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.

David Sheeren

Université Paris 6
Laboratoire d'informatique (LIP6)
–
Institut Géographique National
Laboratoire COGIT

THESE DE DOCTORAT

Spécialité : Informatique – Intelligence Artificielle

présentée par

David Sheeren

pour l'obtention du titre de

DOCTEUR DE L'UNIVERSITE PARIS 6

METHODOLOGIE D'EVALUATION DE LA COHERENCE INTER-REPRESENTATIONS POUR L'INTEGRATION DE BASES DE DONNEES SPATIALES

Une approche combinant l'utilisation de métadonnées et l'apprentissage automatique

Soutenue publiquement le 20 mai 2005 devant le jury composé de :

J.-D. Zucker :	Directeur de Thèse Professeur, Université Paris 13
S. Mustière :	Encadrant Docteur, Institut Géographique National
C. Weber :	Rapporteur Directrice de Recherche CNRS, Université Louis Pasteur, Strasbourg
S. Spaccapietra :	Rapporteur Professeur, Ecole Polytechnique Fédérale de Lausanne
A. Doucet :	Examinateur Professeur, Université Paris 6
C. Puech :	Président du Jury Directeur de Recherche CEMAGREF, UMR3S Cemagref – Engref
J.-P. Donnay :	Invité Professeur, Université de Liège

Remerciements

Si cette thèse a été menée à bien, c'est notamment grâce à l'encadrement sans faille de Sébastien Mustière. Je lui dois beaucoup. J'ai pu à tout moment venir discuter avec lui, exposer mes difficultés et mes incertitudes, et il a toujours été présent. Ses connaissances et son expérience dont j'ai bénéficié m'ont grandement aidé. J'ai sincèrement apprécié de travailler avec toi. Merci.

Je tiens également à remercier Jean-Daniel Zucker qui a dirigé ce travail. Il s'est toujours montré très intéressé par le sujet et m'a fait partager sa passion pour l'apprentissage automatique. Les discussions que nous avons eues furent très enrichissantes et m'ont guidées tout au long du parcours. Merci Jean-Daniel. Ton enthousiasme et la confiance que tu m'as accordée m'ont chaque fois remotivé. Merci de m'avoir aidé à prendre du recul sur mon travail.

Merci également à Anne Ruas, directrice du laboratoire COGIT. C'est elle, avec François Vauglin, qui est à l'origine du sujet. Merci pour l'encadrement durant les premiers mois de ma thèse et pour les remarques constructives que tu as pu formuler et qui m'ont fait progresser.

J'adresse également ma gratitude à Serge Motet et Patrice Bueso, directeurs successifs du Service de la Recherche, pour m'avoir accueilli au sein de l'IGN.

J'exprime par ailleurs ma reconnaissance à mes deux rapporteurs, Stefano Spaccapietra et Christiane Weber, ainsi qu'aux autres membres du Jury, Anne Doucet, Christain Puech et Jean-Paul Donnay. J'ai apprécié l'intérêt qu'ils ont manifesté à l'égard de mon travail et les nombreuses remarques émises durant la soutenance.

Je n'aurais sans doute pas entrepris une thèse de doctorat si Jean-Paul Donnay et Dimos Pantazis ne m'avaient pas encouragé à poursuivre ma formation à la suite de mes études initiales menées à l'Université de Liège. Je voudrais les remercier pour leurs incitations car je suis particulièrement heureux d'avoir choisi cette voie. Je tiens à leur exprimer ici toute ma reconnaissance.

Durant ces quelques années passées au COGIT, j'ai partagé mon bureau avec Olivier Bonin. Cette cohabitation fut très agréable et l'ambiance sereine qui régnait dans notre bureau a aussi contribué au bon déroulement de cette thèse. Merci pour les coups de main de toute nature que tu as pu me donner.

Je n'oublie pas mes autres collègues du COGIT : Patricia, $Fred_1$, Xav, Jenny, Arnaud, Sylvain, Nils, Sandrine, Cécile, $Fred_2$, Thierry, Julien, Hakima, Béné, Yann, Benoît, Catherine, Elisabeth, Christelle, Guillaume. Merci Eric pour ta disponibilité, notamment dans l'organisation du pot de thèse. Merci Jeff pour ces échanges que nous avons eus.

Je tiens également à remercier mes amis, pour tous ces bons moments passés ensemble, que ce soit à Paris ou sur les voies des Calanques, d'Orpierre, de Bourgogne ou de MurMur.

A mes parents, qui m'ont donné cette envie d'apprendre toujours davantage. A mes grands-parents, qui m'ont encouragé constamment.

Enfin, merci Isa pour ton soutien, ton écoute et ta patience qui fut mise à rude épreuve. Tu m'accompagnes maintenant à Strasbourg. Une nouvelle vie à deux nous attend…

INTRODUCTION

« Certains regrettent qu'il n'y ait plus de taches blanches sur les cartes. Ils oublient qu'il est un moyen simple de trouver l'aventure. Il suffit de partir sans carte, de n'avoir pour documents que les pages vierges du carnet de route, et somme toute de laisser carte blanche au voyage. »

A. Berroux
Le voyage inconnu, traité de l'insatisfaction
(Extrait de B. Amy, *Le voyage à la cime*)

1. CONTEXTE

LE PASSAGE DU MONDE REEL A LA BASE DE DONNEES GEOGRAPHIQUES

Avant de concevoir une base de données géographiques[1] (BDG), il est nécessaire de réduire la complexité de la réalité. D'une part, il n'est pas envisageable de représenter la totalité des phénomènes géographiques. Nous ne percevons de toute façon qu'une partie de la réalité et nous serions incapables de représenter l'ensemble de ce monde perçu. D'autre part, une base de données géographiques n'est généralement définie que pour un domaine d'applications spécifiques et pour une gamme d'échelles d'utilisation particulière. La représentation de l'ensemble des phénomènes ne serait donc ni utile, ni possible, et une représentation trop détaillée pourrait ne pas être adaptée à l'échelle d'utilisation.

Pour constituer sa base de données géographiques, le cartographe fait donc abstraction d'une série de phénomènes et crée son modèle de l'univers qui répond à une série de spécifications préalablement établies. Ce modèle de l'univers, à une date donnée, est appelé *terrain nominal* [Guptill et Morrison 1995, David et Fasquel 1997]. Celui-ci n'est pas directement accessible car il correspond à une base de données au contenu virtuellement parfait, sans erreur, respectant l'ensemble des spécifications. C'est une notion théorique qui est utile pour les contrôles qualité (figure 1). Les *spécifications*, permettant de définir ce modèle de l'univers, décrivent le contenu de la base (*le quoi*) et la manière de représenter les objets (*le comment*). Elles contiennent aussi le *schéma*[2] de la base de données qui reflète le quoi, et constitue en fait une partie des *métadonnées* (terme qui désigne d'une manière générale des données décrivant d'autres données).

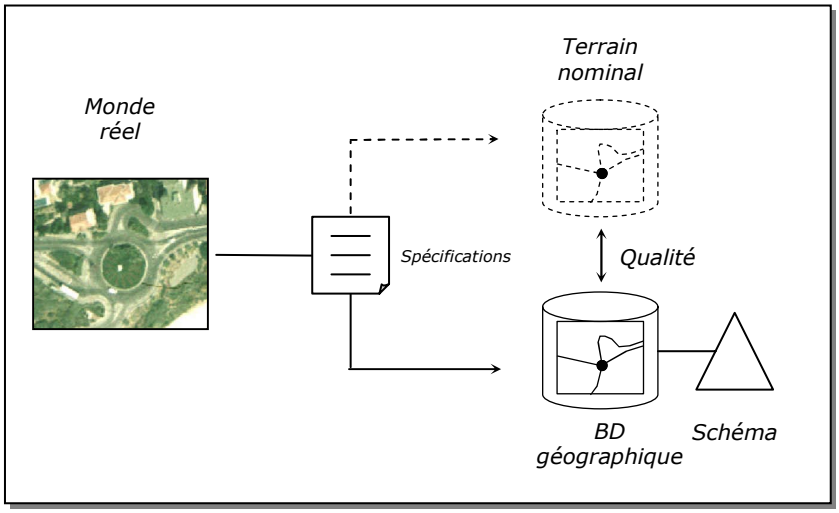


Figure 1. *Le monde réel, la notion théorique de terrain nominal et la base de données, dont le contenu est guidé par ses spécifications et son schéma.*

[1] Nous faisons référence ici aux bases de données géographiques *vectorielles* dans lesquelles la géométrie d'un objet est modélisée par une primitive de type point, ligne ou polygone. C'est ce mode de représentation que nous adoptons dans cette thèse. Par ailleurs, nous utiliserons les termes *bases de données géographiques* (BDG) et *bases de données spatiales* (BDS) comme synonymes.

[2] La notion de schéma d'une base de données sera définie dans le chapitre A.

Une base de données géographiques a un niveau de détail qui lui est propre [Ruas 2002a]. Ce niveau de détail concerne à la fois la géométrie et les informations attributaires. Pour la géométrie, on fait souvent référence aux notions de *granularité* et de *précision géométrique* [Vauglin 2002]. La granularité désigne la taille des plus petites formes représentées dans la base (ex : taille minimum des décrochements d'un bâtiment). La précision géométrique fait référence à l'écart estimé entre la position de l'objet dans la base et sa position nominale[3].

Pour les informations attributaires, le niveau de détail est défini par la *résolution sémantique*, qui correspond à une précision de description des attributs. Pour l'occupation du sol par exemple, on peut se limiter à ne distinguer que deux catégories (zones bâties et zones non bâties), ou au contraire définir de multiples catégories (bâtiments industriels, bâtiments commerciaux, vignes, cultures céréalières, cultures fourragères,…).

A ce niveau de détail de la base de données géographiques correspond une gamme d'échelles de représentation et d'utilisation des données. Pour reprendre les termes d'Anne Ruas : « … *une base de données correspond à une échelle de raisonnement ou d'analyse […] et peut être représentée cartographiquement dans une gamme d'échelles mathématiques compatible avec l'ordre de grandeur de l'ensemble des objets correspondants à cette analyse* » [Ruas 2002a, p. 43]. Ainsi, on n'utilise pas les données d'une base de résolution métrique[4] pour réaliser des cartes au 1/1.000.000 ou encore, on ne représente pas tous les bâtiments de manière individualisée dans une carte au 1/250.000. Ils seraient illisibles, et leur représentation ne serait pas adaptée aux analyses effectuées à ce niveau. Trois extraits de bases de données présentant des niveaux de détails différents et répondant à des besoins d'applications distincts sont présentés à la figure 2.

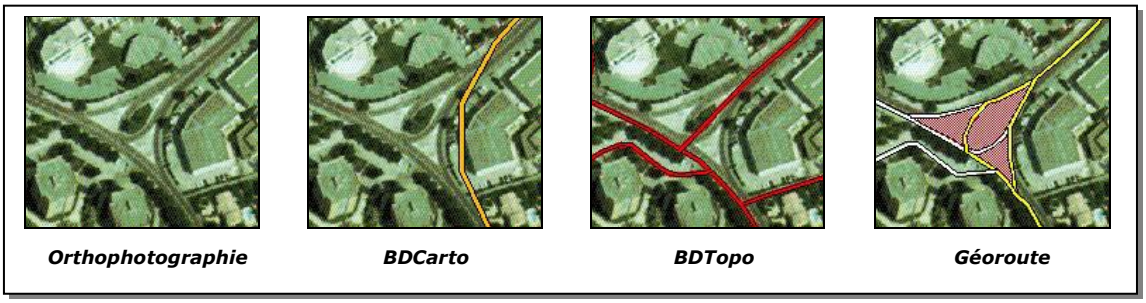


***Figure 2**. Une orthophotographie représentant le monde réel et la saisie du réseau routier selon les spécifications de trois bases de données de résolutions différentes.*

GENERALISATION CARTOGRAPHIQUE

Le passage d'une représentation cartographique à grande échelle à une représentation cartographique à petite échelle ne se limite pas une simple homothétie qui engendrerait des problèmes de lisibilité (figure 3). Il s'accompagne entre autres d'une élimination d'objets, d'une simplification de leur forme, d'une modification de

[3] Le terme précision est utilisé ici au sens large du terme bien que les standards de qualité distinguent la précision de l'exactitude.

[4] Le terme *résolution*, qui donne une idée sur le niveau de détail, est fréquemment employé pour les bases de données géographiques. Contrairement au mode image (ou raster), sa définition est néanmoins ambiguë pour le mode vectoriel [Ruas 2002a]

leur dimension, d'une agrégation de leur géométrie : c'est ce qu'on appelle la *généralisation cartographique* [Cuenin 1972].

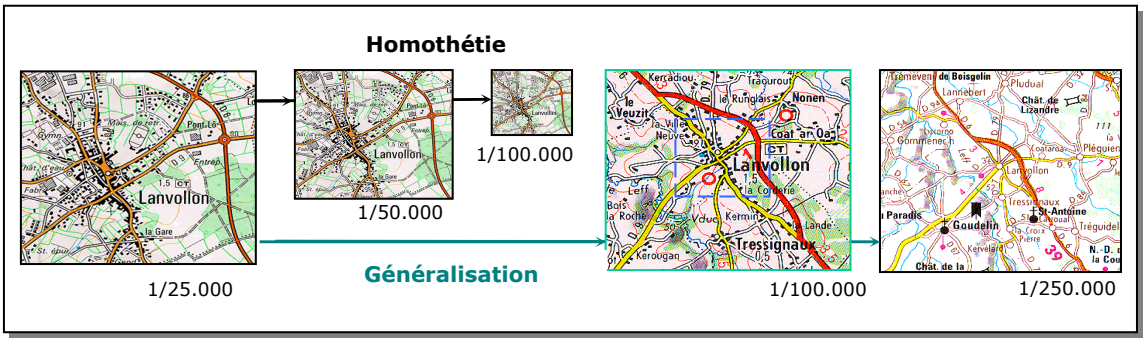


***Figure 3**. Le passage d'une carte à grande d'échelle à une carte à petite échelle ne se limite pas à une homothétie. Pour que objets soient différenciables et interprétables, on les généralise.*
(Source : [Bard 2004])

La généralisation est mise en œuvre, chaque fois que l'on souhaite réduire la quantité d'informations, pour ne retenir que la plus pertinente, celle adaptée à l'échelle de représentation et au niveau d'analyse. On l'utilise donc pour passer d'une carte au 1/25.000 à une carte au 1/50.000 par exemple. On l'utilise aussi pour passer de la base de données géographiques à la base de données cartographiques. La première contient les données brutes de granularité et de précision généralement plus grande que celles requises pour la production et l'impression des cartes auxquelles elles sont destinées. La seconde contient les données prêtes à être imprimées, qui ont donc été transformées et symbolisées en respectant les contraintes du langage cartographique.

L'INDEPENDANCE DES BASES DE DONNEES GEOGRAPHIQUES

On imagine assez facilement la production d'une base de données géographiques très détaillée à partir de laquelle on dériverait l'ensemble des cartes à grande, moyenne et petite échelle par un processus de généralisation automatique [Brassel et Weibel 1988]. Toutes les cartes proviendraient ainsi d'une même source et seraient générées à la demande suivant les besoins de l'utilisateur. C'est une solution qui paraît idéale mais qui ne peut malheureusement pas être mise en œuvre aujourd'hui pour diverses raisons.

