structure dans notre prototype.

Maintenant que nous avons décrit notre structure orientée-objet pour la manipulation des procédures de représentation, nous allons indiquer brièvement comment on passe de la description des procédures dans notre langage à leur représentation dans cette structure.

4.1.5 Interprétation du langage de description des procédures de représentation

Nous avons utilisé, pour interpréter notre langage et générer la structure associée, un générateur de parseurs en java, javaCC[1]. Ce générateur permet de définir une fonction associée à chaque règle de production du langage, cette fonction pouvant prendre des paramètres et renvoyer une valeur. Il est ainsi possible de passer des informations d'une règle à l'autre, dans les deux sens.

Nous avons défini une classe Contexte pour regrouper les informations contextuelles nécessaires à la lecture du langage. Cette classe comporte une partie statique représentant le contexte général, c'est-à-dire la liste des types d'entités géographiques et celle des classes des bases. Chaque type d'entité et chaque classe de la base est associé à un identifiant, qui est obtenu à partir du nom du type ou de la classe en le convertissant en minuscules, supprimant les accents puis remplaçant tous les caractères non alphabétiques par des blancs soulignés (_). Ce sont par les identifiants ainsi déterminés que doivent être appelés les types et classes dans la procédure de modéli-

1. disponible sur `https://javacc.dev.java.net/`

sation, afin qu'ils soient reconnus par le parseur. Deux classes de base peuvent avoir le même nom (ou des noms conduisant au même identifiant) du moment qu'elles n'appartiennent pas à la même base de données ; deux types d'entités géographiques ne peuvent pas avoir des noms conduisant à des identifiants semblables.

Cette classe contient également une partie dynamique, spécifique à chaque procédure de représentation, qui est passée en argument aux règles de production qui en ont besoin. Cette partie dynamique comprend une référence à l'objet Modelisation auquel appartient la procédure de représentation, et une référence à l'objet BlocModelisation en cours. Elle comprend également une liste des propriétés et des attributs locaux utilisables, associés à des identifiants. L'identifiant d'une propriété est généré de la même façon que ceux des types d'entités géographiques. À l'intérieur d'une règle d'instanciation, elle contient enfin la liste des attributs de la classe concernée par la règle, également repérés par un identifiant.

Une instance de cette partie dynamique est créée au début de la lecture de la procédure de modélisation ; l'ensemble des propriétés accessibles est déterminé en prenant l'intersection des ensembles de propriétés de chacun des types d'entités géographiques concernés. Un premier objet BlocModelisation est également créé, avec pour parent l'objet Modelisation.

Lors de la lecture d'une section *sélection* (3.3.4), les contraintes définies dans cette section sont affectées à l'objet BlocModelisation en cours. Lors de la lecture d'une contrainte sur propriété, la propriété (ou l'attribut local) est cherchée dans le contexte actuel. S'il s'agit d'une contrainte sur propriété complexe, la sous-contrainte qui la décrit est lue avec un contexte vide : seules les propriétés génériques sont accessibles. Lors de la lecture d'une contrainte de nature ou de relation, le type d'entité géographique est cherché dans le contexte général. S'il y a une contrainte sur l'entité en relation, celle-ci est lue avec un contexte contenant seulement les propriétés du type en relation.

Lors de la lecture d'une section *attributs* (3.3.2), un nouvel attribut local est créé pour chaque définition ; il est associé au bloc modélisation en cours et est ajouté au contexte.

Lors de la lecture d'une section *agrégation* (3.3.6), un nouvel objet BlocModelisation est créé avec pour bloc parent le bloc en cours. La contrainte d'agrégation lui est associée et il devient le bloc en cours jusqu'à la mention « Fin agrégation ».

Lors de la lecture d'une section *découpage* (3.3.5), de nouveaux objets BlocModelisation sont créés à la lecture de chacune des éventuelles mentions « sections : » et « limites : ». L'ensemble des règles de découpage leur est associé. Le bloc correspondant à l'une de ces mentions devient le bloc en cours jusqu'à l'autre mention ou à la mention « Fin découpage ».

Enfin, lors de la lecture d'une section *instanciation* (3.3.7), pour chaque règle d'instanciation la liste des attributs de la classe correspondante est chargée dans le contexte avant la lecture des affectations d'attributs. Dans le cas de règles d'instanciation conditionnelles, la contrainte de la partie « Si » est lue et associée à toutes les règles d'instanciation de la partie « Alors », puis sa négation est associée aux règles de la partie « Sinon ». Si plusieurs conditions sont imbriquées, les différentes contraintes sont combinées au sein de contraintes complexes de manière à toujours associer au plus une contrainte à chaque règle d'instanciation. D'autre part, chaque

classe de la base concernée par une règle d'instanciation est directement reliée à l'objet Modelisation par la relation « participe à » (figure 4.3, en haut à gauche).

4.2 Exemples

Nous allons détailler dans cette partie la réalisation de deux exemples de procédures de représentation : la représentation des cours d'eau dans la BDCarto et dans la BDTopo Pays. Nous présenterons ainsi certains des choix possibles lors de l'utilisation de notre modèle, les possibilités qu'il offre déjà et certaines de ses limites.

4.2.1 Cours d'eau dans la BDCarto

La figure 4.5, extraite de la figure 3.8[1], nous montre que quatre types de COURS D'EAU sont représentés par la BDCarto : RIVIÈRE, CANAL, RUISSEAU et AQUEDUC. Ces COURS D'EAU sont représentés par quatre classes différentes : [tronçon hydrographique], [nœud hydrographique][2], [cours d'eau] et enfin [zone d'occupation du sol]. Notons que les trois premières représentations ne sont pas indépendantes : en effet un [cours d'eau] est composé de [tronçons hydrographiques], et un [tronçon hydrographique] est toujours lié à deux [nœuds hydrographiques], le nœud initial et le nœud final. Notre modèle ne permet pas actuellement de décrire l'instanciation des relations entre classes de la base, ceci ne sera donc pas apparent dans la procédure de représentation.

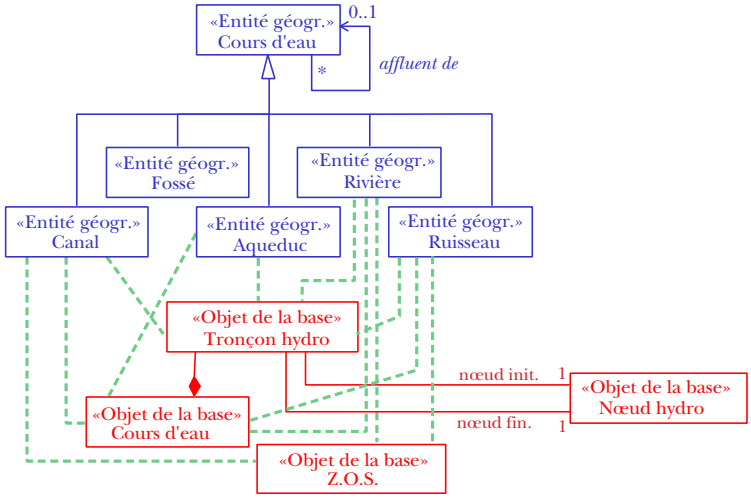


FIG. 4.5 *Représentation des* COURS D'EAU *dans la BDCarto.*

Nous avons tout d'abord saisi dans notre interface les différents éléments représentés sur la figure 4.5. La figure 4.6 montre les éditeurs d'objets pour la classe [cours

1. et complétée avec le type d'entité géographique RUISSEAU.
2. En fait, il n'y a pas de lien direct avec la classe [nœud hydrographique], car les spécifications de cette classe n'indiquent pas directement qu'elle participe à la représentation des COURS D'EAU en tant que tels (la classe est définie comme représentant « une modification de l'écoulement de l'eau »), mais la double relation nœud initial/nœud final nous conduit à la prendre en compte.

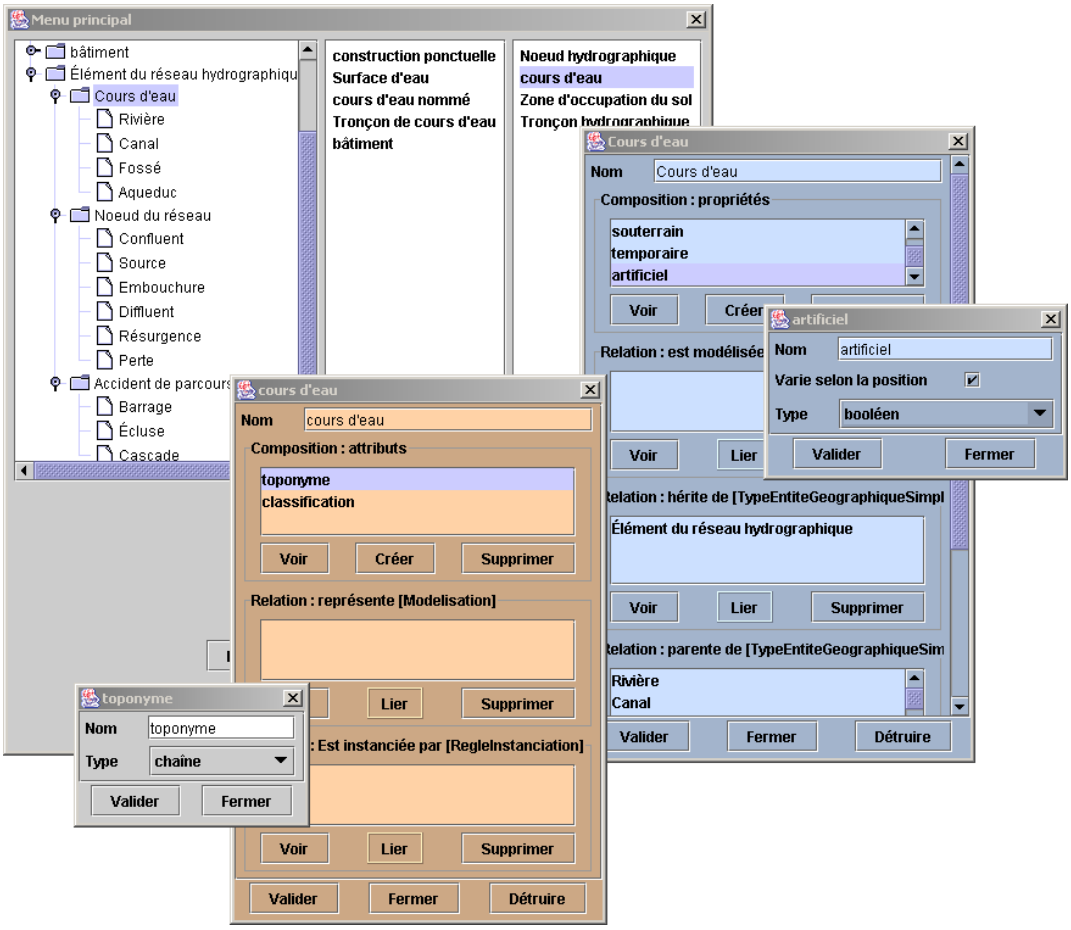


FIG. 4.6 *Type d'entité géographique* COURS D'EAU *(en bleu) et classe de la BDCarto [cours d'eau] (en orange) dans notre interface*

d'eau] et le type d'entité géographique COURS D'EAU. Nous avons défini une interface provisoire (fenêtre « Menu principal » sur la figure) qui affiche tous les types d'entités géographiques définis sous forme d'arbre, ainsi que toutes les classes définies dans chacune des deux bases (BDTopo Pays à gauche, BDCarto à droite). Toutes ces informations sont persistantes.

Ensuite, nous avons créé un objet Modélisation que nous avons relié aux types d'entités géographiques concernés (figure 4.7), et nous avons tapé dans cet objet la procédure de représentation que nous allons maintenant décrire[1].

Critères de sélection

Le critère de sélection est assez simple à exprimer étant donné que nous avons introduit dans notre modèle la possibilité d'indiquer des contraintes spécifiques pour les culs-de-sac et les parcours secondaires : *La BDCarto contient : [...]*

1. Nous avons utilisé dans notre implémentation du langage de description des procédures de représentation les caractères « // » pour introduire les commentaires.

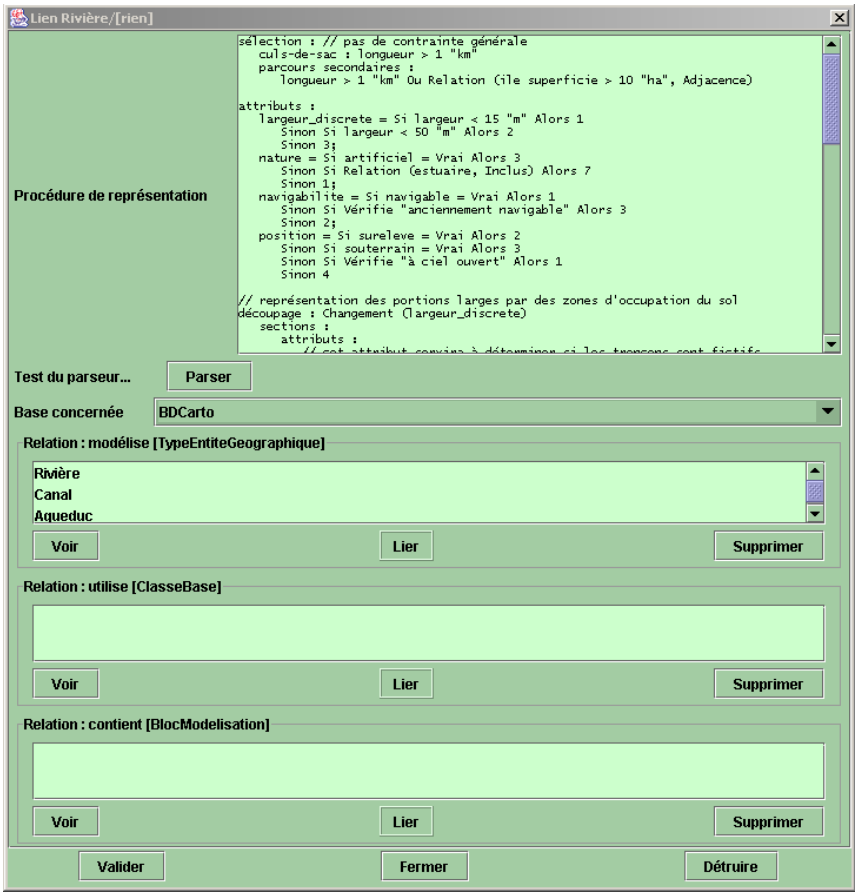


FIG. 4.7 *Objet Modélisation contenant la procédure de représentation des* COURS D'EAU *dans la BDCarto.*

— *tous les axes principaux, à l'exception des « culs-de-sac » d'une longueur inférieure à un kilomètre […]*

— *outre l'axe principal, les axes des bras secondaires d'une longueur supérieure à un kilomètre ou qui délimitent une île d'une superficie supérieure à dix hectares quand un cours d'eau se subdivise en plusieurs.*

On peut l'exprimer ainsi :

```
sélection : // pas de contrainte générale
   culs-de-sac : longueur > 1 "km"
   parcours secondaires :
      longueur > 1 "km" Ou Relation
         (ile superficie > 10 "ha", Adjacence)
```

Nous avons choisi dans cet exemple d'utiliser une contrainte topologique pour décrire la relation entre l'ÎLE et le COURS D'EAU ; en effet un bras de cours d'eau « délimite » une île s'il a une frontière commune avec cette île. Cependant ce choix peut être discuté. L'avantage, par rapport à une simple chaîne de caractères, est que la relation pourra être plus facilement traitée par un ordinateur ; l'inconvénient est

qu'il y a une perte sémantique par rapport au terme « délimite ». Une meilleure solution pourrait être, si l'on dispose d'une ontologie structurée avec des relations entre les concepts, de définir la relation « un cours d'eau délimite une île » dans l'ontologie. Il faudrait alors étendre notre langage pour permettre l'utilisation, dans des contraintes de relation, de types de relations définis dans l'ontologie.

Attributs locaux

La première de nos classes, [tronçon hydrographique], correspond à un découpage aux changements de valeurs d'attributs. Pour pouvoir spécifier ce découpage, il est donc nécessaire de définir d'abord ces attributs, sous forme d'attributs locaux.

Regardons donc les attributs de la classe [tronçon hydrographique]. L'attribut [état] indique si le cours d'eau est permanent, temporaire ou fictif. Nous avons défini dans l'ontologie TEMPORAIRE comme une propriété, booléenne, du COURS D'EAU. L'[état] est considéré comme *fictif* si le [tronçon hydrographique] est situé à l'intérieur d'une [zone d'occupation du sol] de poste 51 (voir figure 2.6-b page 27). Cette valeur d'[état] ne peut donc être déterminée que par les règles d'instanciation de la [zone d'occupation du sol]. En regardant les spécifications de cette dernière classe, on voit qu'elle concerne les sections du COURS D'EAU de plus de 50 mètres de large couvrant au moins 4 hectares. Le critère de superficie portant sur des sections de cours d'eau individuelles, il ne peut être exprimé globalement; il devra être évalué au sein d'une section *découpage* de notre procédure de représentation.

Le second attribut, [largeur], sert simplement à classer les tronçons en trois catégories suivant leur LARGEUR. Un attribut local ne peut porter le même nom qu'une propriété, nous avons donc défini l'attribut `largeur_discrete` :

```
attributs :
   largeur_discrete = Si largeur < 15 "m" Alors 1
      Sinon Si largeur < 50 "m" Alors 2
      Sinon 3;
```

Le troisième attribut, [nature], distingue les COURS D'EAU en quatre catégories : cours d'eau naturels (1), voies d'eau artificielles (2), aqueducs (3), et estuaires (7). Nous utilisons une contrainte de nature pour la valeur 3, aqueduc. Pour les deux premières, nous avons préféré définir ARTIFICIEL comme une propriété du COURS D'EAU plutôt que d'utiliser une contrainte de nature et le type d'entité CANAL, en raison d'un doute sur le fait que les notions de voie d'eau artificielle et de canal se recouvrent exactement. Cependant, l'autre choix était possible. Notons que si l'on utilisait une ontologie comportant des connaissances sur les différents types d'entités, une telle ontologie pourrait par exemple savoir que pour un CANAL la propriété ARTIFICIEL vaut toujours Vrai. Le système pourrait alors utiliser cette connaissance pour déduire automatiquement que dans la représentation d'un CANAL, l'attribut [nature] des [tronçons hydrographiques] vaut toujours 3.

La dernière valeur possible de l'attribut [nature], 7, indique l'écoulement d'un COURS D'EAU dans la zone d'estran. Nous proposons de considérer qu'ESTUAIRE est un type d'entité géographique et d'utiliser une contrainte de relation pour cette dernière valeur.

Nous définissons donc ainsi l'attribut local `nature` :

```
nature = Si Est aqueduc Alors 4
   Sinon Si artificiel = Vrai Alors 3
   Sinon Si Relation (estuaire, Inclus) Alors 7
   Sinon 1;
```

Le quatrième attribut, [navigabilité], peut être défini par la propriété NAVIGABLE, que nous avons associée dans l'ontologie au type général ÉLÉMENT DU RÉSEAU HYDROGRAPHIQUE. Cependant, cet attribut prend une valeur particulière pour les voies d'eau *anciennement* navigables. Nous utilisons une contrainte descriptive pour exprimer cette possibilité.

```
navigabilite = Si navigable = Vrai Alors 1
   Sinon Si Vérifie "anciennement navigable" Alors 3
   Sinon 2;
```

Enfin, le cinquième attribut, [position par rapport au sol], indique si le COURS D'EAU passe sur un pont ou sous terre. On pourrait envisager de définir un tel attribut en utilisant une contrainte de relation métrique sur la dénivelée entre l'entité et le sol, avec un seuil nul, cependant ce n'est pas très satisfaisant et oblige à considérer le sol comme un type d'entité géographique. Une autre possibilité, préférable à notre avis, aurait pu être de définir une propriété générique ÉLÉVATION qui correspondrait à la différence d'altitude entre une entité et le sol. Nous avons choisi dans notre exemple une troisième possibilité consistant à définir deux propriétés booléennes, SURÉLEVÉ et SOUTERRAIN. Cette dernière possibilité nous semble être la plus proche des spécifications de départ, mais il s'agit néanmoins d'un choix arbitraire.

D'autre part, cet attribut distingue les COURS D'EAU au sol à ciel ouvert de ceux qui passent dans des tuyaux posés au sol. Nous utilisons une contrainte descriptive pour cette dernière distinction.

```
position = Si sureleve = Vrai Alors 2
   Sinon Si souterrain = Vrai Alors 3
   Sinon Si Vérifie "à ciel ouvert" Alors 1
   Sinon 4
```

Découpages et instanciation

Nous avons vu qu'il était nécessaire de connaître les parties du COURS D'EAU qui seraient représentées par des [zones d'occupation du sol] avant de pouvoir spécifier le découpage en [tronçons hydrographiques]. Nous allons donc maintenant décrire la représentation par les [zones d'occupation du sol]. Le découpage consiste simplement à distinguer les zones de largeur supérieure à 50 mètres et celles de largeur inférieure ; il n'y a pas a priori de découpage aux intersections du réseau, la représentation surfacique étant presque totalement indépendante de la représentation sous forme de réseau. L'attribut `largeur_discrete` permet de faire le découpage nécessaire puisque les portions qui nous intéressent sont exactement celles où cet attribut vaut 3 (LARGEUR > 50 m).

```
// représentation des portions larges par des zones d'occ. du sol
découpage : Changement (largeur_discrete)
```

On peut remarquer qu'aucune longueur minimale n'est spécifiée pour la prise en compte du changement d'attribut, ce qui ne signifie pas qu'il n'en existe pas dans la pratique : il est vraisemblable que le fait qu'un fleuve repasse pour quelques mètres en-dessous du seuil de 50 m au milieu d'une section large n'empêchera pas qu'on saisisse une seule [zone d'occupation du sol]. L'absence du seuil indique donc que le seuil est inconnu (non spécifié), et non qu'il est nul.

On a pu voir sur la figure 2.6-b (page 27) que la zone d'occupation du sol peut être interrompue par la présence d'un pont, toutefois ceci n'est pas dû à une règle de découpage à proprement parler mais à un conflit entre zone de bâti et zone d'eau. En effet, les [zones d'occupation du sol] de la BDCarto forment une partition de l'espace géographique et il est donc nécessaire de choisir entre eau et bâti. La façon dont ce choix doit être opéré n'est pas décrite dans les spécifications actuelles, nous ne pouvons donc pas le mentionner dans les spécifications formalisées.

Les sections de ce découpage sont représentées par des [zones d'occupation du sol] si et seulement si elles font plus de 50 mètres de large et couvrent au moins 4 hectares. Il nous faut représenter cette condition d'instanciation par un attribut local afin de pouvoir l'utiliser plus tard dans le découpage en [tronçons hydrographiques] :

```
découpage : Changement (largeur_discrete)
   sections :
      attributs :
// cet attribut servira à déterminer si les tronçons sont fictifs
         large = Si largeur_discrete = 3 Et superficie > 4 "ha"
            Alors Vrai Sinon Faux
      instanciation :
         Si large = Vrai
         Alors Instancie zone_d_occupation_du_sol
            (geometrie = contour, poste = 51) // 51 = eau libre
Fin découpage
```

Nous pouvons maintenant spécifier le découpage en [tronçons hydrographiques]. Les tronçons sont découpés selon leurs valeurs d'attributs avec un seuil d'un kilomètre, comme dit précédemment. Les attributs à prendre en compte sont `largeur_discrete`, `nature`, `navigabilite` et `position`, ainsi que TEMPORAIRE et `large` qui correspondent à l'attribut [état]. Cependant `large` ne peut changer de valeur que quand `largeur_discrete` change de valeur, il est donc inutile de le mentionner dans la règle de découpage.

Les règles de découpage suivantes ne sont pas mentionnées explicitement en tant que telles dans les spécifications actuelles, mais elles sont implicites dans la structure de la représentation topologique du réseau en tronçons et en nœuds. En effet, cette structure signifie que la présence d'un [nœud hydrographique] à un endroit donné implique un découpage, le nœud ne pouvant se situer qu'à l'extrémité d'un tronçon. Nous devons donc regarder les spécifications de la classe [nœud hydrographique].

Cette classe représente en premier lieu « les confluences, diffluences, sources, embouchures et pertes de cours d'eau retenus dans la BDCarto ». Il s'agit très précisément des intersections du réseau sélectionné ; on découpe donc aux intersections du réseau. Elle représente ensuite des BARRAGES, CASCADES et ÉCLUSES avec divers

critères de sélection. Nous avons regroupé tous ces types d'entités dans le type ACCIDENT DE PARCOURS, et le critère qui importe pour le découpage en [tronçons] est qu'il existe effectivement un objet [nœud hydrographique] dans la base : il s'agit d'une contrainte « base de données ». Le découpage est donc décrit par :

```
//représentation en tronçons et noeuds
découpage :
   Changement > 1 "km" (temporaire, largeur_discrete, nature,
      navigabilite, position);
   Intersections;
   Rencontre
      accident_de_parcours Sélectionné comme noeud_hydrographique
```

Remarquons qu'il aurait été possible de remplacer la mention Intersections par une règle de rencontre d'un NŒUD DU RÉSEAU avec la contrainte que ce dernier soit sélectionné dans la base comme [nœud hydrographique]. Cependant le découpage aux intersections du réseau sélectionné apparaît dans presque toutes les représentations de réseau, c'est pourquoi nous avons jugé utile d'en faire une règle générique. Par ailleurs, la règle de sélection des CONFLUENTS et DIFFLUENTS devra de toute façon être exprimée pour la BDCarto, car ces entités y sont représentées en tant que telles ; mais ce n'est pas le cas pour d'autres bases comme la BDTopo Pays.