La première raison est liée à la complexité du processus de généralisation et à son automatisation [Ruas 2002b]. La généralisation reste encore aujourd'hui un processus peu formalisé et subjectif : deux cartographes peuvent ainsi réaliser deux généralisations différentes d'une même scène avec un niveau de qualité comparable. Les opérations qu'ils utiliseront seront différentes mais les résultats en termes de lisibilité et d'efficacité seront équivalents (figure 4). Par ailleurs, la généralisation est un processus qui dépend fortement du contexte et du type d'objet à traiter. Un réseau hydrographique ne sera pas généralisé de la même manière qu'un réseau routier, et les opérations sur les objets ne seront pas appliquées de la même façon sur l'ensemble de la carte (par exemple, il est probable qu'une route de campagne menant à une maison isolée soit conservée, contrairement à une voie sans issue en plein centre-ville). L'automatisation du processus est donc complexe car en plus de la difficulté de concevoir des algorithmes de transformation géométrique, s'ajoute le problème de l'analyse de l'objet à traiter et de son contexte. S'il existe aujourd'hui des prototypes opérationnels capables de généraliser automatiquement certaines

catégories d'objets [Lecordix et al. 1997, Ruas 1999, Duchêne et Regnauld 2002, Duchêne 2004], il n'est pas encore envisageable de les utiliser dans un contexte de production, sans envisager une tâche de retouches interactives [Ruas 2002b].

***Figure 4**. Résultats de la généralisation d'une même source par plusieurs cartographes (Source : [Spiess 1995]).*

La seconde raison qui empêche d'adopter cette solution de base unique très détaillée est liée aux contraintes de production elles-mêmes. Plus une base est précise et plus il faut de temps pour la concevoir. Cela signifie que pour couvrir l'ensemble d'un territoire et produire les cartes à petites échelles, il faut attendre d'avoir saisi l'ensemble des objets de référence et les données à grande échelle, ce qui n'est pas non plus envisageable.

Enfin, on peut se demander, indépendamment de ces contraintes techniques, dans quelle mesure un produit est dérivable d'un autre. En plus de la notion d'échelle, il faut tenir compte de la notion de point de vue. Il n'est pas évident qu'à partir d'une seule base de données géographiques, aussi précise soit elle, on puisse dériver un éventail de BDG destinées à des applications très différentes et répondant à des besoins divers.

Face à ces difficultés, les instituts cartographiques et plus généralement les producteurs de données géographiques ont opté pour la production et la maintenance de plusieurs bases de données de différentes résolutions [Guptill 1989]. A l'IGN par exemple, la BDTopo qui présente une résolution métrique est destinée, entre autre, à la production de cartes au 1/25.000. La BDCarto, de résolution décamétrique, est produite notamment pour pouvoir effectuer des analyses aux échelles régionale et départementale. Des études existent pour élaborer le processus permettant de réaliser depuis cette source, les cartes aux 1/100.000 [Jahard et al. 2003]. La BD Géoroute, qui présente une résolution similaire à la BDTopo est, quant à elle, produite pour des applications de navigation routière. La géométrie des carrefours et des ronds-points est plus détaillée et l'information attributaire se rapportant aux routes est plus riche.

Toutes ces bases de données sont donc indépendantes et produites de manière séparée. On retrouve un même phénomène géographique dans plusieurs sources mais aucun lien explicite n'existe entre les diverses représentations de ce phénomène.

LES RAISONS QUI MOTIVENT L'INTEGRATION DES BASES DE DONNEES GEOGRAPHIQUES

L'indépendance des bases de données géographiques pose aujourd'hui un certain nombre de problèmes. Le fait qu'il n'existe aucun lien entre les objets homologues (objets représentant le même phénomène) implique : une répétition des opérations de

mises à jour, un manque de cohérence entre les différentes BD, et une impossibilité d'effectuer des analyses à plusieurs niveaux de détail.

Le problème lié à la duplication des opérations de mises à jour concerne essentiellement les producteurs de données. Chaque donnée d'évolution est aujourd'hui intégrée séparément dans chaque base de données, à partir d'informations de sources diverses, ce qui a pour effet d'augmenter les coûts de maintenance des produits. La mise en correspondance des différentes bases de données permettrait d'éviter cette multiplication des opérations en propageant directement les évolutions dans les différentes sources [Badard 2000]. Ceci suppose néanmoins une transformation systématique des données d'évolution dans les sources pour que les représentations respectent les spécifications de chaque base dans laquelle elles seront intégrées.

Le manque de cohérence entre les données des BD provient principalement de deux faits : une différence d'actualité entre les bases et la présence d'erreurs de saisie dans ces bases. Les bases n'ont pas les mêmes rythmes de mise à jour et la politique d'entretien des données n'est pas nécessairement la même (mise à jour à la demande, mise à jour en continu, etc.). Les données d'évolution ne sont donc pas intégrées au même moment et des incohérences entre les jeux de données peuvent apparaître. Il en va de même des erreurs de saisie existant dans les bases. En explicitant le lien existant entre ces différentes sources, il est possible de détecter et corriger des erreurs, de réactualiser certains thèmes et au final, de rendre l'ensemble cohérent.

L'indépendance des bases présentant des résolutions différentes ne permet pas non plus d'effectuer des analyses multi-niveaux. Un exemple de telles analyses fréquemment mentionné concerne le domaine des transports [Car et Frank 1994, Devogele et al. 2002]. On pourrait envisager d'utiliser deux niveaux de détail différents pour calculer le déplacement en voiture d'une ville à une ville autre. Dans un premier temps, on utiliserait la représentation la plus détaillée pour naviguer dans la ville de départ. Une fois sorti de la ville, on passerait alors à la représentation la moins détaillée pour laquelle seul le réseau routier principal serait utilisé. Enfin, on changerait à nouveau de représentation lors de l'arrivée dans la ville cible. Cette navigation n'est envisageable que si les deux jeux de données sont associés. En géographie, la combinaison et le passage d'une échelle à une autre est essentiel car les échelles rendent compte des différents niveaux d'organisation spatiale. La problématique d'une question géographique change ainsi avec son échelle [Ferras 1995].

Toutes ces raisons conduisent les producteurs de données géographiques ainsi que les utilisateurs à vouloir établir des relations entre les différentes bases de données spatiales. En d'autres termes, on souhaite aujourd'hui *intégrer* des données géographiques provenant de sources multiples pour accroître les potentialités d'utilisation des produits et assurer une meilleure cohérence entre eux. Cette problématique d'intégration constitue le cadre de notre travail de recherche.

2. SUJET

Le sujet de recherche que nous traitons dans cette thèse porte en particulier sur l'étude de la cohérence entre les données des différences sources. Lors de l'intégration des bases, à l'issue de la mise en correspondance des données, les données reliées peuvent présenter des différences de représentation. La question qui se pose est de

savoir si ces différences se justifient ou pas (figure 5). Sont-elles cohérentes entre elles ? Puisque les bases peuvent présenter des niveaux de détail différents et plus généralement, des terrains nominaux différents, il est naturel que des différences de représentation apparaissent. Toutefois, nous l'avons déjà souligné, il peut exister des erreurs de saisie dans les bases et ces bases n'ont pas nécessairement la même actualité au moment de l'intégration. Dans ce cas, il peut apparaître des incohérences entre les données. Ces incohérences sont problématiques car elles peuvent conduire à des réponses contradictoires en fonction de la représentation utilisée dans un système intégré. Afin de garantir une intégration cohérente des données, il faut être capable de détecter les différences de représentation et les interpréter.

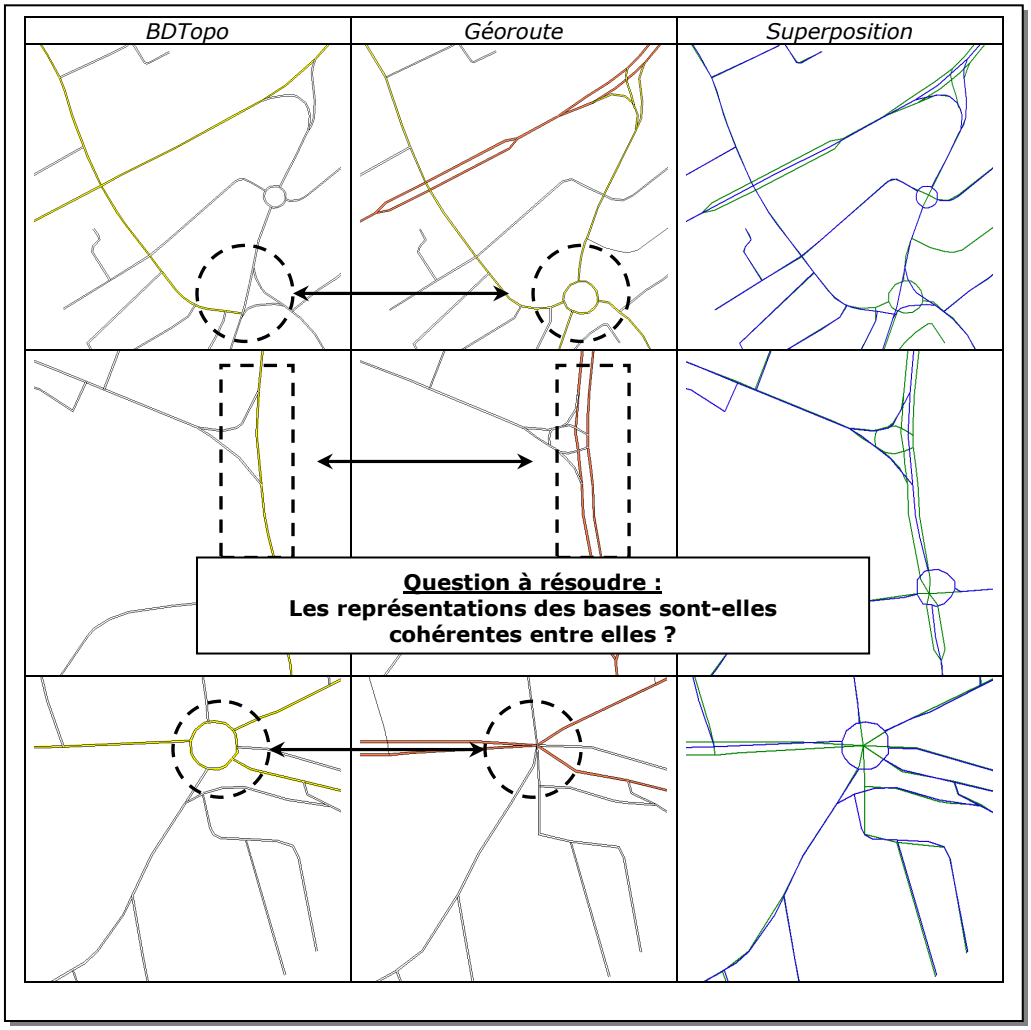


Figure 5. *Exemples de différences de représentation entre deux BDG.*

L'objectif de cette thèse est précisément de proposer une solution à ce problème qui reste non résolu aujourd'hui. Peu de travaux existent à ce sujet dans le domaine des bases de données spatiales. Nous voulons ainsi combler ce hiatus en proposant une méthodologie permettant d'évaluer la cohérence inter-représentations dans le cadre de l'intégration. Nous voulons également appliquer cette méthodologie en développant un système informatique capable d'expliquer l'origine de chaque différence de représentation de manière la plus automatique possible. Nous nous intéressons uniquement à la cohérence entre les *données* des bases et non aux différences entre les schémas. Aussi, notre objectif est d'expliquer les différences et non de corriger les incohérences. Nous ne traitons qu'une partie de l'intégration des

bases de données spatiales et en particulier, celle de la mise en correspondance des données et de l'évaluation de leur cohérence.

3. ÉLEMENTS DE L'APPROCHE PROPOSEE

Pour apprécier la cohérence des représentations, il est nécessaire de choisir une référence. Dans cette thèse, les *spécifications* servent de référence pour juger la conformité des représentations. Ces documents décrivent les règles de sélection et de modélisation des objets. Ils constituent des métadonnées permettant de juger si les représentations sont *équivalentes* ou *incohérentes* (ces termes seront définis dans le chapitre B). Nous considérons que les différences de représentations sont normales, même si ces représentations sont fortement éloignées, pourvu qu'elles soient justifiées par les spécifications de chacune des BD. Dans le cas contraire, les représentations sont jugées incohérentes.

L'utilisation de ces documents est toutefois insuffisante. Les spécifications fournies par les producteurs décrites en langue naturelle peuvent être imprécises ou incomplètes. Puisque les données des bases reflètent dans une certaine mesure les spécifications utilisées lors de leur saisie (spécifications constatées), nous exploitons aussi cette seconde source de connaissances pour mener l'évaluation de la cohérence.

Ayant fait ces choix, nous devons répondre aux deux questions suivantes :

- Comment acquérir ces connaissances et les représenter dans un langage manipulable par une machine pour réaliser l'évaluation de la cohérence ?
- Comment réaliser l'évaluation de la cohérence en exploitant ces connaissances ?

Nous répondons à la première question par la définition de la méthode *MACO* (Méthode d'Acquisition de connaissances pour l'évaluation de la COhérence). Cette méthode est composée d'une étape d'analyse des spécifications qui permet de déduire des connaissances des spécifications, et d'une étape d'acquisition de connaissances par apprentissage automatique supervisé qui permet d'induire des connaissances à partir des données.

Nous répondons à la seconde question par la méthode *MECO* (Méthode d'Évaluation de la COhérence). Cette méthode est composée de différentes étapes : enrichissement des bases, contrôle intra-base, appariement, contrôle inter-bases, évaluation globale.

Ces deux méthodes, *MACO* et *MECO*, constituent notre méthodologie générale d'évaluation de la cohérence. L'approche que nous proposons est synthétisée à la figure 6.

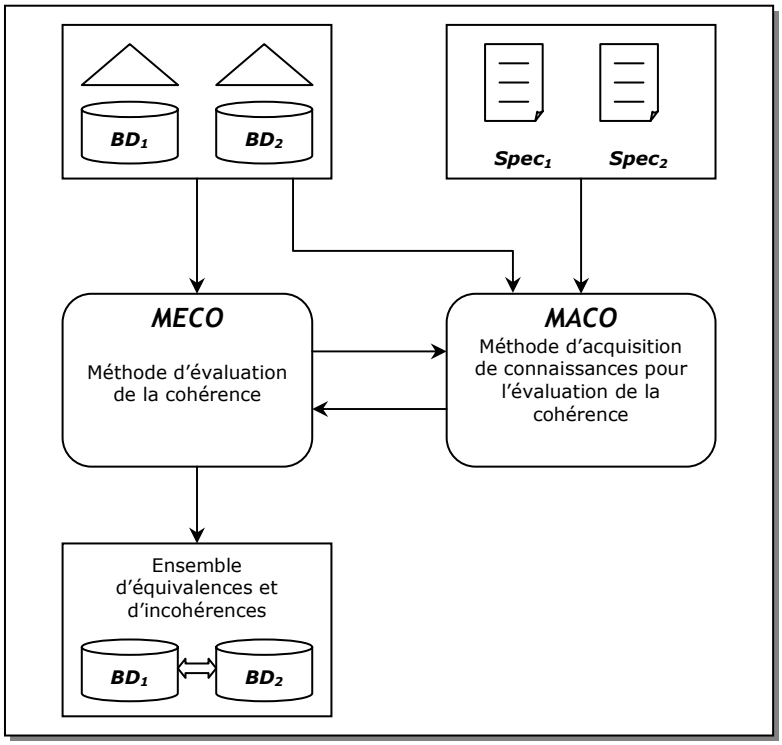


Figure 6. *Méthodologie d'évaluation de la cohérence inter-représentations fondée sur les méthodes MACO et MECO.*

4. ORGANISATION DE LA THESE

Le chapitre A constitue un état de l'art sur l'intégration des bases de données classiques et géographiques. L'objectif de ce chapitre est de fournir une vision générale du problème d'intégration et de montrer la diversité des approches existantes pour y répondre. Cette présentation nous permet de préciser à quel niveau se situe l'étape que nous traitons en nous plaçant dans un processus d'intégration, celui proposé par [Parent et Spaccapietra 1996, Parent et Spaccapietra 2001]. Nous exposons également les spécificités de l'intégration des bases de données spatiales. Nous détaillons les solutions proposées aujourd'hui pour évaluer la cohérence inter-représentations. Nous concluons en montrant qu'à notre connaissance, il n'existe pas à l'heure actuelle de méthodologie globale d'évaluation et que les spécifications ne sont pas utilisées formellement dans le cadre de l'intégration de bases de données géographiques.