Nous pouvons maintenant spécifier la règle d'instanciation du [tronçon hydrographique]. AXE est une propriété générique (voir page 92).

```
sections :
   instanciation :
      Instancie troncon_hydrographique (
      // l'état 3 signifie "fictif", c'est-à-dire que le cours d'eau
      // est large et représenté par une zone d'occupation du sol.
         etat = Si large = Vrai Alors 3
            Sinon Si temporaire = Vrai Alors 2 Sinon 1,
         largeur = largeur_discrete,
         nature = nature,
         navigabilite = navigabilite,
         position_par_rapport_au_sol = position,
         geometrie = axe
      )
```

Certaines des limites de sections de ce découpage correspondent à des entités géographiques représentées par des [nœuds hydrographiques], comme on l'a vu. Ces représentations seront décrites dans les procédures de représentation correspondantes. Cependant, des [nœuds hydrographiques] doivent également être créés aux *autres* limites de section du découpage. Ceci n'est pas écrit explicitement dans les spécifications actuelles, mais est sous-entendu par la structure de la représentation du réseau, et également par l'existence d'une valeur « changement d'attribut » parmi les valeurs possibles de l'attribut [nature] du [tronçon hydrographique]. Ces nœuds ne font que participer à la représentation du COURS D'EAU et ne représentent aucune autre entité, nous devons donc les mentionner ici. CENTRE est une propriété générique.

```
limites :
   instanciation :
   // on crée un noeud de changement d'attribut seulement s'il n'y a
   // pas déjà un noeud à cet endroit pour une autre raison
      Si Non Relation
         (noeud_du_reseau Sélectionné comme noeud_hydrographique,
         Superposition)
      Alors Instancie noeud_hydrographique
         (nature = 50, // changement d'attribut
         geometrie = centre ("sur l'axe"),
         toponyme = "",
         classification = 9, // sans importance cartographique
         caractere_touristique = 2) // non touristique
Fin découpage
```

Enfin, le [cours d'eau] est défini dans les spécifications comme représentant une « portion connexe du réseau hydrographique liée à un toponyme, possédant une source ou origine et un confluent ou embouchure. » Il suffit donc de découper suivant la propriété TOPONYME. Pour l'instanciation, nous avons dit plus haut que notre modèle ne permettait pas actuellement de décrire l'instanciation des relations entre classes de la base. Nous ne pouvons donc pas écrire que le [cours d'eau] ne possède pas de géométrie propre mais est composé de [tronçons hydrographiques]. La seule chose qu'on pourrait faire est de représenter dans la procédure le découpage en [tronçons hydrographiques] à l'intérieur du découpage en [cours d'eau], ce qui signifierait que le premier est un sous-découpage du second, comme ceci :

```
découpage : Changement (toponyme)
  sections :
     instanciation : Instancie cours_d_eau [...]
     découpage : Changement > 1 "km" (temporaire, [...])
        sections :
           instanciation : Instancie troncon_hydrographique [...]
        limites : [...]
     Fin découpage
Fin découpage
```

Cependant, tel que notre langage a été défini pour le moment, une telle construction n'indique rien de particulier sur les relations entre les instances définies dans le premier découpage et celles définies dans le sous-découpage. Pour que cette construction soit intéressante, il faudrait donc lui ajouter une sémantique particulière. Sinon, la structure obtenue sera différente mais la signification sera strictement équivalente à la spécification séparée des deux découpages. Nous avons donc spécifié dans notre exemple les deux découpages séparément, ce qui simplifie la procédure de représentation, même si l'autre choix pourrait être justifié par le fait que la structure obtenue reproduirait davantage celle de la base de données[1].

1. En fait, dans le cas général et pour des règles de découpage comprenant un choix arbitraire (donc non déterministes), comme la plupart des règles de découpage à un changement d'attribut, la définition d'un sous-découpage a l'avantage de signifier explicitement que le second découpage tient compte des choix arbitraires effectués pour le premier. L'écriture sous forme de deux découpages

Notons que pour que la solution des deux découpages indépendants soit vraiment équivalente à celle du sous-découpage, il faudrait a priori ajouter la ligne `Changement (toponyme)` aux règles du découpage en tronçons. Nous ne l'avons pas fait car nous considérons, et les spécifications actuelles sous-entendent, qu'un COURS D'EAU n'est pas censé changer de TOPONYME ailleurs qu'à une intersection du réseau. Plus exactement, selon les spécifications actuelles, un COURS D'EAU peut posséder plusieurs toponymes locaux, appelés *alias*, mais il ne peut posséder qu'un seul TOPONYME ; et par ailleurs, un COURS D'EAU ne peut commencer et se terminer qu'à une intersection (ou une extrémité) du réseau. Ces deux propriétés font partie de la définition du COURS D'EAU ; utiliser une autre définition reviendrait à changer d'ontologie, et il faudrait alors modifier la procédure de représentation en conséquence.

En plus de cette relation de composition, le [cours d'eau] possède deux attributs. Le premier, [toponyme], ne pose pas de problème puisque TOPONYME est une propriété. En revanche, le second, [classification], n'est pas définissable de façon très satisfaisante par notre modèle. La définition est la suivante :

Classification

La classification établit une hiérarchie décroissante entre les cours d'eau. Par construction, tout cours d'eau de niveau n se jette dans un cours d'eau de niveau inférieur ou égal à n.

1. *tout cours d'eau naturel se jetant dans une « embouchure logique »*[1] *ou d'une longueur supérieure à 100 km ou d'une longueur quelconque mais en aval d'un cours d'eau de niveau 1, et tout cours d'eau artificiel d'une longueur supérieure à 100 km,*
2. *tout cours d'eau naturel d'une longueur comprise entre 50 et 100 km, ou en amont d'un cours d'eau de niveau 1 ou en aval d'un cours d'eau de niveau 2 (hors niveau 1), et tout cours d'eau artificiel d'une longueur comprise entre 50 et 100 km,*
3. *tout cours d'eau naturel en amont d'un cours d'eau de niveau 2 ou en aval d'un cours d'eau de niveau 3 (hors niveaux 1 et 2), et tout cours d'eau artificiel d'une longueur comprise entre 25 et 50 km,*
4. *tout cours d'eau naturel en amont d'un cours d'eau de niveau 3 ou en aval d'un cours d'eau de niveau 4 (hors niveaux 1 à 3), et tout cours d'eau artificiel d'une longueur comprise entre 10 et 25 km,*
5. *tout cours d'eau naturel en amont d'un cours d'eau de niveau 4 ou en aval d'un cours d'eau de niveau 5 (hors niveaux 1 à 4), et tout cours d'eau artificiel d'une longueur comprise entre 5 et 10 km,*
6. *tous les autres cours d'eau.*

Le problème posé par cette définition d'attribut est qu'elle est récursive. Notre modèle ne permet pas actuellement ce type de définition. Plus généralement, il n'est

indépendants pourrait autoriser que le second découpage ne soit pas réellement un sous-découpage du premier, même si les règles de découpage semblent l'impliquer, en raison de choix différents. Ici le problème ne se pose pas vraiment, pour deux raisons. D'une part, le découpage aux intersections du réseau sélectionné ne permet a priori pas de choix arbitraires (plus précisément, ces choix ont déjà été faits lors de l'étape de sélection). D'autre part, on ne sait de toute façon pas décrire l'instanciation de la relation, et l'objet [cours d'eau] ne possède pas de géométrie propre, c'est-à-dire qu'on n'utilise aucune information géométrique sur le découpage selon le toponyme ; par conséquent, la précision géométrique de ce découpage n'a aucune importance.

1. Une embouchure logique est une interruption du réseau formé par les cours d'eau naturels : mer, puits...

pas actuellement possible dans notre modèle de faire référence à un *attribut* d'une autre entité, mais seulement à une de ses *propriétés*. Généralement, il est possible de transformer une contrainte exprimée, dans les spécifications actuelles, sur les attributs d'une autre entité en contrainte sur les propriétés de cette entité, en utilisant la définition des attributs. Cela nous semble même souhaitable, au moins dans les cas simples, même si la possibilité de faire référence à des attributs locaux définis dans d'autres procédures de représentation est une extension envisageable de notre modèle[1]. Cependant, le remplacement d'un attribut par sa définition n'est clairement pas applicable dans le cas d'une définition récursive. En fait, dans ce cas précis, il est théoriquement possible d'en exprimer la définition sans récursivité, car le nombre de valeurs est fini et la définition des cas concernés par une valeur n'utilise que les valeurs inférieures ; cependant une telle définition n'a guère d'intérêt et comprendrait un nombre de lignes considérable :

```
Si longueur > 100 "km"
  Ou (artificiel = Faux Et (
       (Non Relation (cours_d_eau, "affluent de"))
       Ou Relation (cours_d_eau longueur > 100 "km", "en aval de")))
  Alors 1
Sinon Si longueur > 50 "km"
  Ou (artificiel = Faux Et (
       Relation (cours_d_eau [...], "affluent de")
       Ou Relation (cours_d_eau longueur > 50 "km", "en aval de")))
  Alors 2
etc.
```

où « [...] » reprend la contrainte entre le « Si » et le « Alors 1 ». La contrainte correspondant à la valeur 3 devrait reprendre celle correspondant à la valeur 2, et ainsi de suite.

Cette définition n'étant pas utilisable dans la pratique, et notre modèle ne permettant pas la récursion, nous devons nous contenter de contraintes descriptives. On obtient donc pour le [cours d'eau] :

```
// représentation des cours d'eau nommés
// (N.B. dans la base ils sont composés de tronçons ;
// ils n'ont pas de géométrie)
découpage : Changement (toponyme)
  sections :
     instanciation :
        Si toponyme =/= "" Alors Instancie cours_d_eau
           (toponyme = toponyme,
            classification =
               Si longueur > 100 "km" Ou Vérifie
"cours d'eau naturel se jetant dans une embouchure logique
ou en aval d'un cours d'eau de niveau 1"
               Alors 1
```

1. En effet il nous semble préférable, quand c'est possible, de privilégier la description du contenu de la base à partir du terrain le plus directement possible, plutôt que la description des classes de la base les unes par rapport aux autres : il sera ainsi plus facile de relier entre elles des classes de bases différentes.

```
                Sinon Si longueur > 50 "km" Ou Vérifie
"cours d'eau naturel en amont d'un cours d'eau de niveau 1
ou en aval d'un cours d'eau de niveau 2"
                Alors 2
                Sinon Si (artificiel = Vrai Et longueur > 25 "km")
                  Ou Vérifie "..." Alors 3
                Sinon Si (artificiel = Vrai Et longueur > 10 "km")
                  Ou Vérifie "..." Alors 4
                Sinon Si (artificiel = Vrai Et longueur > 5 "km")
                  Ou Vérifie "..." Alors 5 Sinon 6
           )
Fin découpage
```

Ceci termine notre procédure de représentation pour les COURS D'EAU dans la BDCarto. Nous allons maintenant traiter le même exemple sur la BDTopo Pays.

4.2.2 Cours d'eau dans la BDTopo Pays

Nous allons traiter cet exemple beaucoup plus rapidement, car il est en grande partie similaire au précédent. En particulier, la BDTopo Pays possède également une triple représentation des cours d'eau, en tronçons, en surfaces et en cours d'eau nommés (figure 4.8). Les critères de sélection et d'instanciation sont toutefois, bien sûr, différents, notamment par les seuils qui interviennent.

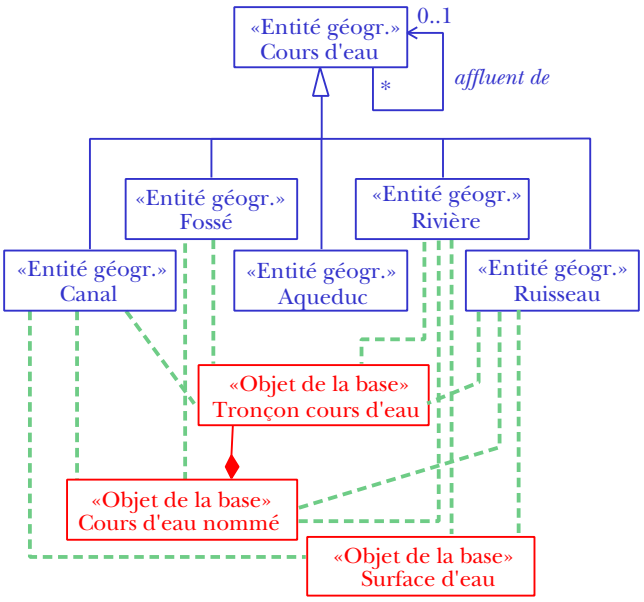


FIG. 4.8 *Représentation des* COURS D'EAU *dans la BDTopo Pays*

Les classes qui interviennent dans la représentation sont cette fois-ci [tronçon de cours d'eau], [surface d'eau] et [cours d'eau nommé]. D'autre part, les types d'entités géographiques concernés par la procédure de représentation ne sont plus RIVÈRE, CANAL et AQUEDUC mais RIVIÈRE, CANAL et FOSSÉ. (La représentation des aqueducs est complètement différente).

Le critère de sélection de la classe [tronçon de cours d'eau] est le suivant : *Tous les cours d'eau permanents [...] sont inclus. Les cours d'eau temporaires naturels sont inclus, à l'exception des tronçons de moins de 200 m situés aux extrémités amont du réseau. Les cours d'eau temporaires artificiels ou artificialisés sont sélectionnés en fonction de leur importance et de l'environnement : les tronçons longeant une voie de communication sont exclus, ainsi que les fossés.*

Nous pouvons représenter ce critère de la façon suivante :

```
sélection :
   temporaire = Faux Ou artificiel = Faux Ou
      (Non (Est fosse Ou Relation (voie_de_communication, "longe")))
   culs-de-sac : temporaire = Faux Ou longueur > 200 "m"
```

Notons que la contrainte « un cours d'eau longe une voie de communication » pourrait être considérée comme une contrainte métrique, par exemple une contrainte sur la distance de Hausdorff entre le cours d'eau et la voie de communication. Cela serait probablement plus compliqué que ça, car le cours d'eau pourrait par exemple être plus court que la voie de communication, et la distance de Hausdorff serait alors grande bien qu'il ne s'éloigne jamais de la voie ; mais cela nous semble néanmoins envisageable.

D'autre part, aucun seuil n'est spécifié. Il peut être intéressant, dans le cas de critères métriques flous comme « près de », d'étudier les données de la base pour voir si un tel critère peut être transformé en critère métrique avec un seuil déterminé par fouille de données. Cette forme d'enrichissement des spécifications par l'étude des données est une des propositions de [Sheeren, 2005]. Une extension intéressante de notre modèle serait donc de permettre l'utilisation de contraintes métriques avec des critères flous, tels que « faible », « élevé » ou « proche de », plutôt que de simples contraintes de relation « autres » : ceci permettrait à un logiciel de retrouver facilement les contraintes qu'il peut tenter de préciser davantage par analyse des données.

Notons que même pour les critères définissant un seuil, la fouille de données est intéressante car elle permet d'estimer la tolérance admise autour de ce seuil.

Voyons maitenant les attributs qui vont intervenir dans le découpage en tronçons. Il s'agit de [fictif], [régime des eaux] et [position par rapport au sol]. Le premier de ces attributs indique, comme pour la BDCarto, la superposition avec une surface d'eau ; le critère pour la représentation par une surface est que la LARGEUR soit supérieure à 7,5 m. Le second correspond à la propriété TEMPORAIRE. Le troisième n'est pas représentable dans notre modèle actuel, en effet sa définition est la suivante :

Donne le niveau de l'objet par rapport à la surface du sol (valeur négative pour un objet souterrain, nulle pour un objet au sol et positive pour un objet en sursol). Si l'objet en sursol passe au dessus d'autres objets en sursol, sa valeur « position par rapport au sol » est égale à « 1 plus le nombre d'objets intercalés ». De la même façon, un souterrain peut prendre une valeur « position par rapport au sol » égale à « −1 moins le nombre d'objets souterrains intercalés ».

Cette définition consiste à compter des objets, ce que notre modèle ne prévoit pas actuellement. Il pourrait être intéressant de permettre la spécification, non de valeurs d'attributs mais de contraintes sur la valeur des attributs. En effet notre modèle actuel ne permet pas ici de *calculer* la valeur de l'attribut, mais il permet de déterminer si elle est positive, négative ou nulle ; or cette information pourrait déjà être utilisée par

un système pour vérifier la cohérence avec, par exemple, la BDCarto, sans qu'il soit nécessaire d'étendre le modèle pour permettre le calcul complet de l'attribut. Ceci n'est cependant pas implémenté pour le moment.

L'attribut [régime des eaux] a des changements de valeur qui correspondent exactement à ceux de la propriété TEMPORAIRE, il n'est donc pas nécessaire de le définir préalablement pour spécifier le découpage. Cependant, cet attribut est défini dans les mêmes termes pour la [surface d'eau] et pour le [tronçon de cours d'eau] : le définir comme un attribut local évite de recopier la définition deux fois. Nous définissons donc deux attributs locaux :

```
attributs :
   large = Si largeur > 7,5 "m" Alors Vrai Sinon Faux;
   regime = Si temporaire = Vrai Alors "intermittent"
      Sinon "permanent"
```

Tous les COURS D'EAU de plus de 7,5 m de large sont représentés par des [surfaces d'eau], sans seuil minimal spécifié. Le découpage peut s'exprimer ainsi :

```
découpage : Changement (large, temporaire)
   sections :
      instanciation :
         Si large = Vrai Alors Instancie surface_d_eau
            (nature = "surface d'eau",
            regime_des_eaux = regime,
            geometrie = contour)
Fin découpage
```

Le découpage en [cours d'eau nommés] ne pose pas de problème particulier, à part celui de la relation de composition déjà mentionné dans l'exemple de la BDCarto. Les critères de sélection mentionnés sont plus précis que ceux de la BDCarto.

```
découpage : Changement (toponyme)
   sections :
      instanciation :
         Si toponyme =/= "" Et temporaire = Faux
            Et Vérifie "nommé sur la carte au 1:25 000 en service"
         Alors Instancie cours_d_eau_nomme (toponyme = toponyme)
Fin découpage
```

Enfin, le découpage en [tronçons de cours d'eau] est similaire à celui de la BDCarto : intersections du réseau, changements d'attributs ou rencontre d'un obstacle représenté dans la base. Dans la BDTopo Pays, les obstacles sont représentés par des [tronçons de cours d'eau] et non par des nœuds. On ne sait pas exprimer dans notre modèle la définition de l'attribut [position par rapport au sol], mais on peut néanmoins savoir quand il change de valeur, grâce aux propriétés SURÉLEVÉ et SOUTERRAIN. Ainsi :

```
découpage :
   Changement (temporaire, artificiel, sureleve, souterrain);
   Changement > 25 "m" (large);
   // "large" correspond à l'attribut "fictif" (il y a une surface)
   Intersections;
```

```
   Rencontre accident_de_parcours
      Sélectionné comme troncon_de_cours_d_eau
   sections :
      instanciation :
         Instancie troncon_de_cours_d_eau
            (nature = "cours d'eau indifférencié",
            artificialise = artificiel,
            fictif = large,
            regime_des_eaux = regime,
            geometrie = axe)
// on ne sait pas exprimer la valeur de l'attribut
// position par rapport au sol
Fin découpage
```

Les nœuds du réseau ne sont pas représentés dans la BDTopo Pays, ceci achève donc notre procédure de représentation.

4.2.3 Structure créée à partir de ces procédures de représentation

Une fois ces procédures de représentation saisies dans notre prototype, on peut créer la structure correspondante et l'explorer avec notre interface. Afin de s'y repérer plus facilement, nous utilisons les mêmes couleurs que sur les figures 4.3 et 4.4 : bleu pour les éléments de l'ontologie, vert pour les éléments de structure de la procédure de représentation, violet pour les contraintes. Comme il y a deux bases de données, nous utilisons l'orange et le rose comme variantes du rouge.

La figure 4.9 montre quelques-uns des éléments créés par la lecture de la procédure de représentation décrite en 4.2.1 : l'objet Modelisation contient un bloc modélisation principal, qui comprend les contraintes de sélection des culs-de-sac et des parcours secondaires. Ce bloc contient également trois sous-blocs correspondant aux différents découpages. En ouvrant l'un de ces blocs découpage, on constate qu'il contient une règle d'instanciation. Cette règle d'instanciation est liée à une classe de la BDCarto, [cours d'eau], et possède une condition d'instanciation : « toponyme =/= "" ».

La figure 4.10 montre une partie de la structure des contraintes dans la même procédure de représentation : la contrainte de sélection des parcours secondaires est une contrainte complexe. Sa deuxième composante est une contrainte de relation topologique, dont le type est Adjacence, et qui est liée au type d'entité géographique ÎLE. Cette contrainte de relation contient une contrainte sur l'entité en relation qui est une contrainte sur la propriété générique SUPERFICIE.

Enfin, la figure 4.11 montre des éléments de la procédure de représentation correspondante dans la BDTopo Pays (décrite en 4.2.2), qui, nous l'avons vu, est très similaire dans sa structure à celle de la BDCarto.

4.2.4 Bilan des exemples

Globalement, sur les exemples que nous venons de présenter, nous sommes parvenu à exprimer les spécifications dans notre modèle, à l'exception de certaines définitions d'attributs et de la description des relations entre objets de la base. Les règles concernant les relations entre objets ont en effet été laissées de côté dans notre modèle actuel, en raison notamment de leur nombre relativement faible dans

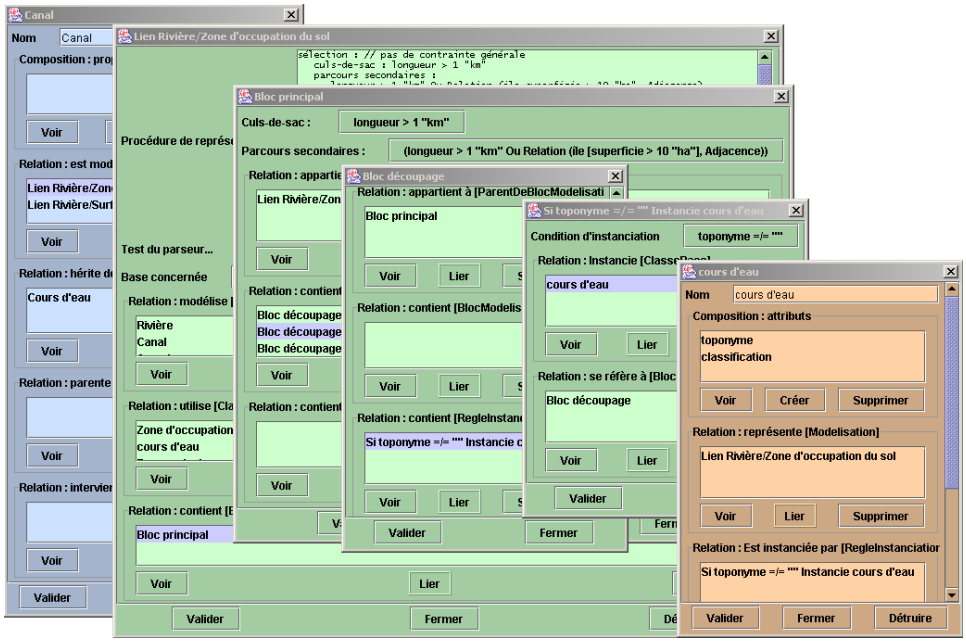


FIG. 4.9 *Capture d'écran montrant différents éléments de la procédure de représentation des* COURS D'EAU *dans la BDCarto*

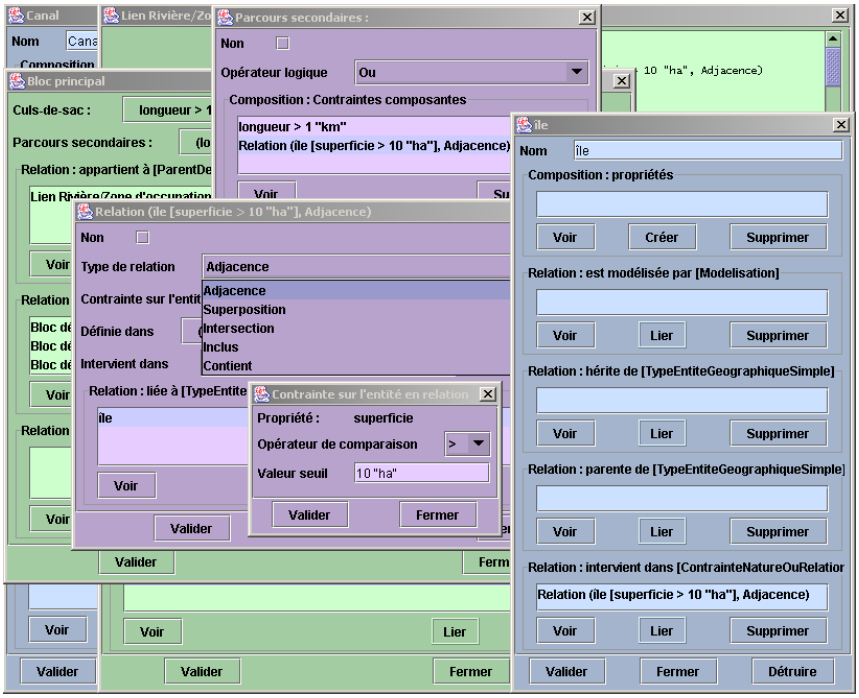


FIG. 4.10 *Capture d'écran montrant quelques-unes des contraintes définies dans la procédure de représentation des* COURS D'EAU *de la BDCarto*

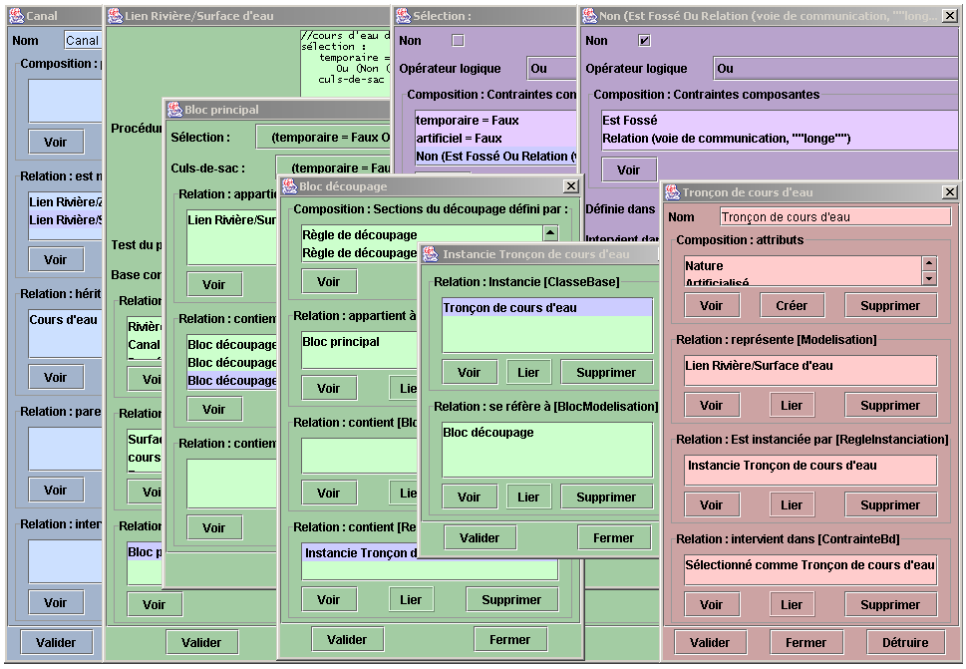


FIG. 4.11 *Capture d'écran montrant différents éléments de la procédure de représentation des* COURS D'EAU *dans la BDTopo Pays*

les schémas. Cependant ces exemples rappellent que, bien que peu nombreuses, ces relations existent. La formalisation des règles les concernant devra être étudiée afin de compléter notre modèle.

En ce qui concerne les attributs, nous avons remarqué qu'il serait intéressant, dans les cas où le modèle ne permet pas de décrire complètement la règle déterminant la valeur de l'attribut, de permettre l'expression d'informations sur cette valeur qui, elles, sont faciles à déterminer, telles que « la valeur est positive ».

Les règles de découpage telles que nous les avons définies semblent quant à elles être raisonnablement expressives : il n'a pas été très difficile d'exprimer les trois découpages différents dont les COURS D'EAU font l'objet dans chacune des deux bases. Remarquons que nous avons été amené à utiliser dans un découpage un attribut local défini dans un *autre* découpage pour exprimer le lien entre les deux représentations que constitue l'attribut [fictif].