Le chapitre B se compose de deux parties. La première est consacrée à la définition de la notion de cohérence que nous adoptons dans cette thèse. Nous exposons cette notion de cohérence en nous appuyant sur le modèle *KRA* [Saitta et Zucker 2001, Zucker 2001, Zucker 2003]. La seconde partie expose la représentation actuelle des spécifications fournies par les producteurs. Nous mettons en évidence la difficulté de les utiliser en l'état et la nécessité de changer de langage de représentation pour les manipuler automatiquement. Nous montrons également que l'acquisition de connaissances à partir de données peut être utile.

La méthode *MECO* est ensuite définie dans le chapitre C. Les étapes de la démarche d'évaluation ainsi que les outils et les connaissances utiles à sa mise en

œuvre sont exposés. Nous décrivons chaque étape de la méthode : l'enrichissement, le contrôle intra-base, l'appariement, le contrôle inter-bases et l'évaluation globale. Nous considérons à ce niveau que les connaissances sont acquises. Certaines d'entre elles, en particulier celles permettant de réaliser les contrôles intra-base et inter-bases, sont représentées dans un langage à base de règles et introduites dans un système-expert. Les caractéristiques des systèmes-experts sont présentées en fin de chapitre.

La méthode *MACO* fait l'objet du chapitre D. Elle est consacrée à la démarche d'acquisition de connaissances pour l'évaluation de la cohérence. L'étape d'analyse des spécifications est d'abord présentée. La description actuelle des spécifications rendant l'analyse interactive laborieuse, nous proposons de formaliser la description des spécifications selon un modèle formel. Ce modèle est un outil d'aide à l'acquisition des spécifications fournies par les producteurs de données. La seconde étape de *MACO* est une étape d'induction de connaissances par apprentissage automatique supervisé. Elle doit être mise en œuvre si les spécifications fournies par les producteurs sont incomplètes ou imprécises, ou si l'on souhaite réaliser l'évaluation à l'aide des spécifications constatées dans les données. Deux approches sont proposées pour définir la forme des exemples d'apprentissage : l'approche par *classification directe* et l'approche par *prédiction, comparaison et classification*.

La chapitre E présente les expérimentations et les résultats de la méthodologie élaborée. Nous décrivons d'abord une application développée pour étudier la cohérence entre les représentations de ronds-points issues de deux bases de données géographiques de l'IGN. La seconde expérimentation est consacrée à l'évaluation de la cohérence entre les représentations de bâtiments. Nous montrons l'applicabilité de la méthodologie à des bases de données géographiques présentant des niveaux de détails différents. Finalement, nous étudions les possibilités d'utilisation de l'apprentissage automatique pour acquérir des règles de correspondance entre attributs de routes. Toutes ces applications ont été développées dans un prototype conçu à cette fin : le prototype HÉTÉROGENE.

Nous concluons en résumant la méthodologie proposée avec les résultats obtenus lors de son application et exposons les perspectives de recherche.

GUIDE DE LECTURE

Le lecteur familier avec le problème d'intégration de bases de données non spatiales pourra survoler la section A.2. du premier chapitre sans conséquence sur la compréhension de la suite du mémoire. De même, le lecteur familier à l'apprentissage automatique supervisé peut survoler la section D.4.1. du chapitre D qui expose les principes de ces techniques.

CHAPITRE A

INTEGRATION DES BASES DE DONNEES CLASSIQUES ET GEOGRAPHIQUES

A.1 INTRODUCTION

Ce chapitre est consacré à l'intégration des bases de données classiques et géographiques. Nous présentons d'abord dans la section A.2. le problème d'intégration de manière générale. Nous nous focalisons ensuite sur l'intégration des bases de données classiques en présentant dans un premier temps les architectures qu'il est possible de mettre en place pour relier des bases (A.3.1) et en exposant ensuite un processus d'intégration (A.3.2). A la suite de cette partie, nous mettons en évidence les particularités de l'intégration des bases de données spatiales (A.4.1.) et présentons les approches proposées dans la littérature pour intégrer ce type de BD (A.4.2. et A.4.3.). Nous concluons finalement le chapitre en montrant que l'évaluation de la cohérence inter-représentations dans le cadre des BDG a été peu étudiée. Ceci nous permet de souligner l'intérêt d'effectuer notre recherche.

A.2 LE PROBLEME D'INTEGRATION

Selon la définition du dictionnaire [Rey-Debove et Rey 1988], l'*intégration* désigne « *l'établissement d'une interdépendance plus étroite entre les parties d'un être vivant ou les membres d'une société* ». C'est également une « *opération par laquelle un individu ou un groupe s'incorpore à une collectivité, à un milieu* ». Le dictionnaire renvoi par ailleurs le lecteur à la définition des termes *assimilation*, *fusion*, *incorporation*, *insertion* et *unification*. L'intégration désigne usuellement l'opération qui consiste à incorporer une chose dans une autre.

En informatique, l'intégration est liée à la notion d'*interopérabilité* [Parent et Spaccapietra 2001]. Celle-ci se traduit par la capacité d'un système ou des composants d'un système à partager ses données et ses fonctions avec d'autres systèmes [Bishr 1997]. Plus précisément, l'interopérabilité est assurée si des messages et des requêtes peuvent être échangés entre deux systèmes et s'il est possible de les faire opérer comme une unité pour réaliser une tâche commune [Solar et Doucet 2002]. L'intégration désigne une opération qui permet de rendre plusieurs systèmes interopérables. L'intégration fait donc davantage référence à la notion d'association, de connectivité, de coopération.

Dans cette thèse, nous nous intéressons à l'intégration de systèmes qui sont des bases de données (BD). Un base de données est constituée d'un *schéma* et de *données* relatives à un domaine. Le schéma est une « *description au moyen d'un langage particulier d'un ensemble de données particulier* » [Gardarin 1999, p.16]. Un schéma est donc définit dans un langage. Un langage de description de données est un « *langage supportant un modèle et permettant de décrire les données d'une base d'une manière assimilable par une machine* » [Gardarin 1999, p.16]. Le langage UML (« *Unified Modelling Language* ») est un exemple [Muller 1997]. Un langage repose donc sur un *modèle*. Un modèle de description de données est un « *ensemble de concepts et de règles de composition de ces concepts permettant de décrire des données* » [Gardarin 1999, p.16]. UML repose ainsi sur le modèle *orienté-objet* (O.O.).

On distingue différents schémas pour une base de données. Ces schémas sont associés aux trois niveaux de description d'une BD (niveaux définis par le groupe de normalisation ANSI/X3/SPARC) : niveau conceptuel, niveau interne (logico-physique) et niveau externe. Le *schéma conceptuel* décrit la structure de la base sans se soucier de l'implémentation machine. Nous verrons beaucoup d'exemples de ce type de

schéma dans ce mémoire. Le schéma *logique* est une traduction du schéma conceptuel selon le modèle du système qui gère les données de la base, le SGBD (Système de Gestion de Base de Données). Il est donc cette fois lié au système informatique adopté. Le schéma *interne* donne finalement une description des données en termes de représentation physique (structure de stockage supportant les données). Il suit la définition du schéma logique.

Il existe également le schéma externe (qu'on qualifie aussi de vue externe). Il se définit comme la « *description d'une partie de la base extraite ou calculée à partir de la base physique, correspondant à une vision d'un programme ou d'un utilisateur, donc à un arrangement particulier de certaines données* » [Gardarin 1999, p.20]. Le modèle sur lequel repose ce schéma est le même que celui du niveau logique. Contrairement aux schémas conceptuel et interne, il peut exister plusieurs schémas externes qui correspondent à plusieurs « visions » de la base.

Nous nous intéressons donc à l'intégration de systèmes qui sont des bases de données, lesquelles sont décrites par leurs schémas et leurs données.

Deux bases de données interopérables ne sont pas nécessairement intégrées. L'interopérabilité peut être réalisée de différentes manières en supportant plusieurs niveaux d'intégration [Parent et Spaccapietra 2001]. La communication entre différentes bases de données à l'aide d'une connection ODBC[5] par exemple est une forme d'interopérabilité. Les bases de données ne sont pas pour autant intégrées. Elles peuvent seulement communiquer. Une première forme d'intégration existe à partir du moment où un utilisateur peut interroger simultanément plusieurs bases au moyen d'un langage commun [Rusinkiewitz et al. 1989, Litwin et al. 1989]. L'intégration est faible dans ce cas car on ne se soucie pas de l'hétérogénéité du contenu des bases. L'hétérogénéité est seulement traitée du point de vue de l'accès et du langage d'interrogation des sources. Une intégration plus forte des bases de données apparaît lorsqu'on tient compte des différences sémantiques apparaissant entre les schémas et les données et qu'on fait en sorte de fournir une vision unifiée des bases [Sheth et Larson 1990]. Dans ce cas, l'hétérogénéité est traitée non seulement du point de vue de l'accès aux bases mais aussi du point de vue de leur représentation. On définit une représentation intégrée des bases.

Ce qui rend le problème d'intégration particulièrement complexe, c'est l'existence de l'hétérogénéité des bases. L'hétérogénéité se traduit notamment par des différences d'ordre technique (différences de langages d'interrogation, différences de SGBD, différences de formats, etc.), des différences de structuration des bases (différences de modélisation) et des différences sémantiques (différences de signification ou d'interprétation qu'on pourrait attribuer aux éléments des schémas et des données qui se correspondent). Pour les bases de données spatiales, l'hétérogénéité est encore plus importante en raison de l'existence d'une géométrie associée aux données. Il faut tenir compte notamment des différences de mode de représentation des données (vecteur/raster), des différences de niveau de détail et de modélisation des données, des différences de qualité des données, des différences de systèmes de référence géodésique, etc.

Le problème que nous traitons dans cette thèse, l'évaluation de la cohérence des représentations de plusieurs bases de données géographiques, est un problème lié à l'hétérogénéité. Nous voulons étudier si l'hétérogénéité des représentations des données découle des différences de critères de saisie des bases (et plus généralement,

[5] ODBC est l'acronyme de « *Open DataBase Connectivity* »

des différences de terrains nominaux), ou des différences de qualité des données ou encore, des différences d'actualité.

Nous allons exposer plus en détails le problème d'intégration tout au long de ce chapitre en présentant différentes approches qui permettent d'y apporter des solutions. Nous nous intéressons d'abord à l'intégration de bases de données classiques.

A.3 INTEGRATION DE BASES DE DONNEES CLASSIQUES

Nous nous focalisons dans cette partie sur le problème d'intégration des bases de données dites classiques, par opposition aux bases de données géographiques. Nous présentons d'abord les différents types de systèmes intégrés, c'est-à-dire les architectures qu'il est possible de mettre en place pour faire coopérer des bases de données (partie A.3.1.). Nous les classons en fonction du degré d'intégration. Nous exposons ensuite un processus d'intégration et détaillons les différentes étapes le composant (partie A.3.2).

Cette première partie permet de mettre en évidence la complexité du problème d'intégration et la diversité des approches existantes pour répondre à ce problème. Notre objectif n'est pas de recenser de manière exhaustive l'ensemble des travaux sur le sujet. C'est une tâche qui s'avère impossible aujourd'hui en raison du nombre même de contributions existantes. Nous souhaitons plutôt fournir un aperçu des grandes approches proposées dans la littérature et préciser la terminologie couramment utilisée dans le domaine de l'intégration.

A.3.1 TYPOLOGIE DES SYSTEMES INTEGRES

Il existe plusieurs taxonomies des systèmes intégrés [Sheth et Larson 1990, Solar et Doucet 2002]. Celles-ci se fondent sur différents critères tels que *le degré d'intégration* (systèmes faiblement couplés, fortement couplés), le *type de données* à intégrer (BD, données peu structurées), les *dimensions* des systèmes d'informations (distribution, autonomie, hétérogénéité). Nous avons adopté la classification de [Busse et al. 1999] qui se fonde sur le degré d'intégration des données. C'est un critère qui est souvent retenu dans la littérature.

Nous discuterons ainsi des *systèmes faiblement couplés* que sont les *systèmes multibases* [Rusinkiewitz et al. 1989, Litwin et al. 1989]. Leurs caractéristiques sont présentées dans la section A.3.1.1. Nous découvrirons également les *systèmes fortement couplés* correspondant aux *systèmes fédérés* [Sheth et Larson 1990, Ahmed et al. 1991]. Leur présentation fait l'objet de la section A.3.1.2. Les *systèmes de médiation* présentent aussi un niveau d'intégration élevé. Ils diffèrent cependant des approches fédérées sur plusieurs aspects. Nous les présenterons donc séparément dans la section A.3.1.3. Enfin, il existe aujourd'hui les *entrepôts de données* qui sont construits à partir d'un ensemble d'informations extraites de différentes bases [Doucet et Gançarski 2001]. Ces systèmes sont différents des précédents. L'intégration est cette fois *matérialisée*. Nous les exposerons dans la section A.3.1.4.

Il faut préciser qu'il n'existe pas un véritable consensus sur la terminologie utilisée. Certains auteurs qualifient les systèmes multibases de systèmes fédérés particuliers faiblement couplés [Jakobovits 1997], tandis que d'autres utilisent le

terme « multibase » (plus exactement « *multidatabase* ») d'une manière plus générale, pour désigner tout système intégré [Sheth et Larson 1990].

A.3.1.1 SYSTEMES MULTIBASES

Les systèmes multibases sont des systèmes dits faiblement couplés [Busse et al. 1999]. On les caractérise de cette manière car ils n'offrent pas une vision unifiée des données. Il n'existe pas de *schéma global* permettant un accès transparent aux différentes sources de données. La coopération est seulement assurée par l'intermédiaire d'un langage commun : le *langage multibase* (de type SQL notamment [Litwin et al. 1989]).

L'utilisateur peut poser une requête aux différentes systèmes à l'aide de ce langage commun, sans se soucier de l'hétérogénéité des systèmes source. Ceci ne signifie pas qu'une requête nécessitant l'accès à diverses sources est exécutée en une seule fois. L'utilisateur doit envoyer autant de requêtes qu'il y a de sources impliquées. C'est donc a l'utilisateur de relier les différents réponses aux requêtes formulées.

Ces systèmes sont donc faiblement intégrés et gardent une grande autonomie. Les sources peuvent évoluer de manière indépendante, sans conséquence sur leur accès. Cependant, la cohérence entre ces sources n'est pas assurée. L'hétérogénéité n'est pas traitée en amont. L'intégration est dynamique : les correspondances entre les données ne sont pas prédéfinies.

Certains systèmes offrent la possibilité de rendre les correspondances persistantes entre les sources par la création de *vues multibases*. Les requêtes multibases sont exprimées sur ces vues multi-bases. C'est le cas du système MSQL [Litwin et al. 1989].

A.3.1.2 SYSTEMES FEDERES

A l'inverse des systèmes multibases, les systèmes fédérés sont dits fortement couplés [Busse et al. 1999]. Ils se caractérisent par l'existence d'un schéma unifié appelé *schéma fédéré* qui constitue l'interface d'accès au système intégré. L'intégration se situe au niveau des schémas.

La conception de ce schéma fédéré suppose d'unifier les schémas source et de traiter leur hétérogénéité. Il est nécessaire d'identifier les correspondances et de résoudre les conflits entre les éléments des schémas. Ces correspondances peuvent être exprimées à l'aide de différents langages, au moyen de règles ou à travers une ontologie par exemple. Il y a donc cette fois une vision unifiée des sources (ou plus justement d'une partie). L'intégration offre un accès commun aux sources et une représentation commune.