Pour ce qui est des contraintes, enfin, leur utilisation dans ces exemples concrets nous a permis de remarquer plusieurs faits :

— on s'aperçoit qu'un choix est souvent possible entre différents types de contraintes ;

— en particulier, si l'on s'autorise à définir autant de propriétés qu'on veut sur les types d'entités géographiques, beaucoup de contraintes intrinsèques, en particulier celles qui interviennent dans les définitions d'attributs, peuvent s'exprimer comme des contraintes sur propriétés. Cependant il n'est pas utile de définir dans l'ontologie des propriétés extrêmement spécifiques qui ne seront jamais référencées que dans une seule base de données : cela ne présente

aucun avantage par rapport à une contrainte descriptive. Seules des propriétés suffisamment générales devraient être définies dans l'ontologie ; le problème est alors de déterminer où se situe le « suffisamment ». Notons que, dans nos exemples, nous avons défini les propriétés en même temps que les procédures de représentation, en fonction des besoins qui apparaissaient. Cependant le choix de ce qui sera et ne sera pas considéré comme des propriétés devrait plus logiquement être fait en amont, lors de la constitution de l'ontologie ;

— il semble intéressant de permettre la définition de types de relations dans l'ontologie, correspondant à la structure du terrain ; ces types de relations devraient alors pouvoir être utilisés dans les contraintes de relations.

Conclusion

Ce travail s'inscrit dans une action de recherche sur l'intégration de bases de données géographiques. Le processus d'intégration de bases de données comprend deux étapes principales, la mise en correspondance des schémas et celle des données. La mise en correspondance, ou appariement, des données consiste à déterminer quelles données, dans les différentes bases, représentent des informations sur le même phénomène. La mise en correspondance des schémas consiste à déterminer dans un premier temps quels éléments des schémas décrivent les mêmes types de phénomènes, donc sont susceptibles de contenir des données qui se correspondent, et dans un second temps comment ces descriptions sont reliées.

Dans le cas des bases de données géographiques, ce que représente chaque élément du schéma et la façon dont il le représente sont décrits dans les spécifications des bases de données. Ces informations sont nécessaires à la mise en correspondance des schémas. Or s'il existe des formalismes pour décrire les correspondances inter-schémas, et des outils logiciels permettant de réaliser l'intégration des schémas en s'appuyant sur ces correspondances [Sotnykova, 2003], les spécifications elles-mêmes restent à l'heure actuelle des documents textuels volumineux et complexes dont un expert doit réaliser une analyse croisée pour déterminer les correspondances.

Notre travail s'est intéressé à cette étape d'analyse des spécifications, située en amont de la mise en correspondance des schémas. Les spécifications décrivent les relations entre le contenu des bases de données et le terrain qu'elles décrivent ; ces relations sont actuellement décrites de façon informelle, et selon une structure qui dépend de la base de données considérée. Nous pensons que la définition d'un modèle formel capable de représenter ces relations de façon unifiée facilitera d'une part ce travail d'analyse des spécifications, et permettra d'autre part d'automatiser en bonne partie la détermination des correspondances entre schémas consécutive à cette analyse. Nous avons présenté dans ce mémoire des propositions pour un tel modèle, et nous avons réalisé un prototype logiciel mettant en œuvre ces propositions.

Bien que les spécifications décrivent le passage du terrain aux données, ou, plus précisément, du terrain conceptualisé au terrain nominal, leur organisation actuelle est commandée par le schéma de la base de données qu'elles décrivent. Nous avons montré que cette structure présentait deux inconvénients relativement à notre objectif : tout d'abord, elle dépend de la base considérée et contredit donc l'idée d'une représentation unifiée. Ensuite, certains choix arbitraires possibles lors de la représentation du terrain ne peuvent pas être exprimés de façon simple en s'appuyant uniquement sur le schéma de la base : les spécifications textuelles font appel pour les

décrire à des concepts qui ne sont pas des éléments du schéma. Il s'agit notamment des choix concernant le découpage d'un réseau en tronçons.

En conséquence, notre première proposition pour le modèle de représentation des spécifications est de s'affranchir des schémas des bases de données et de définir un schéma s'appuyant sur les *concepts* utilisés par les spécifications. Ce schéma ne décrit plus la structure de la *représentation* du terrain dans une base de données mais la structure du terrain lui-même, tel que les experts ayant rédigé les spécifications le perçoivent. Un tel schéma est appelé *ontologie* du domaine. Dans la mesure où les spécifications décrivent les liens entre le terrain conceptualisé par les experts, structuré par l'ontologie, et le terrain nominal de la base de données, structuré par son schéma, nous avons alors proposé de représenter formellement ces spécifications par des liens entre ontologie et schéma.

De nombreux travaux mentionnent l'utilité, lors de l'intégration de bases de données, d'utiliser une ou plusieurs ontologies et d'établir des liens entre schémas des bases et ontologies (voir 2.2.1). [Uitermark, 2001] indique que ces liens sont décrits par les spécifications mais ne s'intéresse pas à leur formalisation. Nous allons plus loin en proposant de formaliser les spécifications *en tant que* liens entre ontologies et schémas.

Notre seconde proposition est un langage formel permettant de décrire les différentes règles de représentation intervenant dans ces liens, regroupées dans des *procédures de représentation*. Une procédure de représentation relie un ou plusieurs concepts de l'ontologie à une ou plusieurs classes de la base et décrit comment les entités géographiques instances de ces concepts sont représentées par des objets instance de ces classes.

Pour définir les éléments de ce langage, nous avons recherché dans les spécifications les types de règles qui revenaient fréquemment, les critères les plus exceptionnels pouvant être laissés en langue naturelle. Nous avons regroupé les règles en différentes catégories : les plus importantes sont les règles de *sélection* et d'*instanciation*. Les règles de sélection précisent, dans une procédure de représentation, donc pour un ensemble de types d'entités géographiques donné, comment on filtre l'ensemble des entités géographiques appartenant à ces types pour ne retenir que celles qui sont suffisamment importantes pour la représentation considérée. L'étude des spécifications nous a conduit à remarquer que dans la grande majorité des cas, ces règles peuvent être représentées comme un ensemble de contraintes intrinsèques et de contraintes contextuelles que doit vérifier chaque entité individuelle. Le principal cas où une telle représentation pose problème est celui des entités formant réseau, telles que les cours d'eau ou les routes, car la notion d'« entité individuelle » n'est alors pas très bien définie. Nous avons donc étudié les critères de sélection concernant les réseaux et constaté que, dans toutes les spécifications que nous avons consultées, on pouvait pour l'essentiel en rendre compte sous forme de trois contraintes concernant différentes parties du réseau : une contrainte générale, une contrainte sur les parcours secondaires et une contrainte sur les culs-de-sac, étant entendu que la détermination de ce que sont les parcours secondaires et les culs-de-sac peut inclure une part plus ou moins grande d'arbitraire, mais doit être cohérente sur l'ensemble du réseau. Nous n'avons pas remarqué, à la lecture des spécifications, d'autre cas flagrant où la contrainte sur les entités individuelles n'était pas suffisante pour exprimer les cri-

tères de sélection, hormis les cas où des entités sont agrégées avant sélection, ce que nous traitons en faisant porter une contrainte sur les agrégats comme s'ils étaient des entités individuelles. Cependant il est possible que certains critères de sélection ne s'expriment ni comme une sélection individuelle ni comme un filtrage de réseau, notamment dans un thème comme l'occupation du sol, que nous avons peu étudié.

Les contraintes elles-mêmes ont été classifiées en différents types et sous-types correspondant à ce qu'on rencontre le plus fréquemment dans les spécifications. Un grand nombre de contraintes ne peuvent toujours pas être formalisées par notre modèle et se trouveront représentées comme « contraintes descriptives » ou « autres contraintes de relation », mais remarquons que beaucoup de ces contraintes sont de toute façon très spécifiques et ont peu de chances d'être comparables avec des critères spécifiés dans une autre base de données. La formalisation des contraintes mériterait cependant, à notre avis, d'être poussée plus avant, notamment celle des contraintes de relation.

Les règles d'instanciation décrivent, une fois qu'on a déterminé l'ensemble des entités représentées, *comment* ces entités sont représentées. Plus précisément, elles indiquent quelles classes de la base sont instanciées pour représenter chaque entité et comment sont déterminées les valeurs d'attributs des objets créés ; nous considérons la représentation géométrique de l'objet comme un attribut. Dans beaucoup de cas, les valeurs d'attributs sont déterminées à partir de *propriétés* de l'entité correspondante. Nous avons proposé de définir ces propriétés dans l'ontologie, puisqu'elles sont des éléments de la conceptualisation du monde. Dans presque tous les autres cas, l'attribut prend différentes valeurs suivant que l'entité vérifie telle ou telle condition ; nous avons introduit pour exprimer cela des expressions conditionnelles. Il existe, marginalement, quelques attributs qui n'entrent dans aucune de ces deux catégories ; ces attributs pourraient être décrits par des définitions textuelles, dans la mesure où ils sont peu nombreux. L'essentiel du problème de la formalisation des définitions d'attributs concerne à notre avis les attributs définis à partir de propriétés, et plus précisément la description de la façon dont sont déterminées les valeurs des propriétés.

À ces règles de sélection et d'instanciation, nous avons ajouté deux types de règles permettant de décrire les cas où l'on n'a pas une représentation par entité, mais des représentations agrégées de plusieurs entités ou des représentations de portions d'entités : les règles d'agrégation et de découpage. Les règles d'agrégation sont décrites sous forme de contraintes que doit vérifier un ensemble d'entités pour être transformé en un agrégat ; ces contraintes portent notamment sur les similitudes entre les entités composant l'ensemble et la distance entre elles. Elles peuvent être utilisées pour de nombreux types d'entités, dès que la représentation n'est pas très détaillée ; l'agrégation est en effet une forme simple de *généralisation* au sens cartographique.

Les règles de découpage sont de nature différente et n'ont à notre connaissance d'utilité que pour la représentation des réseaux. Une règle de découpage consiste à indiquer un ensemble de lieux où l'entité va être coupée ; ces lieux sont typiquement les intersections du réseau sélectionné, des endroits où la valeur d'un attribut change, et des endroits où l'entité rencontre une autre entité.

La très large majorité du texte des spécifications de contenu qui nous ont servi d'exemple, celles des bases de données vecteur de l'IGN, peut être classé dans un de

ces quatre types de règles. L'exception la plus notable concerne les relations entre classes de la base définies dans le schéma, qui, quoique peu nombreuses pour la taille dudit schéma, existent néanmoins, et que nous n'avons pas traitées.

Utilisations

Comme nous l'avons mentionné au début de cette conclusion, notre modèle vise tout d'abord à faciliter la détermination des correspondances entre schémas de bases de données. Mais les spécifications sont également utiles pour d'autres étapes du processus d'intégration de bases de données géographiques, notamment pour la vérification de la cohérence entre bases lors de l'étape finale d'intégration des données. David Sheeren [Sheeren, 2005], qui a par ailleurs collaboré à la classification des contraintes intervenant dans les spécifications, propose une méthodologie d'évaluation de la cohérence faisant notamment intervenir des règles déduites des spécifications. Il pose un certain nombre de questions auxquelles l'analyse des spécifications doit répondre afin d'en extraire les règles utiles à l'évaluation de la cohérence : « pour chaque base : existe-t-il plusieurs classes représentant le même phénomène ? [...] Entre les bases : pour un phénomène représenté dans une base, existe-t-il une représentation dans l'autre base ? [...] ». L'utilisation de notre modèle permet de répondre directement à la plupart de ces questions. Les règles elles-mêmes pourraient être, au moins dans certains cas, déduites automatiquement des spécifications exprimées dans notre modèle, en raisonnant sur les contraintes. Par exemple, dans deux bases de données de l'IGN, la BDCarto et Géoroute, les ronds-points ont plusieurs représentations différentes possibles, suivant leur diamètre. Un rond-point donné possède une représentation dans chaque base ; ces représentations sont dites incohérentes s'il ne peut exister de rond-point réel tel que les deux représentations vérifient toutes deux leurs spécifications. Les règles déduites des spécifications doivent permettre à un système-expert de déterminer si les représentations sont incohérentes ou non. Dans cet exemple, le problème peut être reformulé selon les termes de notre modèle de la façon suivante : il s'agit de déterminer si les règles d'instanciation qui mènent du type d'entité géographique ROND-POINT à la première représentation et celles qui mènent de ce même type d'entité à la seconde représentation possèdent des conditions d'instanciation compatibles. Si toutes les conditions d'instanciation sont bien des contraintes simples sur la propriété DIAMÈTRE, les règles pour le système-expert peuvent être générées automatiquement. Dans des cas plus complexes incluant des contraintes non formalisées, les règles pourraient devoir être écrites manuellement, mais la formalisation préalable des spécifications dans notre modèle faciliterait néanmoins leur détermination.

La description formalisée des spécifications peut également être exploitée, en tant que métadonnées, par des logiciels souhaitant mesurer l'adéquation d'une base de données à un besoin, par exemple pour déterminer la base la plus adaptée à la requête d'un utilisateur. Supposons par exemple que l'utilisateur souhaite se procurer des données indiquant quels sont les cours d'eau navigables dans une zone qui l'intéresse. Le système peut disposer d'un catalogue de bases de données couvrant la zone considérée, et utiliser les spécifications formalisées pour déterminer lesquelles de ces bases de données peuvent convenir. Un premier tri pourra consister à ne choisir que les bases de données pour lesquelles la propriété NAVIGABLE du COURS D'EAU intervient dans au moins une règle de représentation. Ce tri pourra ensuite

être affiné en regardant dans quel type de règle de représentation cette propriété intervient : par exemple, si elle n'intervient que dans les critères de sélection et qu'elle n'est pas une condition nécessaire d'instanciation (c'est-à-dire si elle apparaît dans une contrainte complexe du type navigable = Vrai Ou largeur > 5 "m"), la base de données ne répond en fait pas bien au besoin. En revanche, si la propriété NAVIGABLE intervient dans une règle d'affectation d'attribut, la probabilité pour que la base convienne est plus élevée ; d'autres critères peuvent ensuite entrer en jeu, selon le besoin précis.