L'intégration des systèmes fédérés est statique : les liens de correspondances entre les schémas sont prédéfinis. Ce n'est plus à l'utilisateur d'établir ces liens. Ce sont les administrateurs des bases sources qui définissent les sous-ensembles des données qu'ils souhaitent intégrer. L'accès aux données peut se faire de deux manières différentes : via les schémas locaux (schémas des BD source) ou via le schéma fédéré. Les bases sources gardent leur autonomie et restent sous contrôle de leur administrateur. C'est une des particularités des systèmes fédérés. En général, seule une partie des données des différentes sources est mise en commun. Pour cette raison, ces systèmes sont parfois considérés comme faiblement couplés [Laurini et

Millert-Raffort 1993]. Le niveau d'intégration est en effet moins élevé que celui imposé par une base de données répartie ou distribuée[6]. Cette dernière suppose une intégration totale des données sources [Parent et Spaccapietra 2001]. Les bases de données initiales perdent donc complètement leur autonomie. La fédération est un compromis entre une intégration nulle et totale.

[Sheth et Larson 1990] ont défini une architecture de référence pour les systèmes fédérés (figure 7). Elle est composée de 5 couches différentes :

- Les *schémas locaux* : il s'agit des schémas conceptuels initiaux des différentes BD source. Il en existe autant qu'il y a de systèmes sources à intégrer. Ces schémas locaux peuvent être exprimés dans des modèles différents.
- Les *schémas pivots* : il s'agit des schémas locaux traduits dans le *modèle commun* (ou *modèle canonique*), c'est-à-dire le modèle utilisé pour la fédération.
- Les *schémas d'export* : ils correspondent à un extrait des schémas pivots. Seuls les éléments que les administrateurs des bases sources souhaitent fédérer sont exportés.
- Les *schémas fédérés* : il s'agit des schémas d'export intégrés. Il peut en exister plusieurs. Ils offrent un vue unifiée des schémas exportés selon le modèle canonique.
- Les *schémas externes* : il s'agit de vues définies pour des groupes d'utilisateurs particuliers du système fédéré (reposant sur le modèle canonique). Ils n'offrent l'accès qu'à un sous-ensemble des données.

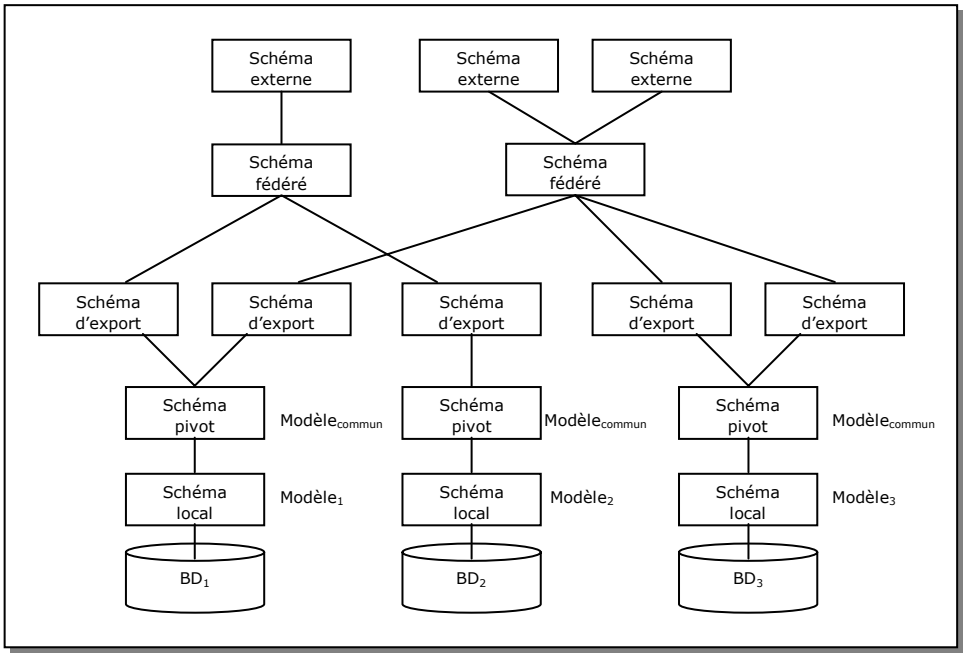


***Figure 7**. Architecture des systèmes fédérés (D'après [Sheth et Larson 1990])*

[6] Les bases de données réparties (BDR) correspondent à des bases constituées d'un ensemble de parties logiquement reliées entre elles et physiquement distribuées sur un réseau. La somme de toutes les parties forme la BDR gérée par un SGBD réparti. Les parties ne sont donc pas indépendantes. L'accès ne se fait que via le schéma global qui contient tous les concepts des schémas locaux (principe de transparence). On trouvera davantage d'informations sur les bases de données réparties et distribuées dans [Valduriez et Ozsu 1999].

La conception de systèmes fédérés se fait de manière *ascendante* [Busse et al. 1999]. On définit le schéma fédéré à partir des schémas source après une analyse des correspondances entre les schémas source (plan horizontal), et après avoir traité les différents conflits entre les éléments des schémas (figure 8). L'identification des relations entre les schémas source permet donc d'aboutir au schéma fédéré. La conception est différente de celle qui peut être adoptée pour les bases de données réparties. Pour celles-ci, les schémas locaux sont définis à partir du schéma global (*approche descendante*). Les correspondances entre le schéma global et les schémas locaux sont analysées dans un plan vertical (décomposition) et le principal problème est de traiter la répartition des données (fragmentation, duplication,…).

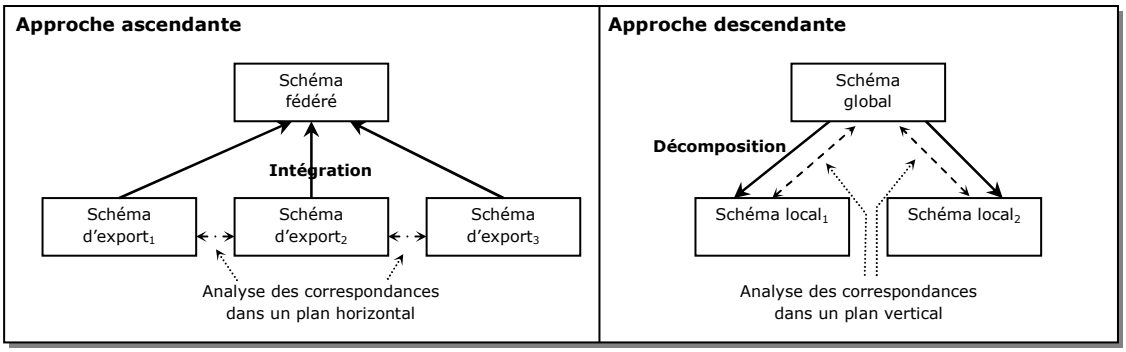


***Figure 8**. Stratégies de développement des systèmes fédérés et répartis (D'après [Busse et al. 2000])*

Il faut noter que la fédération est une intégration *virtuelle*. L'intégration n'est pas matérialisée, comme pour les entrepôts de données. Les données restent dans leur source d'origine. Grâce aux correspondances définies entre les schémas source et à l'existence de mécanismes permettant de traduire les requêtes posées sur le schéma fédéré dans les termes des schémas source, il est possible d'accéder aux différentes bases [Busse et al. 1999]. Ce type de couplage a des conséquences importantes sur l'évolution du système. Un changement de configuration ou de schéma dans les bases source doit se répercuter dans le schéma fédéré. Suivant le degré d'autonomie et d'hétérogénéité, des problèmes d'incohérences peuvent apparaître entre le système fédéré et les systèmes locaux. C'est la raison pour laquelle on considère généralement que cette intégration est bien adaptée lorsqu'il n'existe qu'un petit nombre de sources à intégrer.

En termes de fonctionnalités, les systèmes fédérés doivent en principe permettre d'accéder aux données en lecture et écriture à partir du schéma fédéré [Busse et al. 1999]. Des mises à jour peuvent être propagées dans les sources à partir de la base unifiée virtuelle. Cette capacité de mise à jour est une des caractéristiques qui distingue les systèmes fédérés des systèmes fondés sur la médiation. Ces derniers sont principalement conçus pour l'interrogation. Nous les présentons dans la partie suivante.

A.3.1.3 SYSTEMES DE MEDIATION

L'approche d'intégration par médiation constitue sans doute aujourd'hui la solution la plus courante pour relier différentes sources qui cette fois, ne correspondent pas nécessairement à des bases de données. Le notion de médiateur a été initialement proposé par [Wiederhold 1992]. Il définit un médiateur comme suit : « *A mediator is a software module that exploits encoded knowledge about some sets*

or subsets of data to create information for a higher layer of applications ». Un médiateur doit être vu comme une couche logicielle permettant d'accéder de manière transparente pour l'utilisateur à différentes ressources (BD, fichiers) réparties et hétérogènes. Pour cet accès, le médiateur exploite des connaissances (métadonnées) qui sont utiles à différents services (interrogation, localisation des ressources notamment).

Les systèmes de médiation sont assez proches des systèmes fédérés. Ce qui les distingue concerne notamment le type d'accès (accès en lecture pour la médiation), le type de données qu'il est possible d'intégrer (structurée, semi-structurée ou non structurée pour la médiation) et leur capacité d'évolution (en générale plus grande pour la médiation car la conception suit souvent une approche descendante, nous y reviendrons) [Busse et al. 1999].

L'approche par médiation est fondée sur la définition de *vues* [Rousset et al. 2002]. Les données ne sont pas stockées dans le système de médiation mais résident dans leur source d'origine (comme pour les systèmes fédérés). L'utilisateur a une vision unifiée des données sources : l'interrogation se fait par l'intermédiaire d'un *schéma global*. Il n'a pas connaissance des schémas locaux.

L'architecture générale d'un système de médiation est présentée en figure 9. Une requête globale est posée via le schéma global et celle-ci est ensuite décomposée en sous-requêtes, traduites pour être exécutées sur les différentes sources concernées. Le *médiateur* est chargé de localiser les données pertinentes pour répondre à la requête (en utilisant les métadonnées). L'interrogation effective des sources se fait par des *adaptateurs* (ou « *wrappers* ») qui constituent une interface d'accès aux différentes sources. Ces adaptateurs traduisent les sous-requêtes exprimées dans le langage de requête spécifique de chaque source. Les résultats sont ensuite renvoyés au médiateur qui se charge de les intégrer avant de les présenter à l'utilisateur.

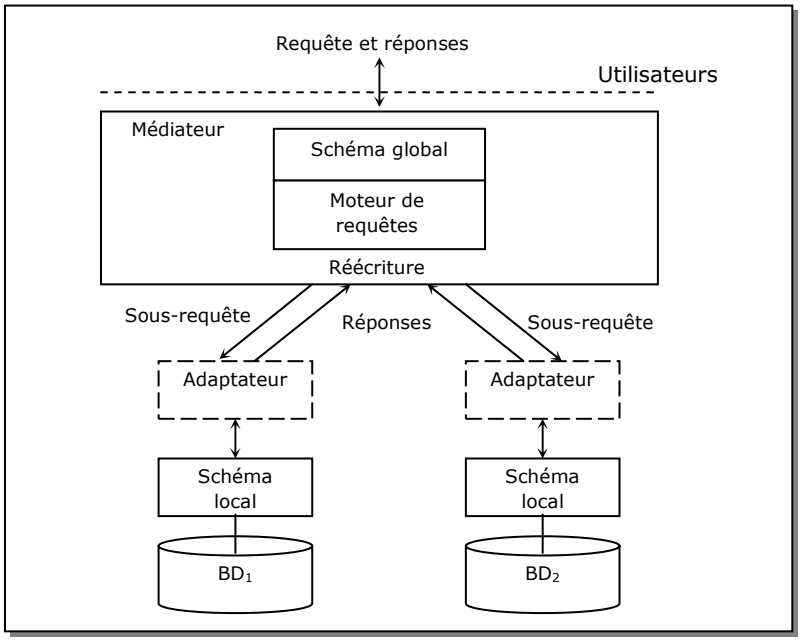


***Figure 9**. Architecture des systèmes de médiation (D'après [Rousset et al. 2002])*

Pour faire l'analogie avec l'architecture des systèmes fédérés, on peut considérer que le schéma global du médiateur correspond au schéma fédéré et que l'adaptateur inclut les schémas d'export et les schémas pivots.

Les médiateurs se distinguent les uns des autres par la façon dont est établie la correspondance (« *mapping* ») entre le schéma global et les schémas locaux pour traduire la requête de l'utilisateur. Deux approches différentes existent : l'approche *Global as View* (GaV) et l'approche *Local as Views* (LaV) [Rousset et al. 2002].

La première (GaV) est à mettre en relation avec les systèmes fédérés dont elle est issue. Elle consiste à définir le schéma global à partir des schémas source à intégrer. Ce schéma global est définit comme une vue sur les schémas locaux. La *réécriture* des requêtes posée sur le schéma global est simple dans ce cas : il suffit de remplacer les vues par leur définition dans les schémas source. On dit qu'on procède au *dépliement* de la requête [Rousset et al. 2002]. Une fois la requête dépliée, elle peut être évaluée de façon classique sur les extensions des sources de données. La reformulation par dépliement est le principal avantage de cette approche. Par contre, elle n'est pas bien adaptée à l'ajout de nouvelles sources car cet ajout peut impliquer une modification globale du schéma unifié (c'est ce qui explique la faible capacité d'évolution des systèmes fédérés).

Il existe plusieurs systèmes de médiation fondés sur l'approche GaV, notamment *TSIMMIS* dont le schéma global repose sur le langage orienté-objet OEM [Garcia-Molina et al. 1997], *DISCO* qui utilise le standard ODMG [Tomasic et al. 1998] et *GARLIC* qui vise à intégrer des informations multimédias [Hass et al. 1997].

La seconde approche (LaV) est l'inverse de la première. Elle consiste à définir les schémas locaux comme des vues sur le schéma global. De cette façon, l'ajout de nouvelles sources (ou la suppression) est cette fois bien supporté puisque cela n'implique pas de modification au niveau du schéma global : seules des vues doivent être ajoutées ou supprimées et les relations seront ainsi définies avec le schéma global (pour les éléments en correspondances). Cette approche est donc beaucoup plus flexible. En revanche, la réécriture des requêtes est un problème particulièrement délicat. Cette réécriture doit se faire en termes de vues qu'il faut ensuite exécuter pour obtenir les résultats d'une requête. Le problème essentiel est d'assurer l'équivalence entre une requête et sa reformulation [Solar et Doucet 2002].

On peut citer plusieurs contributions qui suivent cette approche, notamment les systèmes *Information Manifold* [Kirk et al. 1995], *SIMS* [Arens et al. 1996] et *PICSEL* [Rousset et al. 2002]. *Information Manifold* est fondé sur un schéma global à base de règles (extensions de DATALOG). *SIMS* utilise une logique de description[7]. *PICSEL* exprime son schéma global dans le langage CARIN. Ce dernier combine un langage de règles et la logique de description ALN.

Il faut noter que la médiation ne s'adresse pas uniquement à des bases de données. De nombreux médiateurs permettent d'intégrer des données semi-structurées ayant pour format XML. C'est le cas notamment de *Xylème* dont le but est de construire un entrepôt de données dynamique regroupant des documents XML du Web [Delobel et al. 2003]. L'utilisateur interroge les sources de données à travers une description abstraite des documents. Cette description correspond à un ensemble de *DTD abstraites* (le schéma global) qui structure différents domaines sous forme d'arbres.