Plus simplement, pour une seule base de données, les spécifications formelles pourraient être utilisées par des outils d'aide à l'utilisation des données, pour guider un utilisateur vers les éléments du schéma qui l'intéressent, ou pour reformuler dans les termes du schéma des requêtes formulées dans les termes de l'ontologie, a priori plus familiers à l'utilisateur.

Une autre utilité de notre modèle est documentaire. En effet, les spécifications actuelles ont été écrites par des experts et, dans une bonne mesure, également pour des experts. De ce fait, certains éléments importants mais banals pour les experts ont pu être oubliés ; des ambiguïtés peuvent être présentes. L'expression des mêmes informations sous une forme différente peut mettre en valeur certains aspects qui ne l'étaient pas avant et permettre de déterminer les points ambigus ou qui manquent de précision, et ainsi mieux connaître les données.

Extensions souhaitables de notre modèle

Parmi les nombreux types de règles de représentation que notre modèle ne permet pas encore de formaliser, et représente donc sous forme de chaînes de caractères en langue naturelle, deux points nous semblent particulièrement importants à étudier :

— Les « autres contraintes de relation ». Il ne sera vraisemblablement pas possible de classifier exhaustivement toutes les relations entre entités envisageables par les spécifications pour supprimer complètement ce type de contrainte. Mais certaines relations spatiales sémantiques, non descriptibles simplement en termes métriques ou topologiques, reviennent particulièrement souvent dans les spécifications et mériteraient d'être classées à part. L'exemple typique est la relation « mener à » ; d'autres pourraient vraisemblablement être trouvées en étudiant les spécifications, probablement « longer », peut-être « contourner ». Une autre piste intéressante pour les relations est l'utilisation de relations structurelles, définies dans l'ontologie, telles que « être affluent de ».

— Les « précisions » sur la détermination des valeurs des propriétés. Ce point est délicat car toutes les précisions que nous pensons nécessaires ne sont pas toujours données explicitement dans les spécifications de contenu détaillées. Certaines peuvent être complètement implicites, d'autres mentionnées dans les spécifications générales ou de qualité, notamment tout ce qui est relatif à la résolution et à la précision des données. Rappelons que notre modèle considère comme des propriétés de l'entité ses différentes caractéristiques géométriques servant à instancier les objets qui la représentent, par exemple son centre, son sommet, son contour. Les précisions sur la détermination des valeurs de propriétés comprennent donc notamment tous les critères de modélisation géométrique. Ainsi le fait que le contour d'un bâtiment soit « saisi au niveau du

toit » entre dans cette catégorie, de même que le fait que les cours intérieures ne soient pas prises en compte en-dessous d'une certaine taille. Toutes ces précisions peuvent être des éléments très importants d'explication de différences de représentation, et leur formalisation devrait être étudiée.

D'autre part, il faudrait pour que notre modèle soit complet lui ajouter des règles permettant de décrire l'instanciation des relations du schéma de la base.

Utilisation plus approfondie de l'ontologie

Nous nous sommes dans cette thèse concentré essentiellement sur les liens entre ontologie et schéma, et peu sur l'ontologie elle-même. Nous avons donc utilisé dans nos exemples une ontologie minimaliste. La partie ontologie du modèle gagnerait cependant à être étudiée davantage. Tout d'abord, sa constitution : nous avons utilisé pour nos exemples une ontologie constituée manuellement à partir des spécifications. Il pourrait être intéressant, en conservant l'idée de constituer l'ontologie à partir de mots-clefs des spécifications, de recourir à des outils de traitement automatique de la langue naturelle pour automatiser, au moins partiellement, ce processus.

Ensuite, sa structure : l'ajout dans l'ontologie de relations entre concepts d'une part, et de connaissances sur les concepts d'autre part, pourrait à notre avis enrichir utilement notre modèle. Deux points nous semblent notamment justifier l'utilisation de relations entre concepts :

— Certains concepts géographiques correspondent à des groupes d'entités. On peut avoir une représentation du groupe d'une part et une représentation des entités individuelles d'autre part; de notre point de vue, la représentation des entités individuelles ne constitue pas directement une représentation du groupe, et nous ne pensons pas souhaitable de relier par exemple directement la classe [bâtiment] au type d'entité AGGLOMÉRATION par une procédure de représentation. Cependant le lien entre les deux représentations existe bien, et ce lien peut être exploité : [Sheeren, 2005] a étudié avec succès la cohérence entre des représentations de bâtiments individuels et de zones de bâti, en s'appuyant sur les spécifications pour construire des zones à partir des bâtiments individuels. Le meilleur endroit de notre modèle pour indiquer l'existence de ce lien et donc suggérer que les représentations sont corrélées nous semble être l'ontologie : structurellement, sur le terrain, les ZONES DE BÂTI sont composées de BÂTIMENTS.

— Plus généralement, l'étude des liens structurels entre types d'entités géographiques nous semble importante. Ces liens conduisent à pouvoir représenter plusieurs fois la même spécification, de plusieurs points de vue différents. En effet, nous avons mentionné dans un de nos exemples, en tant que critère de sélection des parcours secondaires des COURS D'EAU dans la BDCarto, le critère suivant : le parcours secondaire doit délimiter une île d'une superficie supérieure à 10 hectares. Mais ce critère peut être considéré comme un critère de sélection des ÎLES, et la classe [tronçon hydrographique] comme représentant, entre autres, les ÎLES. De la même façon, la classe [laisse] possède un critère de sélection sur les ÎLES. On pourrait également citer le critère suivant sur les bâtiments dans la BDTopo Pays, intervenant cette fois dans la *modélisation* géométrique et non dans la *sélection* : « seules les cours intérieures de plus de 10 m

de large sont représentées par un trou dans la surface bâtie ». La définition dans l'ontologie de relations telles que « un COURS D'EAU délimite une ÎLE » semble utile pour représenter ces différentes façons de voir une même spécification de sélection sans stocker cette spécification de façon redondante.

L'introduction de connaissances sur les concepts, quant à elle, pourrait permettre de raisonner plus facilement sur les spécifications formalisées. Par exemple, des connaissances sur les valeurs que peuvent et ne peuvent pas posséder les propriétés d'un certain type de concept permettraient de simplifier automatiquement les procédures de représentation quand on s'intéresse à un type d'entité particulier. Ainsi, supposons qu'on dispose de connaissances sur la taille que peut avoir une MARE. Ces connaissances pourraient permettre de déterminer qu'un PLAN D'EAU de plus de 4 hectares ne peut jamais être considéré comme une MARE, et d'en déduire que les MARES ne sont jamais concernées par la procédure de représentation des PLANS D'EAU dans la BDCarto, dont la contrainte de sélection demande une superficie supérieure à 4 hectares.

Annexe A

Grammaire BNF du langage de description des procédures de représentation

Nous utilisons la variante de BNF suivante : les symboles terminaux sont entre guillemets (") tandis que les non-terminaux sont écrits sans guillemets. La barre verticale indique une alternative, les crochets indiquent une option (élément présent zéro ou une fois), et les accolades indiquent une répétition (élément présent un nombre quelconque de fois).

Dans notre implémentation de cette grammaire, les non-terminaux identifiant_attribut, identifiant_type_eg et identifiant_classe_base sont tous définis lexicalement comme une suite quelconque de lettres minuscules non accentuées et de blancs soulignés. Le non-terminal « chaîne » est défini comme n'importe quelle suite de caractères (hors guillemets) entre guillemets, et le non-terminal « nombre » est défini comme une suite quelconque de chiffres avec éventuellement une virgule. Par ailleurs, les commentaires sont introduits par la séquence « // » et terminés par une fin de ligne.

```
procédure_représentation ::= { élément_représentation }

élément_représentation ::=
   sélection | attributs | découpage | agrégation | instanciation

sélection ::= "sélection :"
   [ contrainte ]
   [ "culs-de-sac :" contrainte ]
   [ "parcours secondaires :" contrainte ]

attributs ::= "attributs :" définition_attribut { ";" définition_attribut }
définition_attribut ::= identifiant_attribut "=" expression

découpage ::= "découpage :" règle_découpage { ";" règle_découpage }
   [ "sections :" { élément_représentation } ]
   [ "limites :" { élément_représentation } ]
   "Fin découpage"
règle_découpage ::=
   "Intersections"
   | "Changement" [ ">" valeur_numérique ] [ "[" précisions "]" ]
      "(" identifiant_attribut { "," identifiant_attribut } ")"
   | "Rencontre" identifiant_type_eg [ contrainte ]

agrégation ::=
   "agrégation :" contrainte_agrégation { élément_représentation } "Fin agrégation"
contrainte_agrégation ::=
   ["Non"] contrainte_agr_élém { ("Et" | "Ou") contrainte_agr_élém }
contrainte_agr_élém ::=
   "(" contrainte_agrégation ")"
   | "Chacun" contrainte_élémentaire
   | "Ensemble" contrainte_élémentaire
   | relation_métrique
   | "Même" identifiant_attribut
   | "Différence" identifiant_attribut opérateur_comparaison valeur_numérique

instanciation ::= "instanciation :" liste_règles_instanciation
règle_instanciation ::=
   Instancie identifiant_classe_base
      "(" instanciation_attribut { "," instanciation_attribut } ")"
   | instanciation_conditionnelle
   | "(" liste_règles_instanciation ")"
instanciation_attribut ::= identifiant_attr_bd "=" expression
liste_règles_instanciation := règle_instanciation { ";" règle_instanciation }
instanciation_conditionnelle ::=
   "Si" contrainte "Alors" règle_instanciation [ "Sinon" règle_instanciation ]


contrainte ::=
   ["Non"] contrainte_élémentaire { ("Et" | "Ou") contrainte_élémentaire }
contrainte_élémentaire ::=
   "(" contrainte ")"
   | contrainte_nature
   | contrainte_sur_propriété_simple
   | contrainte_descriptive
   | contrainte_relation
   | contrainte_bd
   | contrainte_sur_propriété_complexe
```

```
contrainte_nature ::= "Est" identifiant_type_eg
contrainte_sur_propriété_simple ::=
   identifiant_attribut [ "(" précisions ")" ] opérateur_comparaison valeur_littérale
opérateur_comparaison ::= ">" | "<" | "=" | "=/="
contrainte_descriptive ::= "Vérifie" chaîne
contrainte_relation ::=
   "Relation (" identifiant_type_eg [contrainte] "," type_relation ")"
contrainte_bd ::= "Sélectionné comme" identifiant_classe_base
contrainte_sur_propriété_complexe ::= identifiant_attribut"."contrainte_élémentaire

type_relation ::= relation_topologique | relation_métrique | relation_autre
relation_topologique ::=
   "Adjacence" | "Superposition" | "Intersection" | "Inclus" | "Contient"
relation_métrique ::= critère_métrique opérateur_comparaison valeur_littérale
critère_métrique ::= "Distance" | "Dénivelée"
relation_autre ::= chaîne


expression ::=
   identifiant_attribut [ "(" précisions ")" ]
   | expression_conditionnelle
   | "(" expression ")"
   | valeur_littérale
expression_conditionnelle ::= "Si" contrainte "Alors" expression "Sinon" expression

valeur_littérale ::= chaîne | valeur_numérique | valeur_booléenne
valeur_numérique ::= nombre [unité]
valeur_booléenne ::= "Vrai" | "Faux"

unité ::= chaîne
précisions ::= chaîne
```

Annexe B

Glossaire

échelle

rapport entre les distances réelles et les distances apparentes sur une carte. Dans le cas d'une base de données géographique, les données n'ont pas de représentation graphique fixe ; les systèmes d'information géographiques permettent généralement un affichage à n'importe quelle échelle. Cependant, comme les données ont une représentation géométrique plus ou moins *détaillée*, elles ne sont adaptées qu'à une certaine plage d'échelles. On préfère cependant parler, dans le cas des bases de données, de *niveau de détail géométrique*. 8

entité géographique

élément ou ensemble d'éléments du terrain que l'on reconnaît comme formant un individu. Suivant le point de vue adopté, il existe plusieurs façons de décomposer le terrain en entités géographiques. Les entités géographiques peuvent être classées en types, correspondant à des concepts géographiques, tels que « source », « vallée », « océan », etc. 8

fédération de bases de données

architecture d'intégration de bases de données où les bases de données sources conservent leur autonomie et peuvent être utilisées comme avant l'intégration. 20

intégration de données

constitution d'un tout cohérent à partir de données provenant de plusieurs sources différentes et hétérogènes. On parle plus spécifiquement d'intégration de *bases* de données pour désigner la constitution d'une base de données unique et cohérente à partir de plusieurs bases sources. La base de données intégrée ne contient pas nécessairement physiquement les données, elle peut être virtuelle, c'est-à-dire contenir uniquement des informations permettant de retrouver les données. C'est le cas dans l'architecture de *fédération*. 18

interopérabilité

capacité des systèmes informatiques à s'échanger des données de façon cohérente. 18

maillé (ou raster)

mode de représentation des données géographiques utilisant un maillage régulier du terrain avec des informations associées à chaque maille. Les données brutes obtenues par photographie, radar ou d'autres appareils de mesure sont souvent dans ce mode. Par exemple, une image numérique est composée de pixels possédant chacun une valeur radiométrique ; cette grille de pixels correspond à un certain maillage du terrain. Le mode de représentation maillé s'oppose au mode *vecteur*.