[7] Les logiques de description ont été introduites dans le domaine de la représentation des connaissances [Kayser 1997]. Elles font partie de la famille des langages de représentation par réseaux. Le langage des logiques de description est défini par des *concepts* (qu'on peut assimiler aux classes des approches O.O.) et des *rôles* (relations entre les concepts), auquel on associe une sémantique. La propriété de *subsomption* que possède ces logiques permet d'organiser les concepts et les rôles en hiérarchies.

On peut trouver davantage d'informations sur les systèmes de médiation dans [Boucelma et Lacroix 2001, Rousset et al. 2002, Solar et Doucet 2002]. Pour les bases de données spatiales, le système ISIS a notamment été proposé [Leclercq et al. 1999].

A.3.1.4 ENTREPOTS DE DONNEES

Un entrepôt de données (« *Data Warehouse* ») se définit comme « *une collection de données intégrées, orientées sujet, non volatiles, historisées, résumées et disponibles pour l'interrogation et l'analyse* » [Inmon 1996]. Les entrepôts de données sont conçus dans un but bien particulier : rassembler l'ensemble des informations d'une entreprise dans une base unique, pour faciliter l'analyse et la prise de décision rapide.

Une illustration de l'architecture de ces systèmes est présentée en figure 10. Les données stockées dans l'entrepôt proviennent de sources multiples souvent hétérogènes (BD de production et sources externes). Après leur extraction et leur transformation, elles sont stockées et organisées dans l'entrepôt par sujet (clients, produits, ventes,…). Pour des applications particulières, il est possible de regrouper certaines d'entre elles dans des « magasins ». Ces données peuvent être ensuite manipulées par un ensemble d'outils de fouille de données (« *Data Mininig* »), d'analyse en ligne (« *On-Line Analytical Processing* ») et d'interrogation (*requêteurs*) [Doucet et Gançarski 2001].

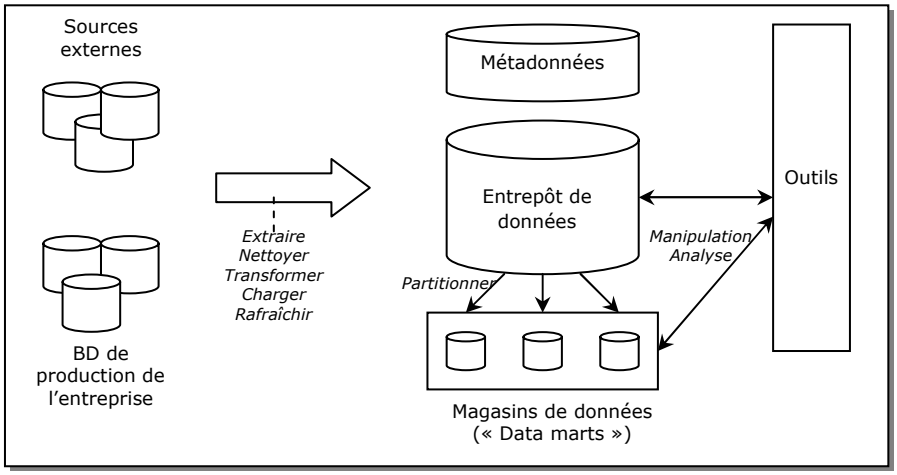


***Figure 10**. Architecture d'un entrepôt de données (D'après [Doucet et Gançarski 2001])*

Il existe donc une phase d'intégration lors la conception d'entrepôts mais cette intégration est différente. Toutes les approches que nous avons présentées jusqu'à présent proposent une intégration *virtuelle* : les données restent dans leur source d'origine. Dans le cas des entrepôts de données, l'intégration est *matérialisée* : les données sont regroupées dans une base unique. Cela signifie qu'il y a une duplication des données dans l'entrepôt (par copie sélective) et qu'il n'est plus nécessaire d'accéder aux sources initiales pour répondre à une requête.

Il existe également un schéma global dans un entrepôt de données mais celui-ci présente des caractéristiques différentes de ceux mentionnés précédemment. Selon, [Metais et Sèdes 2002], le schéma global n'est pas figé mais est amené à évoluer régulièrement. Un entrepôt de données est en effet dynamique et de nouvelles

sources sont susceptibles d'être intégrées fréquemment, des données stockées peuvent être réorganisées (agrégation, ajout, suppression d'attributs, …), etc. Par ailleurs, l'utilisateur n'a généralement pas une compréhension globale de ce schéma [Metais et Sèdes 2002]. Certaines données peuvent provenir de sources parfois étrangères à l'entreprise qu'il ne connaît pas nécessairement.

Certains auteurs préconisent de suivre une approche *Local-as-view* (les sources sont des vues sur le schéma global) pour l'intégration des données dans un entrepôt car toute l'information présente dans les différentes sources n'est pas nécessaire [Calvanese et al. 2001]. Il est donc préférable de définir d'abord le schéma global de l'entrepôt qui reflète l'information nécessaire pour l'analyse et la prise de décision, et puis d'établir la correspondance avec les sources (approche descendante), plutôt que de se concentrer sur les sources, avant de produire le schéma global (approche ascendante). On trouvera davantage d'informations au sujet des entrepôts de données et leur conception dans [Doucet et Gançarski 2001]. C'est une architecture qui commence à apparaître pour les BD géographiques [Miquel et al. 2002].

A.3.1.5 SYNTHESE

Nous venons d'exposer brièvement différents systèmes intégrés, c'est-à-dire des architectures qu'il est possible de mettre en placer pour faire coopérer de manière relativement transparente (selon le niveau d'intégration) des bases de données initialement indépendantes. Nous avons ainsi vu les systèmes multibases, les systèmes fédérés et les systèmes de médiation. Nous avons également présenté les entrepôts de données car il existe une phase d'intégration de données pour les constituer.

Dans cette thèse, nous n'irons pas jusqu'au développement d'une architecture d'un système intégré mais le problème que nous traitons dans ce travail s'inscrit dans un processus d'intégration destiné à concevoir un système fédéré de bases de données géographiques. La fédération est souhaitée car l'intégration dans notre contexte n'est pas uniquement orientée vers l'interrogation de plusieurs sources. Les instituts cartographiques souhaitent relier leurs bases de données pour propager les mises à jour et les rendre cohérente entre elles. De ce fait, l'intégration doit être d'un niveau assez élevé (une représentation unifiée doit être fournie) et l'accès aux données doit être possible en lecture et écriture. Par ailleurs, les bases source doivent garder une certaine autonomie car elles correspondent chacune à une gamme de produits spécifiques. Ces produits doivent continuer d'exister indépendamment de la fédération. Ajoutons encore que la fédération est bien adaptée car l'intégration doit être statique (les correspondances doivent être prédéfinies). Elle ne peut pas se faire dynamiquement car les processus permettant de relier les données (processus d'appariement) sont trop complexes. Les relations entre les données géométriques ne pourraient pas toujours être calculées à la volée. Nous y reviendrons ultérieurement.

A.3.2 PROCESSUS D'INTEGRATION

L'approche fédérée étant adoptée, il convient maintenant d'exposer la manière d'aboutir à un tel système. Quelle est la démarche à suivre pour créer le schéma fédéré ? Comment garantir la cohérence entre les différentes sources ?

Pour répondre à ces interrogations, nous avons décidé de présenter le processus d'intégration proposé par [Parent et Spaccapietra 1996, Parent et Spaccapietra 2001]. Il s'agit d'un processus destiné à la conception d'un système fédéré. Sa présentation

nous permettra de bien préciser l'étape que nous traitons dans l'intégration. Nous pourrons également mettre en évidence les solutions déjà proposées pour répondre à notre problème dans le cadre des bases de données classiques. Nous exposerons par la suite les particularités de l'intégration des bases de données spatiales (A.4.).

Il convient d'abord d'indiquer qu'il existe plusieurs méthodologies permettant l'intégration de bases de données classiques. Une bonne revue des contributions sur le sujet peut être trouvée dans [Batini et al. 1986, Lawrence 2001, Rahm et Bernstein 2001]. Si nous présentons le processus de [Parent et Spaccapietra 1996, Parent et Spaccapietra 2001], c'est notamment parce qu'il a été étendu par [Devogele 1997] pour l'intégration de bases de données géographiques vectorielles.

Le processus d'intégration est ainsi décomposé en trois phases distinctes :

- La *pré-intégration*. Cette phase vise à préparer l'intégration des schémas en les rendant plus homogènes. Elle consiste principalement à traduire les schémas initiaux dans un modèle de données commun (réduction de l'hétérogénéité *syntaxique*). Elle s'attache également à enrichir leur sémantique.
- L'*identification des correspondances*. Durant cette phase, les correspondances entre les éléments des schémas source sont détectées et formalisées de même que les différents conflits.
- L'*intégration*. Cette phase finale produit le schéma intégré et fournit les règles de traduction permettant de passer des schémas source au schéma intégré et inversement (« *mapping* »).

Ce processus est illustré en figure 11. Nous exposons plus en détails chacune de ces étapes dans les sections suivantes en adoptant la trame de [Parent et Spaccapietra 2001].

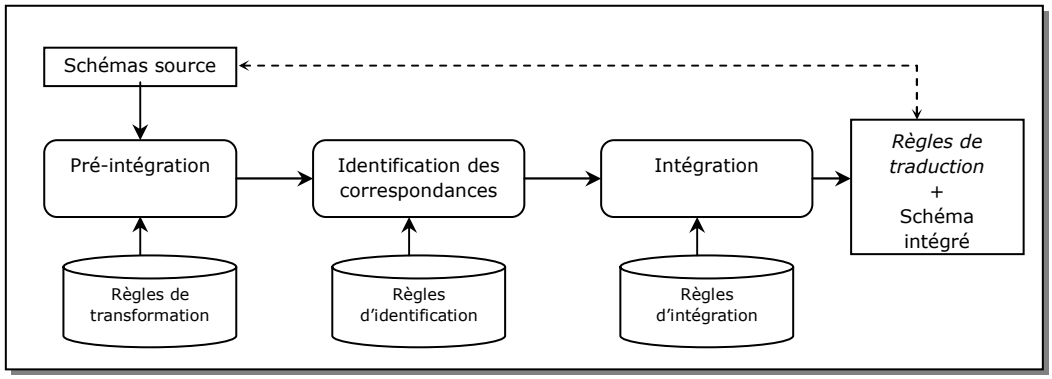


***Figure 11**. Les étapes du processus d'intégration des bases de données classiques permettant d'aboutir à système fédéré (D'après [Parent et Spaccapietra 1996]).*

A.3.2.1 PRE-INTEGRATION

L'étape de pré-intégration a pour objectif de rendre les schémas des bases à intégrer plus homogènes. On prépare l'intégration en s'efforçant de réduire l'hétérogénéité entre les schémas pour les faire tendre vers un même niveau de description. Cette préparation est nécessaire pour plusieurs raisons. Tout d'abord, les modèles utilisés peuvent être différents (relationnel, orienté-objet). Ensuite, les schémas peuvent suivre des logiques différentes, dues aux différentes approches de modélisation des concepteurs. Enfin, la sémantique exprimée dans les schémas est

souvent insuffisante pour l'intégration (la signification des classes n'apparaît pas dans les schémas par exemple).

Il convient donc d'une part de réduire l'hétérogénéité *syntaxique*. Cela consiste généralement à traduire les schémas initiaux dans un modèle de données commun ou canonique. Il convient d'autre part de réduire l'hétérogénéité *sémantique*. A ce niveau, cela se traduit par le recueil d'informations annexes aux schémas source (des métadonnées). Enfin, il s'agit de réduire l'hétérogénéité *structurelle* (différence de structure entre les schémas). Des règles de normalisation peuvent être imposées pour diminuer les différences de modélisation entre les schémas.

CHOIX D'UN MODELE COMMUN ET TRADUCTION DES SCHEMAS SOURCE

On réduit généralement l'hétérogénéité syntaxique des modèles source en traduisant les schémas initiaux dans un modèle commun. Cela suppose d'une part de choisir ce modèle commun : il peut s'agir du modèle entité-association (E/A) [Beynon-Davies et al. 1997, McBrien et Poulovassilis 1998], orienté-objet (O.O.) [Hammer et McLeod 1993], relationnel. Cela implique d'autre part de définir des règles permettant de transformer les schémas initiaux vers les schémas pivots. Autrement dit, des mécanismes doivent être conçus pour traduire des concepts exprimés dans un modèle (relationnel par exemple) vers un autre modèle (orienté-objet).

Le choix du modèle commun reste encore aujourd'hui une question ouverte et celle-ci a souvent été débattue [Saltor et al. 1991]. Le modèle commun doit être suffisamment expressif pour pouvoir représenter les concepts présents dans les schémas initiaux et supporter des informations supplémentaires récoltées lors de l'enrichissement sémantique. Le choix du modèle commun induit généralement un compromis entre la richesse d'expression des schémas et la simplicité de leur intégration. Le modèle canonique le plus populaire aujourd'hui est l'orienté-objet [Conrad et al. 1997] mais la communauté des bases de données s'intéresse aussi de plus en plus à XML (« *eXtensible Markup Language* ») [Bellahsène et Baril 2001].

ENRICHISSEMENT SEMANTIQUE

L'autre étape importante lors de la pré-intégration est l'enrichissement sémantique des schémas. Lorsqu'on modélise un domaine à l'aide de schémas, il n'est pas possible de représenter toute la richesse sémantique du domaine considéré. Le schéma ne reflète qu'une partie de celui-ci. Cet effet réducteur est lié au modèle utilisé mais aussi au processus de modélisation lui-même : la modélisation a pour objectif de fournir une vision globale du système, volontairement simplifiée pour faciliter la compréhension. Certains éléments des schémas peuvent également porter un sens particulier pour les administrateurs de la base mais être incompréhensibles pour un personne étrangère à la base, en raison de facteurs techniques ou de conventions internes : le concepteur peut utiliser sa propre terminologie, les noms des attributs peuvent être limités à quelques caractères, etc. [Bonjour et al. 1994]. Pour faciliter la mise en correspondance des schémas et réduire les ambiguïtés d'interprétation, il est donc utile de les enrichir.

L'enrichissement porte à la fois sur le schéma lui-même et sur la signification des éléments du schéma [Parent et Spaccapietra 1996]. Comprendre à quoi correspondent les éléments des schémas dans la réalité est essentiel si l'on veut réaliser une intégration sémantique et ne pas se limiter à réduire l'hétérogénéité structurelle (différence de modélisation des concepts). Pour interpréter les éléments des schémas, de nombreuses approches se fondent sur l'utilisation de *métadonnées* [Siegel and

Madnick 1991, Heiler et al. 1996, Kashyap et Sheth 1996]. Leur description peut prendre diverses formes. Il peut s'agir de dictionnaires de mots et de thesaurus qui fixent le vocabulaire des termes utilisés dans les schémas. Il peut s'agir également de bases de concepts, contenant l'ensemble des concepts du domaine dont le schéma est issu, avec des informations associées à chacun de ces concepts (informations terminologiques et linguistiques notamment) [Bonjour et al. 1994]. Les métadonnées peuvent être aussi stockées dans des entrepôts (« *Repository* ») [Heiler et al. 1996]. La tendance actuelle s'oriente vers l'utilisation d'une *ontologie* [Wache et al. 2001]. Nous reviendrons sur le rôle que peut jouer une ontologie pour l'intégration sémantique lorsque nous exposerons les contributions concernant les bases de données géographiques.

DEFINITION DE REGLES DE MODELISATION ET DE NORMALISATION

La diminution des différences de modélisation est la dernière tâche à accomplir durant la phase de pré-intégration [Parent et Spaccapietra 2001]. Elle consiste essentiellement à imposer aux schémas des règles de normalisation. Ces règles imposent par exemple d'adopter une convention pour la dénomination de termes ou d'utiliser des patrons prédéfinis pour la modélisation [Wohed 2000, Filho et al. 2002].

A.3.2.2 IDENTIFICATION DES CORRESPONDANCES

La seconde étape du processus d'intégration concerne l'identification des correspondances et la détection des conflits entre les éléments des schémas. Il convient de préciser quels sont les éléments qui expriment les mêmes phénomènes dans le monde réel et comment ils se correspondent dans les bases (par exemple, il faut préciser que la classe « Livre » dans une base correspond à la classe « Ouvrage » dans la seconde). Un langage peut être utilisé à cette fin. Celui proposé par [Spaccapietra et al. 1992] s'appuie sur la notion d'*Assertion de Correspondance Inter-schémas* (ACI). Les ACI définissent les correspondances en intention, c'est-à-dire au niveau des types (non au niveau des instances). Ce langage semble bien adapté aux bases de données géographiques. Il a d'ailleurs été retenu et étendu par Thomas Devogele dans le cadre de sa thèse [Devogele 1997, Devogele et al. 1998]. Il a également été utilisé lors du projet mené par la société *EADS Matra S&I* en collaboration avec le laboratoire COGIT (projet « Serveur Géographique Multi-Echelles »), pour le compte de la DGA (Délégation Générale pour l'Armement) [Badard et al. 2001]. Nous décrivons ce langage ci-dessous. Nous l'utiliserons dans le cadre des expérimentations (cf. E.3.4.1).

DECLARATION DES ASSERTIONS DE CORRESPONDANCE INTER-SCHEMAS

La forme générale d'une ACI est la suivante :

ACI BD_1.$Element_1$ <Rel> BD_2.$Element_2$
AIC (identifiants correspondants)
AAC (attributs correspondants)

Une ACI permet d'abord de préciser quels sont les éléments en correspondance qui représentent le même phénomène du monde réel. Ces éléments (*$Element_1$* et *$Element_2$*) peuvent correspondre à l'ensemble des objets d'une classe, ou à une portion de cet ensemble, ou encore, à un ensemble d'objets de plusieurs classes. Par exemple, on peut déclarer des ACI du type :

ACI_1 BD_1.Bâtiment <Rel> BD_2.Bâtiment

ACI_2 BD_1.Chemin <Rel> $SELECTION_{(BD2.Route.Type = 'Chemin')}BD_2$.Route

ACI_3 BD_1.(Route,Chemin) <Rel> BD_2.Route

On précise donc dans l'ACI_1 que les éléments de la classe « Bâtiment » dans la BD_1 correspondent aux éléments de la classe du même nom dans la BD_2. L'ACI_2 indique que les éléments de la classe « Chemin » dans la BD_1 correspondent aux éléments de la classe « Route » dans la BD_2 lorsque l'attribut « Type » prend la valeur 'Chemin'. Enfin, on exprime dans l'ACI_3 que les éléments des classes « Route » et « Chemin » de la BD_1 correspondent aux éléments de la classe « Route » dans la BD_2.

Pour déclarer comment les éléments sont en correspondance, on décrit les relations entre ensembles (<Rel>) à l'aide des opérateurs ensemblistes : équivalence ($\equiv$), disjonction ($\neq$), inclusion stricte ($\subset$), inclusion ($\subseteq$), intersection ($\cap$), contenance stricte ($\supset$), contenance ($\supseteq$). Autrement dit, si on déclare une ACI du type : **ACI_1** BD_1.Bâtiment $\supseteq$ BD_2.Bâtiment, cela signifie que les classes bâtiments des deux BD sont à mettre en correspondance car elles décrivent les mêmes éléments du monde réel, et que l'ensemble des bâtiments présents dans la BD_2 est aussi représenté dans la BD_1 (la réciproque n'est pas vrai).

Les correspondances au niveau des attributs doivent également être déclarées. Elles permettent d'identifier l'information redondante. Une clause particulière est définie à cet effet : *Avec Attributs Correspondants* (AAC). La forme la plus simple d'une AAC est la suivante :

AAC BD_1.Bâtiment.$Attribut_1$ <Rel> BD_2.Bâtiment.$Attribut_{1'}$

Les correspondances entre les attributs peuvent être plus complexes. Il est possible que l'information contenue dans un attribut de la première BD corresponde à une combinaison d'informations contenues dans plusieurs attributs de la deuxième BD. La forme la plus générale d'une AAC est la suivante :

AAC $f(BD_1.Element_1.Attribut_1)$ <Rel> $g(BD_2.Element_{1'}.Attribut_{1'})$

Des fonctions de correspondances peuvent donc être définies. Il existe des fonctions standards comme la somme, la moyenne, le minimum, le maximum, et des fonctions plus évoluées comme la fonction de transfert pour les attributs énumérés. Nous donnons un exemple d'AAC utilisant une fonction ci-dessous :

AAC BD_1.Route.NbVoie <Rel> SOMME(BD_2.Route.NbVoie_sensDirect,
BD_2.Route.NbVoie_sensInverse)

Il existe enfin une clause particulière qui permet d'exprimer comment les instances en correspondances sont identifiées. Autrement dit, cette clause précise sur quels attributs s'appuyer pour mettre en relation les instances homologues (instances représentant le même phénomène du monde réel). Il s'agit de la clause : *Avec Identifiants Correspondants* (AIC). Dans le cas le plus simple, cette clause est exprimée de la manière suivante :

AIC BD_1.Route.Numéro = BD_2.Route.Numéro

Cette clause indique qu'il est possible de s'appuyer sur l'attribut « Numéro » des deux BD pour retrouver les paires d'objets homologues.

Il faut noter que pour les bases de données géographiques, il est rarement possible d'utiliser cette notion d'identifiant commun. C'est ce qui explique l'utilisation de l'appariement géométrique pour relier les instances [Sester et al. 1998]. On considère que si deux objets de même nature occupent la même place, ils

correspondent probablement au même phénomène du monde réel. Une clause particulière remplaçant l'AIC a été définie [Devogele 1997] : *Appariement Géométrique de Données* (AGD). Lorsque l'appariement se limite à des fonctions géométriques simples, on précise la fonction dans la clause :

AGD BD_1.Geom_Point = INSIDE(BUFFER[(BD_2.Geom_Surface),5m])

Cette assertion précise qu'un point dans la BD_1 est apparié à une surface dans la BD_2 s'il est inclus dans celle-ci (*INSIDE*). Cette surface est préalablement dilatée avec une fonction *BUFFER* de rayon égal à 5 mètres (création d'une zone tampon) pour tenir des différences de position des objets (lié à la résolution des bases et à leur qualité respective).

Souvent, l'appariement géométrique fait appel à un ensemble de méthodes complexes qu'il est nécessaire d'enchaîner. Dans ce cas, il n'est plus possible de déclarer l'AGD. Cette clause est alors remplacée par le processus d'appariement lui-même.

Ce langage fondé sur les ACI permet donc de déclarer les correspondances entre les éléments des schémas, mais il exprime aussi les *conflits* entre ceux-ci. En général, les classes à intégrer n'ont pas la même représentation et leurs populations ne sont pas équivalentes. Il existe donc des différences entre celles-ci (en termes de définition et de structure), et ces différences sont appelées *conflits*. Par exemple, dans l'**ACI_1** BD_1.Bâtiment $\supseteq$ BD_2.Bâtiment, on indique un *conflit de généralisation* (ou classification selon les auteurs) entre les classes « Bâtiment » des deux BD. Il ne s'agit pas d'une équivalence.

Il est important de préciser que le terme *conflit* ne veut pas dire *incohérence*. Il porte le sens de *différence*. Ces conflits sont exprimés à travers les ACI, au niveau des schémas (conflits « normaux »). On les retrouve bien sûr dans les données avec en plus des conflits « anormaux » (différences correspondant à des *incohérences* résultant d'erreurs). Nous reviendrons sur ces termes et leur définition dans le chapitre B. Nous présentons ci-dessous différentes catégories de conflits que l'on est susceptible de rencontrer lors de l'intégration des schémas : les conflits d'intégration.

CONFLITS D'INTEGRATION

Les conflits d'intégration ont beaucoup été étudiés dans la littérature. Il existe plusieurs taxonomies à ce sujet [Kim et Seo 1991, Spaccapietra et Parent 1991, Sheth et Kashyap 1992, Colomb 1997, Ram et Ramesh 1999]. Généralement, on distingue trois grandes catégories de conflits :

- Les *conflits syntaxiques;*
- Les *conflits de structure;*
- Les *conflits sémantiques*;

Les *conflits syntaxiques* concernent les différences au niveau des modèles de données (relationnel, orienté-objet,…). Nous en avons déjà discuté lors de la pré-intégration. Nous n'y revenons pas ici.

Les *conflits de structure* se réfèrent aux différences d'organisation des données dans les schémas, indépendamment du modèle. Par exemple, une classe dans un schéma peut être représentée par un attribut dans l'autre schéma ou encore, une classe dans un schéma peut correspondre à deux classes dans l'autre schéma ou à une portion des éléments d'une classe. On peut se reporter à [Kim et al. 1993] pour davantage de détails.

Enfin, les *conflits sémantiques* expriment les différences de signification et de vocabulaire des concepts modélisés. Les différences peuvent se traduire notamment par des noms identiques pour des concepts sémantiquement différents (homonymes), ou des noms différents pour des concepts sémantiquement équivalents (synonymes).

INCOHERENCES ENTRE DONNEES

La déclaration des conflits au niveau des schémas et leur résolution lors de l'étape d'intégration proprement dite (phase suivante) n'empêchent pas l'existence d'incohérences au niveau des données. Comme nous l'avons déjà indiqué, il est possible que pour deux instances en correspondance présentant un attribut en commun par exemple, certaines valeurs de cet attribut soient incohérentes. Il s'agit dans ce cas d'un conflit anormal qui résulte le plus souvent d'une erreur de saisie dans les données.

Les travaux au sujet de la gestion des incohérences entre les données sont moins nombreux. Pour résoudre ce type d'incompatibilité, certains auteurs proposent d'utiliser des fonctions d'agrégation [Dayal 1983]. Il peut s'agir d'une moyenne sur les valeurs numériques en conflit par exemple. Ainsi, si la valeur de l'attribut « salaire » est différente entre deux instances homologues de deux bases de données (s_1 = 2000 et s_2 = 1800), une solution pour résoudre cette incohérence est de prendre la valeur moyenne dans la base intégrée (s_i = 1900). Dans ce cas, aucune hypothèse sur les sources n'est posée. C'est une solution qui n'est valable que pour les attributs numériques. D'autres auteurs suggèrent d'associer une probabilité à chaque valeur intervenant dans un conflit. Les différentes valeurs possibles sont ainsi fournies avec une indication sur la valeur la plus probable [Tseng et al. 1992]. L'approche de [Lim et al. 1994] prend également en compte l'incertitude sur les valeurs mais non plus dans un cadre probabiliste classique mais en termes de degré de plausibilité et de crédibilité, en reprenant le modèle de *Dempster-Shafer.* [Anokhin et Motro 2001] proposent enfin une stratégie de résolution qui suppose l'utilisation de *propriétés* sur les attributs. Cette stratégie se décompose en deux phases. La première est une étape de filtrage des valeurs en conflit en utilisant des *fonctions d'élimination*. Celles-ci sont appliquées sur les valeurs des attributs ou sur des connaissances associées (les propriétés). Les fonctions peuvent être très simples (on élimine le *min* ou le *max* par exemple). La seconde phase consiste à appliquer des fonctions qui fusionnent les valeurs (s'il reste des valeurs en conflit). La stratégie de résolution est guidée par l'expert qui choisit la ou les fonctions à appliquer. L'utilisation de connaissances externes pour répondre à ce problème d'incompatibilité est également suivie par [Wang et Zhang 1996].

Nous exposerons plus en détails les conflits spécifiques aux bases de données géographiques dans la section A.4.2.1.

RECHERCHE DES CORRESPONDANCES

Nous avons jusqu'à présent discuté de la déclaration des correspondances entre les éléments des schémas et présenté les conflits susceptibles d'apparaître. Il reste maintenant à expliquer comment ces correspondances peuvent être recherchées.

Plusieurs méthodes et outils existent pour identifier les éléments équivalents dans les bases à intégrer. Cette recherche est généralement semi-automatique. L'intervention humaine peut être réduite mais elle reste toujours nécessaire, ne fût-ce que pour valider les correspondances calculées.

[Rahm et Bernstein 2001] proposent une classification des techniques existantes pour apparier automatiquement les éléments des schémas. Celle-ci est fondée sur les critères suivants :

- le niveau d'analyse : certaines approches d'appariement se focalisent sur les éléments des schémas tandis que d'autres utilisent les informations des schémas et des données.
- le niveau de granularité : l'appariement peut se faire élément par élément ou peut se référer à une combinaison d'éléments (une structure plus importante du schéma).
- l'approche suivie pour mettre en correspondance les éléments : on distingue les approches linguistiques qui sont fondées sur la comparaison des chaînes de caractères, des approches à base de contraintes qui utilisent par exemple les informations concernant les domaines de valeurs des attributs, les types de données ou les clés primaires.
- la cardinalité du lien d'appariement : cette caractéristique est étroitement liée au niveau de granularité auquel peut travailler l'outil d'appariement. Il est possible d'obtenir des relations du type 0:1, 1:0, 1:1, 1:n, n:1, n:m entre les éléments des schémas source.
- l'utilisation d'informations auxiliaires : la plupart des outils d'appariement s'appuient sur des informations annexes pour apparier les éléments. Cela peut être des dictionnaires de termes par exemple ou des connaissances fournies par l'administrateur. Ces informations sont particulièrement importantes pour permettre une intégration sémantique. La phase d'enrichissement lors de la pré-intégration est notamment réalisée à cette fin.

Les approches les plus classiques pour déterminer les correspondances sont fondées sur l'utilisation de mesures de similarités entre les éléments, que ce soit au niveau des noms, des propriétés, des relations, des méthodes ou des domaines de valeurs [Madhavan et al. 2001]. Ces mesures peuvent provenir du domaine de la recherche documentaire (« *information retrieval* »), comme celle utilisée par [Cohen 1998] ou [Tejada et al. 2001]. Si le coefficient de similitude calculé dépasse un seuil fixé, on considère qu'il y a correspondance.

Des approches plus récentes mettent en œuvre des techniques issues de l'intelligence artificielle (IA). [Li et Clifton 2000] proposent ainsi d'utiliser les réseaux de neurones pour étudier l'équivalence entre attributs. D'autres auteurs utilisent les réseaux Bayesian [Berlin et Motro 2002]. Le système *Automatch* permet ainsi d'apparier les éléments des schémas à l'aide d'une base de connaissances sur les attributs (appelé « *attribute dictionnary* »). Celle-ci est construite par apprentissage automatique à l'aide d'exemples de schémas mis en correspondance par un expert du domaine. Des relations probabilistes entre les éléments des schémas sont donc apprises. La base de connaissances enregistre ces informations pour chaque attribut et celles-ci sont exploitées pour apparier de nouveaux schémas « clients ». [Fan et al. 2001] s'intéressent à la découverte de règles de conversion pour passer de la valeur d'un attribut appartenant à une première base vers la valeur d'un attribut d'une seconde base. Le prototype *DIRECT* a été développé à cet effet. D'autres approches utilisant les arbres de décision [Tejada et al. 2001] ou une combinaison de *classifieurs* ont également été testées [Doan et al. 2003].

Les techniques d'intelligence artificielle ont souvent été utilisées dans le cadre de l'intégration [Levy 1998], que ce soit pour rechercher les correspondances (le

problème du « *schema matching* »), pour représenter et raisonner sur les connaissances relatives aux schémas, ou expliciter et manipuler automatiquement les métadonnées associées. Plusieurs approches utilisent ainsi l'apprentissage (supervisé ou non) et des systèmes à base de règles. Les logiques de description sont souvent utilisées comme langage de représentation. Elles peuvent servir à décrire les éléments des schémas, à représenter les ontologies et à raisonner sur ces connaissances [Calvanese et al. 1998, Hakimpour et Geppert 2001]. Les mécanismes d'inférence peuvent être mis en œuvre pour construire des hiérarchies de concepts et fournir ainsi des relations entre les classes des schémas à intégrer [Savasere et al. 1991].