médiateur

composant logiciel disposant de connaissances sur diverses sources de données et capable d'interpréter ces données pour restituer une information cohérente. 18

modèle de données

ensemble de notions théoriques permettant la description de bases de données. La description de la structure d'*une* base de données, selon un certain modèle de données, est son *schéma*. 18

multi-échelles

ce terme est généralement employé pour parler de bases de données géographiques gérant différentes représentations, correspondant à différents *niveaux de détail*, d'une même entité. Le terme d'échelle est employé comme synonyme de niveau de détail. Il s'agit d'un cas particulier des bases de données *multi-représentations*. 25

multi-représentations

Les bases de données multi-représentations, ou à représentations multiples, sont des bases pouvant stocker simultanément plusieurs représentations différentes d'une même entité (en conservant l'information qu'il s'agit bien de la même entité). De nombreux travaux ont été réalisés à ce sujet. 25

niveau de détail

Selon [Ruas et Bianchin, 2002], on peut parler pour une base de données géographique d'un niveau de détail géométrique et d'un niveau de détail sémantique. Le niveau de détail sémantique correspond à la finesse de la classification des entités, aux critères de *sélection* déterminant quelles seront les entités représentées, à la quantité d'information stockée pour chaque entité et à la précision de cette information. Le niveau de détail géométrique correspond à la précision de la description géométrique des entités sélectionnées. On parle souvent de *résolution*, mais dans le cas des bases de données vectorielles ce terme est ambigu. 10

raster : voir maillé

résolution

taille du plus petit détail représentable dans une base de données : il s'agit d'un des paramètres permettant de décrire le niveau de détail géométrique d'une base de données. Ce terme s'applique surtout aux bases de données en mode maillé, où il correspond à la taille de la maille, mais est parfois employé pour les bases de données vectorielles. 26

schéma

Le schéma d'une base de données est la description de sa structure. On distingue plusieurs niveaux de schémas : schéma physique, schéma logique, schéma conceptuel. Le seul niveau auquel nous nous intéressons dans ce mémoire est le niveau conceptuel, où la structure est décrite de façon indépendante du système. 18

sélection

Dans le domaine des bases de données géographiques, le terme de sélection désigne le choix des entités géographiques à représenter dans une base donnée ; les entités non sélectionnées sont donc absentes de la base. Cette sélection se fait sur des critères mesurant l'importance des entités relativement aux utilisations prévues de la base. 11

terrain nominal

ce que contiendrait une base de données en l'absence de toute erreur. Le terrain nominal est utilisé dans l'évaluation de la qualité, en le comparant au contenu effectif de la base de données. [David et Fasquel, 1997, p. 8] en donne la définition suivante : « image de l'univers, à une date donnée, à travers le filtre défini par la spécification de produit. » 38

terrain conceptualisé

ensemble des *entités géographiques* reconnaissables par l'esprit humain sur le terrain. Nous considérons dans ce travail le terrain conceptualisé comme le point de départ de la représentation, sur une carte ou dans une base de données, du terrain. En effet le terrain physique réel est trop complexe pour être appréhendé sans une étape d'abstraction et de conceptualisation, qui nous permet par exemple de voir un pré ou un champ plutôt que « de nombreuses plantes » ou « beaucoup de vert ». Lorsqu'on représente ou décrit le terrain à l'usage d'autres personnes, on décrit directement cette abstraction. 47

topographique (base de données)

base de données s'attachant à décrire les éléments constitutifs du paysage. Les données topographiques sont relativement indépendantes des applications visées ; elles servent souvent de référentiel pour l'utilisation de données plus thématiques, afin de donner des points de repère. Par données thématiques, nous entendons des données plus spécialisées, telles que des données géologiques ou de population. 9

vecteur ou vectorielle (base de données géographique)

base de données décrivant et localisant des objets à l'aide de primitives géométriques : points localisés par des coordonnées, lignes polygonales décrites par des suites de points, surfaces planes délimitées par des polygones, éventuellement volumes. Les bases vectorielles s'opposent aux bases en mode maillé (aussi appelé raster). Généralement, les bases vectorielles visent à représenter des entités géographiques individualisées, telles que des bâtiments ou des lacs, tandis que les bases en mode maillé servent à représenter des phénomènes continus, comme l'altitude ou la nature du sol ou de la végétation. Cependant cela n'est pas systématique, ainsi le relief peut être représenté par des courbes

de niveau, c'est-à-dire des lignes ; les types de végétation peuvent être représentés par des zones délimitées par des polygones, etc. Inversement, il est possible de représenter sous forme de pixels une carte qui comporte des symboles correspondant à des entités individuelles. En conséquence, on préfère souvent parler de bases de données à objets ou à champs continus pour indiquer le type d'information représenté plutôt que d'utiliser les termes de vecteur et maillé, qui correspondent à des modes de représentation. Les bases de données dont nous traitons dans ce mémoire sont des bases de données à objets en mode vecteur. 9

vue

en bases de données relationnelles, ensemble de tables virtuelles défini en fonction du schéma réel à l'aide de requêtes ; la vue peut être interrogée et les requêtes sont traduites par le système en requêtes sur le schéma réel. Plus généralement, on peut parler de vue pour désigner un schéma dérivé d'un autre schéma, mais hors des bases relationnelles ce terme ne désigne pas un mécanisme précis. 20

Bibliographie

Baader, Franz, Calvanese, Diego, McGuinness, Deborah, Nardi, Daniele et Patel-Schneider, Peter (dir.) *The description logic handbook : Theory, implementation and applications.* Cambridge : Cambridge University Press, 2003.

Backus, J. W., Wegstein, J. H. *et al.* Revised report on the algorithm language ALGOL 60. *Communications of the ACM*, 1963, vol. 6, nº 1, p. 1-17.

Badard, Thierry. *Propagation des mises à jour dans les bases de données géographiques multi-représentations par analyse des changements géographiques.* Thèse : Informatique : Université de Marne-la-Vallée : 2000.

Badard, Thierry et Lemarié, Cécile. Propagating updates between geographic databases with different scales. Chap. 10 de : Atkinson, Peter (dir.) *GIS and GeoComputation.* Londres : Taylor and Francis, 1999. (Innovations in GIS; 7).

Badard, Thierry et Lemarié, Cécile. Associer des données : l'appariement. Chap. 9 de : Ruas, Anne (dir.) *Généralisation et représentation multiple.* Paris : Hermès Science, 2002. (Information géographique et aménagement du territoire/Géomatique). P. 163-183.

Balley, Sandrine. Aide à l'acquisition de données géographiques sur mesure. In : *Proceedings of international conference on spatial analysis and geomatics (SAGEO'2005)*, Avignon, 2005.

Balley, Sandrine, Parent, Christine et Spaccapietra, Stefano. Modeling geographic data with multiple representations. *International journal of geographical information science*, 2004, vol. 18, nº 4, p. 329-354.

Bel Hadj Ali, Atef. *Qualité géométrique des entités géographiques surfaciques, application à l'appariement et définition d'une typologie des écarts géométriques.* Thèse : Université de Marne-la-Vallée : 2001.

Car, Adrijana et Frank, Andrew. General principles of hierarchical spatial reasoning : The case of wayfinding. In : Waugh, Thomas C. et Healey, Richard G. (dir.) *Proceedings of the sixth inernational symposium on spatial data handling (SDH'94)*, Édimbourg, 1994. Vol. 2, p. 646-664.

Chawathe, Sudarshan, Garcia-Molina, Hector, Hammer, Joachim, Ireland, Kelly, Papakonstantinou, Yannis, Ullman, Jeffrey D. et Widom, Jennifer. The TSIMMIS project : Integration of heterogenous information sources. In : *16th meeting of the information processing society of japan*, Tokyo, 1994. P. 7-18.

Claramunt, Christophe. Le concept de vue spatiale. Chap. 3 de : Maingue-naud, Michel (dir.) *Langages pour les SIG : Conception, développement et IHM*. Paris : Hermès Science, 2002. (Information géographique et aménagement du territoire/Géomatique). P. 49-67.

Cody, W. F., Haas, L. M., Niblack, W., Arya, M., Carey, M. J., Fagin, R., Flickner, M., Lee, D., Petkovic, D., Schwarz, P. M., Thomas, J., Roth, M. T., Williams, J. H. et Wimmers, E. L. Querying multimedia data from multiple repositories by content : the Garlic project. In : *Third working conference on visual database systems (VDB'95)*, Lausanne, 1995. URL : <http://www.almaden.ibm.com/cs/garlic/>.

Coulondre, Stéphane. *Samovar : un modèle pour les objets persistants avec rôle*. Thèse : Informatique : Université de Montpellier 2 : 2000.

Cullot, Nadine, Parent, Christine, Spaccapietra, Stefano et Vangenot, Christelle. Des ontologies pour données géographiques. *Revue internationale de géomatique*, 2003, vol. 13, n° 3 : les SIG sur le web, p. 285-306.

David, Benoît et Fasquel, Pascal. *Qualité d'une base de données géographique : concepts et terminologie*. Bulletin d'information de l'IGN n° 67, Institut géographique national, Saint-Mandé, 1997.

Devogele, Thomas. *Processus d'intégration et d'appariement de bases de données géographiques : application à une base de données routières multi-échelles*. Thèse : Informatique : Université de Versailles : 1997.

Devogele, Thomas, Badard, Thierry et Libourel, Thérèse. La problématique de la représentation multiple. Chap. 3 de : Ruas, Anne (dir.) *Généralisation et représentation multiple*. Paris : Hermès Science, 2002. (Information géographique et aménagement du territoire/Géomatique). P. 55-74.

Devogele, Thomas, Parent, Christine et Spaccapietra, Stefano. On spatial database integration. *International journal of geographical information science*, 1998, vol. 12, n° 4, p. 335-352.

Egenhofer, M. J., Clementini, E. et di Felice, P. Evaluating inconsistencies among multiple representations. In : Waugh, Thomas C. et Healey, Richard G. (dir.) *Proceedings of the sixth inernational symposium on spatial data handling (SDH'94)*, Édimbourg, 1994. Vol. 2, p. 901-920.

Egenhofer, Max J. et Herring, John R. A mathematical framework for the definition of topological relationships. In : Brassel, K. et Kishimoto, H. (dir.) *Proceedings of the fourth international symposium on spatial data handling*, Zürich, 1990. Vol. 2, p. 803-813.

Fonseca, Frederico, Davis, Clodoveu et Câmara, Gilberto. Bridging ontologies and conceptual schemas in geographic information integration. *Geoinformatica*, 2003, vol. 7, n° 4, p. 355-378.

Friis-Christensen, Anders. *Issues in the conceptual modeling of geographic data*. Ph. d. diss. : Computer science : Aalborg University (Danemark) : 2003.

GARCIA-MOLINA, Hector, PAPAKONSTANTINOU, Yannis, QUASS, Dallan, RAJARAMAN, Anand, SAGIV, Yehoshua, ULLMAN, Jeffrey, VASSALOS, Vasilis et WIDOM, Jennifer. The TSIMMIS approach to mediation : Data models and languages. *Journal of intelligent information systems (JIIS)*, 1997, vol. 8, n° 2.

GARDARIN, Georges. *Bases de données : objet et relationnel.* Paris : Eyrolles, 1999.

GRUBER, Thomas R. Toward principles for the design of ontologies used for knowledge sharing. In : GUARINO, Nicola et POLI, Roberto (dir.) *Formal ontology in conceptual analysis and knowledge representation.* Dordrecht : Kluwer academic, 1993.

GUARINO, Nicola. Formal ontology and information systems. In : GUARINO, Nicola (dir.) *Formal ontology in information systems : proceedings of FOIS'98, Trento, Italy, 6-8 June 1998.* Amsterdam : IOS Press, 1998. P. 3-15.

GUARINO, Nicola et GIARETTA, Pierdaniele. Ontologies and knowledge bases : Towards a terminological clarification. In : MARS, N. J. I. (dir.) *Towards very large knowledge bases.* Amsterdam : IOS Press, 1995.

HAARSLEV, Volker, LUTZ, Carsten et MÖLLER, Ralf. Foundations of spatioterminological reasoning with description logics. In : COHN, A. G., SCHUBERT, L. K. et SHAPIRO, S. C. (dir.) *Principles of knowledge representation and reasoning : proceedings of the sixth international conference (KR'98).* San Francisco : Morgan-Kaufmann, 1998. P. 112-123.

HAKIMPOUR, Farshad et GEPPERT, Andreas. Global schema generation using formal ontologies. In : SPACCAPIETRA, Stefano, MARCH, Salvatore T. et KAMBAYASHI, Yahiko (dir.) *Conceptual modeling — ER 2002 : 21st international conference on conceptual modeling, Tampere, Finland, october 7-11, 2002, proceedings.* Berlin : Springer, 2002. (Lecture Notes in Computer Science ; 2503). P. 307-321.

HAKIMPOUR, Farshad et TIMPF, Sabine. Using ontologies for resolution of semantic heterogeneity in GIS. In : *Proceedings of the 4th AGILE conference on geographic information science*, Brno, 2001.

IGN, 2002. *BD TOPO Pays version 1.2 : Descriptif de contenu.* Institut géographique national, Saint-Mandé, édition 1.0, 2002. URL : <http://www.ign.fr/telechargement/MPro/produit/BD_TOPO/JT_Agglo/DC_BDTOPOPays_1_2.pdf>.

IGN, 2004. *BD CARTO : Descriptif de contenu.* Institut géographique national, Saint-Mandé, édition 7, 2004.

JONES, Dean, BENCH-CAPON, Trevor et VISSER, Pepijn. Methodologies for ontology development. In : *Proceedings of IT & KNOWS conference of the 15th IFIP world computer congress*, Budapest, 1998.

KASHYAP, Vipul et SHETH, Amit. Semantic heterogeneity in global information systems : the role of metadata, context and ontologies. In : PAPAZOGLOU, M. et SCHLAGETER, G. (dir.) *Cooperative information systems : current trends and directions.* Londres : Academic press, 1996.

KILPELÄINEN, Tiina. Maintenance of multiple representation databases for topographic data. *The cartographic journal*, 2000, vol. 37, nº 2, p. 101-107.

KIRK, Thomas, LEVY, Alon Y., SAGIV, Yehoshua et SRIVASTAVA, Divesh. The information manifold. In : *AAAI spring symposium on information gathering from heterogeneous, distributed environments*, 1995.

LASSOUED, Yassine, MANOAH, Snezhana et BOUCELMA, Omar. Correspondances interschémas dans les SIG. In : *14e congrès francophone reconnaissance des formes et intelligence artificielle (RFIA)*, Toulouse, 2004. Vol. 1.

LAURINI, Robert. Raccordement géométrique de bases de données géographiques fédérées. *Ingénierie des systèmes d'information*, 1996, vol. 4, nº 3, p. 361-388.

LEVY, Alon Y. et ROUSSET, Marie-Christine. CARIN: A representation language combining Horn rules and description logics. In : *Twelveth european conference on artificial intelligence*, 1996. P. 323-327.

LRI, 2002. PICSEL : Production d'Interfaces à base de Connaissances pour des Services En Ligne. LRI, Orsay, 2002.
URL : <http://www.lri.fr/LRI/iasi/picsel/>.

NOY, Natalya F. et MCGUINNESS, Deborah L. *Ontology development 101 : A guide to creating your first ontology*. Rapport technique KSL-01-05, Stanford knowledge systems laboratory, 2001.

PARENT, C., SPACCAPIETRA, S., ZIMÁNYI, E., DONINI, P., PLAZANET, C., VANGENOT, C., ROGNON, N., POULIOT, J. et CRAUSAZ, P.-A. MADS : un modèle conceptuel pour des applications spatio-temporelles. *Revue internationale de géomatique*, 1997, vol. 7, nº 3-4.

PARENT, Christine et SPACCAPIETRA, Stefano. Database integration : The key to data interoperability. In : PAPAZOGLOU, M. P., SPACCAPIETRA, S. et TARI, Z. (dir.) *Advances in object-oriented data modeling*. Cambridge (USA) : MIT Press, 2000.

PARK, Jinsoo. *Schema integration methodology and toolkit for heterogeneous and distributed geographic databases*. Working paper, University of Minnesota, Minneapolis, 2001.

PARTRIDGE, Chris. *The role of ontology in integrating semantically heterogeneous databases*. Rapport technique 05/02, LADSEB-CNR, Padoue, 2002.

RIGAUX, Philippe. La représentation multiple dans les systèmes d'information géographiques. *Revue internationale de géomatique*, 1994, vol. 4, nº 2, p. 137-164.

RUAS, Anne. Les problématiques de l'automatisation de la généralisation. Chap. 4 de : RUAS, Anne (dir.) *Généralisation et représentation multiple*. Paris : Hermès Science, 2002. (Information géographique et aménagement du territoire/Géomatique). P. 75-90.

RUAS, Anne et BIANCHIN, Alberta. Échelle et niveau de détail. Chap. 1 de : RUAS, Anne (dir.) *Généralisation et représentation multiple*. Paris : Hermès Science, 2002. (Information géographique et aménagement du territoire/Géomatique). P. 25-44.

SHEEREN, David. *Méthodologie d'évaluation de la cohérence inter-représentations pour l'intégration de bases de données spatiales : Une approche combinant l'utilisation de métadonnées et l'apprentissage automatique.* Thèse : Informatique : Université de Paris 6 : 2005.

SHEEREN, David, MUSTIÈRE, Sébastien et ZUCKER, Jean-Daniel. How to integrate spatial databases in a consistent way? In : BENCZUR, A., DEMETROVICS, J. et GOTTLOB, G. (dir.) *Proceedings of the 8th east-european conference on advances in databases and information systems (ADBIS'04), Budapest.* Berlin : Springer, 2004. (Lecture Notes in Computer Science ; 3255). P. 364-378.

SHETH, A. P. et LARSON, J. A. Federated database systems for managing distributed, heterogeneous and autonomous databases. *ACM computing surveys,* 1990, vol. 22, n° 3, p. 183-236.

SHETH, Amit P. Changing focus on interoperability in information systems : From system, syntax, structure to semantics. In : GOODCHILD, M. F., EGENHOFER, M. J., FEGEAS, R. et KOTTMAN, C. A. (dir.) *Interoperating geographic information systems.* Dordrecht : Kluwer academic, 1998.

SMITH, Barry et MARK, David M. Ontology and geographic kinds. In : POIKER, Thomas K. et CHRISMAN, Nicholas (dir.) *Proceedings of the eighth international symposium on spatial data handling (SDH'98).* Vancouver : International Geographical Union, Geographic Information Science Study Group, 1998. P. 308-320.

SOTNYKOVA, Anastasiya. *Design and implementation of federation of spatio-temporal databases : Methods and tools.* Final project report BFR99/057, Amber project, 2003.
URL : <http://lbd.epfl.ch/e/research/amber/>.

SUBRAHMANIAN, V. S., ADALI, Sibel, BRINK, Anne, EMERY, Ross, LU, James J., RAJPUT, Adil, ROGERS, Timothy J., ROSS, Robert et WARD, Charles. *HERMES : A heterogeneous reasoning and mediator system.* University of Maryland, 1996.
URL : <http://www.cs.umd.edu/projects/hermes/overview/paper/>.

TIMPF, Sabine. Map Cube model : a model for multi-scale data. In : POIKER, Thomas K. et CHRISMAN, Nicholas (dir.) *Proceedings of the eighth international symposium on spatial data handling (SDH'98).* Vancouver : International Geographical Union, Geographic Information Science Study Group, 1998. P. 190-201.

TIMPF, Sabine. Abstraction, levels of detail, and hierarchies in map series. In : FREKSA, Christian et MARK, David M. (dir.) *Spatial information theory : Cognitive and computational foundations of geographic information science. International conference COSIT'99, Stade, Germany, proceedings.* Berlin : Springer, 1999. (Lecture Notes in Computer Science ; 1661). P. 125-140.

UITERMARK, Harry. *Ontology-based geographic data set integration.* Proefschrift : Universiteit Twente (Pays-Bas) : 2001.

UITERMARK, Harry, VAN OOSTEROM, Peter, MARS, Nicolaas et MOLENAAR, Martien. Propagating updates : finding corresponding objects in a multi-source environment. In : POIKER, Thomas K. et CHRISMAN, Nicholas (dir.) *Proceedings of the eighth*

international symposium on spatial data handling (SDH'98). Vancouver : International Geographical Union, Geographic Information Science Study Group, 1998. P. 580-591.

VANGENOT, Christelle. *Multi-représentation dans les bases de données géographiques*. Thèse : Informatique : École polytechnique fédérale de Lausanne : 2001 ; 2430.

WACHE, H., VÖGELE, T., VISSER, U., STUCKENSCHMIDT, H., SCHUSTER, G., NEUMANN, H. et HÜBNER, S. Ontology-based integration of information : a survey of existing approaches. In : STUCKENSCHMIDT, H. (dir.) *Proceedings of the 17th international joint conference on artificial intelligence (IJCAI'01) workshop on ontologies and information sharing*, Seattle, 2001.

WIEDERHOLD, Gio. Mediators in the architecture of future information systems. *IEEE computer magazine*, 1992, vol. 25, p. 38-49.

WIEDERHOLD, Gio. Interoperation, mediation and ontologies. In : *Proceedings of the international symposium on fifth generation computer systems (FGCS'94), workshop on heterogeneous cooperative knowledge bases*. Tokyo : ICOT, 1994. Vol. W3, p. 33-48.

WIRTH, Niklaus. What can we do about the unnecessary diversity of notation for syntactic definitions ? *Communications of the ACM*, 1977, vol. 20, n° 11, p. 822-823.

ZHANG, Xin Chang. Geometric feature-based edge matching. In : ABRAHART, Robert J. (dir.) *Proceedings of the 3rd international conference on GeoComputation*. University of Bristol, 1998.

Publications

GESBERT, Nils. Recherche de concepts fédérateurs dans les bases de données géographiques. In : *Actes des sixièmes journées Cassini*, Lanvéoc, 2002. P. 365-368.

GESBERT, Nils. Formalisation des spécifications de bases de données géographiques pour une meilleure compréhension des données. In : *Actes du XXIIe congrès Informatique des Organisations et Systèmes d'Information et de Décision (INFORSID)*, Biarritz, 2004. P. 267-281.

GESBERT, Nils. Formalisation of geographical database specifications. In : *Proceedings of the 8th east-european conference on advances in databases and information systems (ADBIS'04)*, Budapest, 2004. P. 202-211.

GESBERT, Nils. Formalisation des spécifications des bases de données géographiques. *Bulletin d'information scientifique et technique de l'IGN*, 2005, n° 75, p. 81-86.

GESBERT, Nils, LIBOUREL, Thérèse et MUSTIÈRE, Sébastien. Apport des spécifications pour les modèles de bases de données géographiques. *Revue internationale de géomatique*, 2004, vol. 14, n° 2 : les ontologies spatiales, p. 239-257.

MUSTIÈRE, Sébastien, GESBERT, Nils et SHEEREN, David. A formal model for the specifications of geographical databases. In : LEVACHKINE, S., SERRA, J. et EGENHOFER, M. (dir.) *Semantic processing of spatial data, proceedings of workshop GeoPro 2003*, 2003. P. 152-159.

MUSTIÈRE, Sébastien, SHEEREN, David et GESBERT, Nils. Unification de bases de données géographiques : Recherches au laboratoire COGIT de l'IGN. *Géomatique expert*, 2004, nº 32/33, p. 50-54.

Étude de la formalisation des spécifications de bases de données géographiques en vue de leur intégration

Objectif : faciliter l'utilisation conjointe de plusieurs bases de données géographiques, et en particulier leur intégration dans un système de bases de données fédérées, grâce à une description homogène entre les différentes bases et autant que possible formelle de la sémantique précise des données, c'est-à-dire des spécifications de contenu.

Méthode : Les spécifications papier ont le défaut d'être structurées conformément aux schémas des bases de données, qui sont hétérogènes et influencés par des choix d'implémentation. Pour s'affranchir de ce problème, on fait l'hypothèse qu'on peut trouver dans les spécifications un certain nombre de termes consensuels correspondant à des concepts géographiques partagés. Ces concepts formeraient ce qu'on appelle une *ontologie* du domaine. L'idée n'est pas de créer cette ontologie de façon exhaustive mais de créer une ontologie partielle *ad hoc*, qu'on étendra au fur et à mesure qu'on rencontrera de nouveaux concepts dans les spécifications. Les spécifications sont alors représentées sous la forme de *procédures de représentation* qui à une entité géographique (instance d'un concept donné) associent une ou plusieurs représentations dans les différentes bases, en fonction de la nature et des propriétés de l'entité ; elles font donc le lien entre l'ontologie et les schémas des différentes bases.

Résultats : sur les thèmes hydrographie de la BDCarto et de la BDTopo, deux bases de données de l'IGN, il semble que l'hypothèse soit vérifiée (on arrive assez facilement à une ontologie commune). On a pu d'autre part déterminer les principaux types de règles élémentaires nécessaires à la construction des procédures de représentation. Un langage formel dont la grammaire, définie en BNF, s'appuie sur ces règles élémentaires a alors été proposé pour décrire les procédures de représentation. Enfin, un prototype logiciel muni d'un parseur pour ce langage a été réalisé pour saisir, stocker et manipuler les spécifications formelles.

Study of the formalization of geographical database specifications for their integration

Goal : make the simultaneous use of several geographical databases easier, and in particular help integrating them into a federated database system, by describing the precise data meaning in a way both homogeneous between databases and as formal as possible. This precise data meaning is contained in the databases' *content specifications* (surveying rules).

Method : The general organization of the present specifications follows that of the databases' schemata, but these schemas are heterogeneous and influenced by implementation problems. To overcome this problem, we suppose that it will be possible to find, in the specifications' text, a number of common terms referring to shared geographical concepts. All these concepts would constitute what is called a *domain ontology*. Our idea is not to create a complete ontology but rather a partial, *ad hoc* one, which would be extended to take new concepts into account as needed. The specifications would then be represented as a bundle of what we call *representation procedures*, which describe how, given a geographic entity (instance of some geographical concept), one or more representations of this entity are built up into the different databases depending on the nature and the properties of the entity. Thus these procedures describe the links between the ontology and the schemata of the databases.

Results : For the example of hydrography in two different IGN databases, BDCarto and BDTopo, our hypothesis seems confirmed : a common ontology could rather easily be defined. Concerning the representation procedures, we were able to establish the main kinds of elementary rules from which they can be constructed. To describe formally these procedures, we then defined a formal language whose grammar has been described in BNF and is based on these elementary rules. Finally, we have made a software prototype, containing a parser for this language, for entering, saving and handling the formal specifications.

Thèse de doctorat de l'université de Marne-la-Vallée
Discipline : informatique
Mots-clefs : spécifications, bases de données géographiques, SIG, multi-représentation, ontologie, intégration de bases de données, représentation des connaissances

Laboratoire COGIT, IGN, 2/4 avenue Pasteur, 94165 Saint-Mandé CEDEX