Enfin, en plus de ces méthodes qui permettent d'automatiser la recherche des correspondances, il existe aussi quelques ateliers de génie logiciel (AGL) qui aident à les déclarer (figure 12). On peut se référer à [Beynon-Davies et al. 1997] pour une description d'un tel outil.

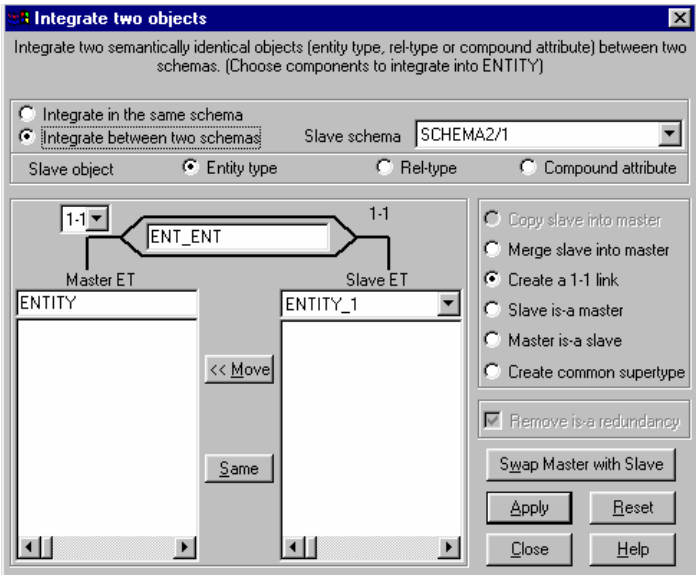


***Figure 12**. Interface de l'AGL DB-MAIN permettant de déclarer des correspondances entre schémas [DB-Main 2004].*

A.3.2.3 INTEGRATION

La dernière étape du processus est l'intégration proprement dite [Parent et Spaccapietra 2001]. Elle se caractérise par :

- La résolution des conflits décrits dans les ACI et la réconciliation des données en cas d'incohérence ;
- L'élaboration du schéma intégré ;
- La production des règles de traduction qui permettent de passer des schémas source au schéma intégré.

La méthode de résolution des conflits et l'élaboration du schéma intégré doivent tenir compte de l'objectif de l'intégration. En fonction de cet objectif, les solutions pour résoudre les conflits peuvent différer. Ainsi, on peut souhaiter produire un schéma intégré le plus complet possible, en préservant l'existence des classes source. On peut aussi vouloir mettre l'accent sur la simplicité du schéma intégré, pour qu'il soit le plus compréhensible. Suivant le cas, on utilisera par exemple les relations de

généralisation/spécialisation dans le schéma intégré pour résoudre les conflits de généralisation, ou on procédera à une fusion des classes. Nous ne détaillons pas davantage les techniques permettant de résoudre les différents conflits dans cette partie. On peut se reporter notamment à [Larson et al. 1989, Kim et al. 1993, Hammer et McLeod 1993, Parent et Spaccapietra 1996] pour davantage d'informations à ce sujet.

La production du schéma intégré est guidé par une stratégie d'intégration. Celle-ci dépend du nombre de schémas à unifier et de leur complexité. [Batini et al. 1986] distinguent plusieurs stratégies (figure 13):

- Les stratégies binaires : seuls deux schémas sont intégrés à la fois. Différentes démarches incrémentales peuvent être adoptées pour intégrer plusieurs schémas;
- Les stratégies n-aires : tous les schémas sont intégrés en une seule passe.

La qualité du schéma intégré peut être évaluée en terme d'exhaustivité (pas de perte d'information), de clarté et de minimalité (pas de redondances) [Batini et al. 1986].

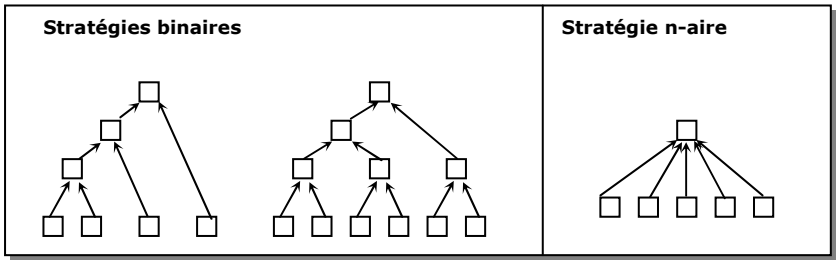


***Figure 13**. Stratégies d'intégration de schémas.*
(D'après [Batini et al. 1986])

Enfin, il est également nécessaire de définir les règles de traduction qui permettent de passer du schéma intégré aux schémas source, et inversement. Ces règles seront exploitées par le module de requête du système intégré. Celui-ci se chargera de décomposer les requêtes effectuées sur le schéma global en sous-requêtes adaptées aux schémas locaux pour extraire l'information demandée.

A.3.2.4 CONCLUSION

Nous venons de présenter un processus d'intégration de bases de données classiques destiné à concevoir un système fédéré. Nous avons vu qu'il existait un nombre important d'approches touchant différents problèmes de l'intégration. Dans la cadre de notre thèse, nous retenons que :

- De nombreuses approches sont fondées sur l'utilisation de métadonnées pour comprendre la sémantique des schémas et relier les éléments des schémas. Ces métadonnées peuvent prendre la forme de dictionnaires de données, de thesaurus, de bases de concepts ou d'ontologies. Dans cette thèse, nous utilisons aussi des métadonnées (les spécifications) pour comprendre le contenu des bases de données géographiques et évaluer la cohérence mais nous ne traitons que des données.
- Plusieurs approches ont été définies pour détecter et gérer les incohérences au niveau des données. Les solutions sont toutefois moins nombreuses que celles proposées pour résoudre les conflits au niveau des schémas. Certains

auteurs suggèrent de gérer les incompatibilités entre les données en associant une probabilité à chaque valeur intervenant dans un conflit. D'autres proposent d'agréger les valeurs sans faire d'hypothèse sur les données. Dans cette thèse, nous étudions aussi les conflits entre les données. Nous déterminons l'origine de chaque conflit entre les données (c'est-à-dire l'origine de chaque différence de représentation d'entités géographiques) en utilisant les spécifications des bases de données. Dans ce sens, la solution de [Anokhin et Motro 2001] fondée sur l'emploi de *propriétés* (connaissances) pour résoudre les conflits entre valeurs d'attributs semble être la plus proche de la nôtre.

- Plusieurs approches exploitent des techniques d'intelligence artificielle pour faciliter l'intégration des schémas et des données (par exemple, pour rechercher les correspondances automatiquement). Nous utilisons aussi ces techniques et en particulier l'apprentissage automatique supervisé pour extraire des connaissances des données.

Cette première partie touchant l'intégration des bases de données classiques va nous permettre à présent de situer les contributions apportées dans le monde des bases de données géographiques, de montrer les manques actuels et de mettre en évidence l'intérêt de notre travail de thèse.

A.4 INTEGRATION DES BASES DE DONNEES GEOGRAPHIQUES

Dans cette partie, nous présentons les travaux effectués sur l'intégration dans le cadre des BD géographiques. Dans un premier temps, nous discutons de la particularité du processus d'intégration et exposons les problèmes spécifiques induits par l'existence de données géométriques. La suite est consacrée à la présentation de l'état de la recherche dans ce domaine en situant les différentes contributions par rapport au processus d'intégration que nous venons d'exposer.

A.4.1 SPECIFICITE DE L'INTEGRATION DES BD GEOGRAPHIQUES

Les méthodologies d'intégration des bases de données traditionnelles peuvent être appliquées pour unifier les BD géographiques mais elles requièrent néanmoins une adaptation. L'existence d'une géométrie associée à chaque objet engendre des difficultés supplémentaires pour rechercher les correspondances et résoudre les conflits entre les schémas et les données. En plus des problèmes classiques d'hétérogénéité, il est nécessaire de prendre en compte les problèmes liés à la nature des données géométriques, leur précision, leurs différences d'abstraction, de représentation, de formats, etc. Nous passons en revue les différentes étapes de l'intégration et discutons des spécificités ci-dessous.

PRE-INTEGRATION

Nous avons vu dans le cadre des BD classiques que la pré-intégration avait pour objectif de préparer l'unification. Cette préparation se traduisait par l'enrichissement sémantique des schémas source, leur normalisation et leur transformation dans des schémas plus proches. Pour les BD géographiques, on retrouve les mêmes étapes mais ce travail ne se limite pas aux schémas.

> *Pour homogénéiser les schémas, il est nécessaire de comprendre ce que contiennent les bases de données. Cela implique une analyse des données elles-mêmes qui renferment des phénomènes géographiques implicites.*

Nous considérons que l'intégration des BD géographiques passe par une étude approfondie des données en collaboration avec les schémas. C'est ce point de vue qui est adopté dans cette thèse. Ceci découle d'une particularité majeure de ce type de BD : la présence d'informations implicites. Cet aspect sera détaillé davantage dans le chapitre B mais il est nécessaire de le signaler à ce stade du mémoire. La représentation des données véhicule davantage d'informations que la base n'en stocke. Pour cette raison, l'étape d'enrichissement des BDG concerne à la fois les schémas et les données.

Prenons un exemple très simple. Considérons l'existence des classes « Route » et « Échangeur routier » dans la première BD et la classe « Route » dans la seconde. Si on compare les schémas des deux BD, il semble que celui de la première soit plus riche que la seconde et par conséquent, que le contenu des bases soit différent (figure 14). Pourtant, les échangeurs routiers existent aussi dans la seconde base. Ceux-ci ne sont pas directement stockés en tant que tels (ils sont noyés avec les instances de la classe « Route »), mais ces objets sont visibles dans les données (on peut les voir en affichant les données). Ils peuvent être extraits et individualisés si nécessaire. Sans une analyse des données géométriques, l'enrichissement dans la deuxième BD ne serait peut-être pas envisagé alors qu'il permettrait une intégration plus simple et plus juste.

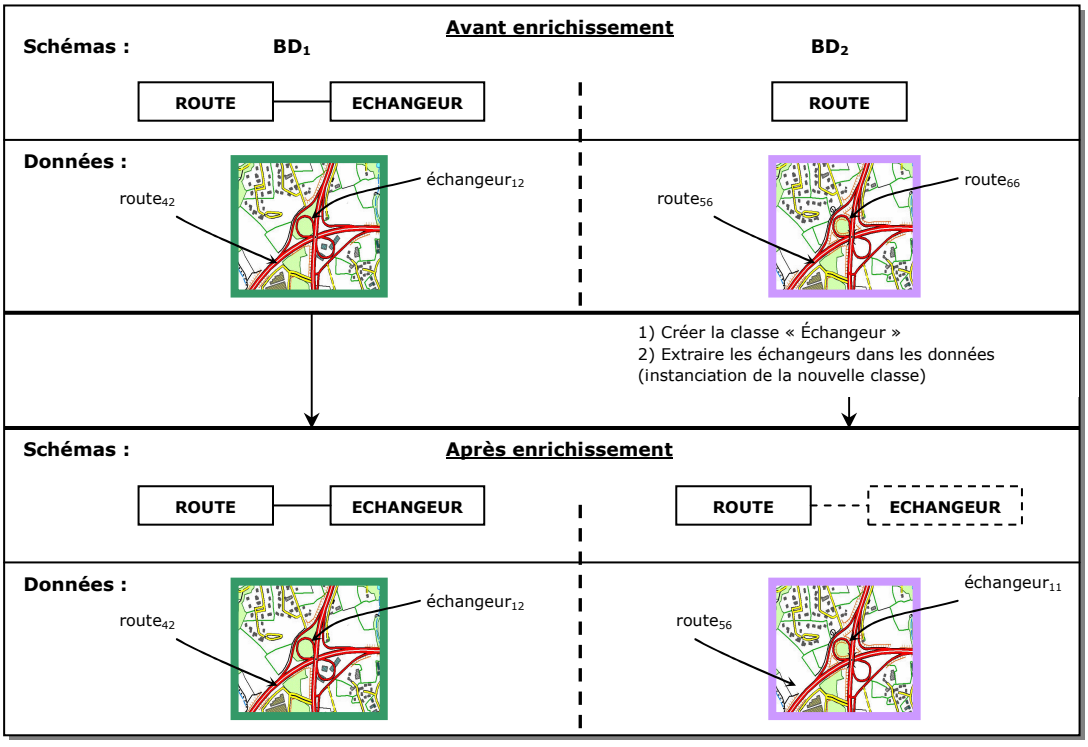


***Figure 14**. Enrichir les données pour préparer l'intégration.*

Pour les bases de données traditionnelles, l'enrichissement sémantique passe par la récolte d'informations auprès de l'administrateur et par la consultation de documents associés à la base (les métadonnées auxquelles nous avons fait référence en A.3.2.1.). Dans le cadre des bases de données géographiques, cet enrichissement peut également être guidé par des connaissances du domaine : les *spécifications*. Il

s'agit de documents qui présentent une description très détaillée du contenu de chaque classe (règles de sélection des objets et de leur modélisation). Ils sont donc particulièrement intéressants pour guider le processus d'intégration. Nous décrirons en détail ces documents dans le chapitre B. Ce sont les métadonnées que nous utilisons pour étudier la conformité des représentations.

En plus de cet enrichissement, nous avons vu que des règles de normalisation devaient être définies pour réduire les différences de modélisation. Pour les BD géographiques, ces règles sont particulièrement nécessaires car l'hétérogénéité des modélisations est plus importante que pour les bases de données classiques. Pour s'en convaincre, il suffit par exemple de comparer les différentes solutions de modélisation de la topologie des objets géographiques (modèle spaghetti, topologique) [Laurini et Milleret-Raffort 1993]. Nous donnons deux modélisations différentes en figure 15.

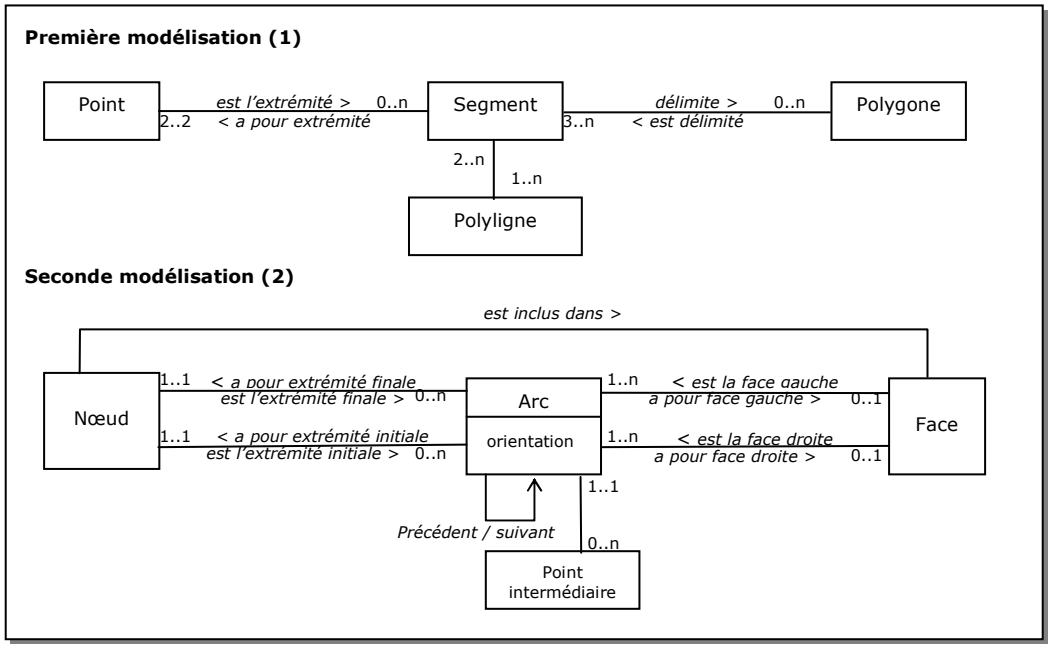


***Figure 15**. Deux exemples de solutions pour modéliser les données géographiques : le modèle spaghetti polygonal unifié (1) et le modèle de la carte topologique (2)*

Ces deux modélisations pourraient constituer la couche des primitives de base pour représenter la géométrie des objets géographiques dans un modèle conceptuel de données. Les relations entre les primitives géométriques sont cependant différentes et illustrent deux modèles différents : le modèle *spaghetti polygonal unifié* [Ubeda 1997] et le modèle de la *carte topologique* [David et al. 1993a].

Ces modèles de représentation de la topologie sont souvent différents entre les bases à intégrer. Les BDG sont généralement structurées selon un modèle propriétaire (Geoconcept, MapInfo, Apic, ArcGis,…). Lors de l'intégration, il est nécessaire d'imposer une modélisation qui peut être déterminée en suivant un standard (comme la norme EDIGéO [Laurini et Milleret-Raffort 1993] ou le modèle de référence de l'OpenGIS par exemple [OpenGIS 2001]). La normalisation de la modélisation et le choix du modèle commun permettent de traiter notamment les *conflits de modélisation de la topologie* et les *conflits de modèle* (relationnel étendu, O.O.) [Devogele 1997].

Il est important de préciser que même si deux bases de données géographiques à intégrer présentent des schémas assez similaires, l'hétérogénéité entre les données sera plus importante que pour des bases traditionnelles. En effet, si les spécifications

précisent les règles de saisie des objets et leur modélisation, elles laissent malgré tout une certaine part à l'interprétation. De plus, à des niveaux de qualité équivalents (même niveau de complétude, même niveau d'exactitude de position des objets, etc.), la nature même des données géographiques introduit des imprécisions. Comment fixer précisément la limite d'une forêt par exemple ? Deux personnes chargées de délimiter cet objet en suivant les mêmes spécifications produiront immanquablement un découpage différent, en raison du caractère flou de la limite de ce phénomène. Nous reviendrons sur cet aspect dans la partie consacrée aux connaissances nécessaires pour évaluer les différences de représentations (chapitre B).

Avant de rechercher les correspondances entre les schémas et les données, il faut également s'assurer que les bases possèdent le même mode de représentation et le même système de référence. Suivant le cas, une transformation pour passer en mode *vectoriel* ou *matriciel* (appelé encore image ou « *raster* ») sera nécessaire. Un changement de projections pourra également s'imposer. Nous ne détaillons pas davantage cet aspect car nous faisons l'hypothèse que les bases que nous utilisons sont en mode vectoriel et que leur système de référence sont identiques.

IDENTIFICATION DES CORRESPONDANCES ET DES CONFLITS ENTRE LES SCHEMAS

L'identification des correspondances entre les schémas est une étape analogue à celle du processus d'intégration classique. Néanmoins, le nombre de conflits entre les éléments des schémas (et par conséquent entre les données) est beaucoup plus important.

En plus des conflits d'hétérogénéité habituels, il existe des conflits spécifiques aux BD géographiques [Laurini 1996, Parent et al. 1996]. Les bases à intégrer présentent généralement des différences de niveaux de détails [Ruas 2002a] et par conséquent, la représentation des objets est souvent différente. En effet, pour des raisons de lisibilité, la résolution impose de ne saisir que des objets d'une certaine taille. De plus, certaines caractéristiques des objets sont gommées ou *généralisées* : des angularités disparaissent, des objets sont fusionnés, simplifiés, des décrochements sont éliminés, etc. Le contenu des bases est donc différent. Ceci ne résulte pas seulement des différences de résolution. Ces différences découlent également des différences de point de vue que l'on peut porter sur l'espace (cf. introduction). Un urbaniste et un agronome auront un regard différent sur l'occupation du sol par exemple. Par ailleurs, le niveau de qualité des BD est rarement le même. Il dépend de la résolution mais également de la tolérance que s'imposent les producteurs. L'un d'eux peut accepter un taux de confusion de 10% entre les classes de la base par exemple (paramètre d'*exactitude sémantique*). Un autre producteur peut fixer son taux à 5%.

Toutes ces différences ne sont pas directement visibles au niveau des schémas. Ainsi, si l'on compare deux schémas de sources différentes, on retrouvera les conflits classiques d'hétérogénéité, comme les différences de classification des éléments, de domaine de valeur des attributs, etc. Pour comprendre la sémantique des éléments du schéma et se rendre compte de la majorité des différences de représentation, il est nécessaire d'étudier les spécifications des sources. C'est seulement à l'issue de cette étude que les différences de représentation vont pouvoir être exprimées de manière précise dans les ACI si comme le propose [Devogele 1997], ce langage est étendu pour supporter la déclaration des conflits spécifiques aux BDG.

Nous avons représenté en figure 16 un extrait de deux schémas à intégrer (nous nous sommes limités à une classe pour simplifier l'exemple). Un examen rapide de ces classes permet d'identifier des différences : la BD_1 possède davantage d'attributs que

la BD_2, un découpage différent semble exister pour l'attribut concernant le sens de circulation des tronçons, et deux attributs présentent une dénomination différente. Après une analyse des spécifications, les différences entre les classes ont pu être précisées. D'abord, un conflit relatif au critère de sélection a pu être mis en évidence et celui-ci n'était pas perceptible auparavant. Ainsi, il a pu être spécifié que les tronçons de la BD_1 n'ont un correspondant dans la BD_2 que si leur longueur est supérieure à 200 mètres (critère d'existence d'un objet de la base BD_2). Ensuite, l'équivalence entre les attributs « type » et « importance » a pu être déclarée (les attributs ont donc le même sens). Enfin, le découpage différent des attributs a été confirmé. La somme des valeurs des attributs relatifs au nombre de voies de la BD_1 correspond à la valeur de l'attribut « Nb_Voies » dans la BD_2.

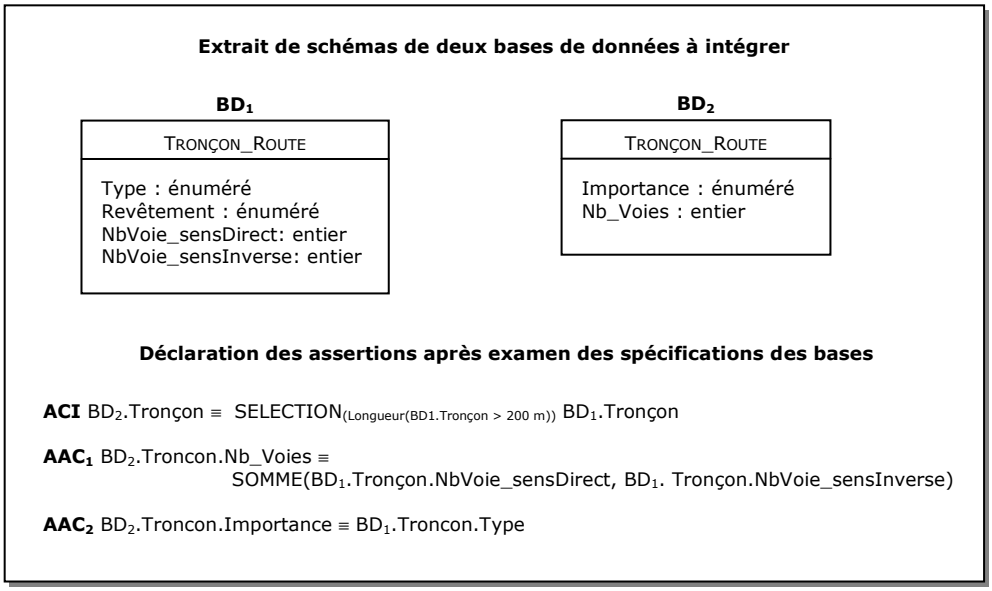


***Figure 16**. L'identification des correspondances et des conflits entre les schémas de BDG nécessite d'étudier les spécifications des bases.*

En fait, si l'étude des spécifications permet de déclarer un grand nombre de conflits entre les éléments des schémas, ce travail n'est pas toujours suffisant. Les spécifications des BD géographiques peuvent être ambiguës et incomplètes. Leur description n'est pas toujours suffisamment riche pour le travail d'intégration car elles n'ont pas été conçues à l'origine pour cela. Ainsi, il est possible que, dans l'exemple précédent, toutes les correspondances au niveau des schémas n'aient pas été déclarées par manque d'informations dans les spécifications et donc, que les correspondances dans les données soient légèrement différentes. On peut par exemple s'apercevoir en analysant les données que toutes les impasses dans la BD_2 existent si elles ont une longueur supérieure à 250 m. Dans ce cas, l'ACI suivante doit également être déclarée :

ACI $BD_2.\text{Tronçon} \equiv \text{SELECTION}_{(\text{Impasse}(BD1.\text{Tronçon} > 250\ m))}\ BD_1.\text{Tronçon}$

Il faut noter que le terme « impasse » dans cette assertion désigne une fonction [Devogele 1997]. Les impasses ne sont pas individualisées dans la base (information implicite). Il faut les extraire des données par une analyse de la topologie du réseau.

Pour mener à bien l'étude des correspondances, nous considérons qu'il est nécessaire de travailler conjointement au niveau des schémas et des données, en s'aidant des spécifications.

IDENTIFICATION DES CORRESPONDANCES ET DES CONFLITS ENTRE LES DONNEES

Nous venons de discuter de l'identification des correspondances et des conflits entre les éléments des schémas. Ils sont exprimés au moyen d'un langage qui nécessite d'être étendu pour prendre en compte les particularités des BDG. Dans le cadre des BD traditionnelles, la deuxième étape du processus d'intégration s'achèverait ici. Dans notre contexte, il est également nécessaire d'identifier les correspondances au niveau des données.

Comme nous l'avons déjà précisé en présentant le langage des ACI, il est rarement possible de s'appuyer sur la notion d'identifiant commun pour mettre en correspondance les instances (avec la clause AIC). En plus des relations à déclarer entre les classes et les attributs des schémas source, il est nécessaire de mettre en œuvre un processus d'appariement spécifique qui relie les instances géométriques. On exploite donc la géométrie des objets et leur position pour les apparier mais rarement leurs informations attributaires. Nous distinguons de ce fait l'étude des correspondances entre les schémas et les données de même que leur intégration.

La distinction de ces étapes ne veut pas dire que l'intégration des schémas et des données est complètement déconnectée. Au contraire, avant d'apparier les instances géométriques, il est nécessaire de sélectionner les classes dont le contenu s'intersecte. On doit avoir une idée, même approximative, des correspondances entre les éléments des schémas (la sélection des candidats à l'appariement peut se limiter à un thème par exemple, comme le thème hydrographique). Réciproquement, l'appariement géométrique peut grandement faciliter la déclaration des correspondances au niveau des schémas. Par exemple, il est relativement facile de constater interactivement qu'à l'issue de l'appariement d'un ensemble de données, les éléments de la classe « Route » de la première BD sont en correspondance avec les éléments de la « Route » et « Chemin » de la seconde. Plus exactement, on peut s'apercevoir que les routes dont l'attribut *nature* a pour valeur *'chemin'* sont appariées avec les instances de la classe « Chemin ». L'appariement peut donc être utilisé pour aider à analyser les spécifications des bases, tâche souvent fastidieuse mais nécessaire pour décrire les correspondances entre les schémas.

Au terme de l'appariement, un ensemble de couples d'objets appariés est fourni et ces couples mettent en évidence un ensemble de différences : les *conflits de données* [Devogele 1997] (figure 17).

A ce niveau, on retrouve tous les conflits exprimés dans les assertions de correspondances inter-schémas avec en plus des correspondances entre les données qui ne sont pas cohérentes au regard des spécifications (conflits anormaux). Il reste donc à l'issue de l'appariement des données une autre grande étape à mettre en œuvre : l'analyse et l'interprétation des différences entre les correspondances pour distinguer les conflits normaux des conflits anormaux. Cette étape est indispensable pour assurer une cohérence dans le système intégré et éviter l'intégration d'erreurs. Elle est pourtant généralement passée sous silence et rarement abordée dans le cadre des BDG (nous le verrons dans la suite de ce chapitre).

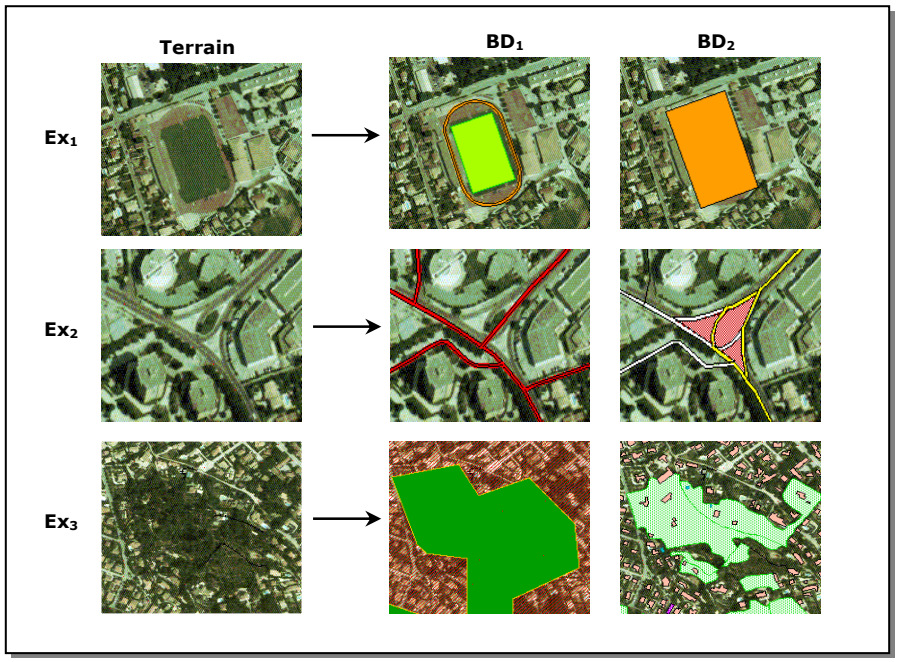


***Figure 17**. Exemples de différences de représentation entre deux bases de données*

Nous venons de présenter les particularités des étapes de pré-intégration et de mise en correspondances des schémas et des données dans le processus d'intégration de BDG. La troisième grande étape du processus est l'intégration proprement dite. Nous la décrivons ci-dessous.

INTEGRATION DES SCHEMAS

Les correspondances et les conflits étant déclarés entre les schémas et les données, l'intégration proprement dite peut être menée. Pour réaliser cette intégration, nous avons vu qu'il était nécessaire d'adopter une stratégie. Celle-ci va guider la résolution des conflits entre les schémas et les données. Dans le cadre des BDG, deux stratégies principales peuvent être adoptées [Devogele 1997] :

- Une stratégie mono-représentation : on ne cherche pas à conserver l'ensemble des représentations mais plutôt à garder et fusionner les informations les plus riches des deux bases en supprimant les redondances.
- Une stratégie multi-représentations : on préserve les représentations des différentes bases qui présentent des niveaux de détails différents en acceptant les redondances.

Prenons l'exemple d'un *conflit de fragmentation* entre classes pour lequel un objet d'une classe d'une BD peut correspondre à plusieurs objets d'une classe dans l'autre BD (nous reviendrons sur ce conflit dans la section A.4.2.1.). En termes d'ACI, ce type de conflit peut être exprimé de la manière suivante :

ACI BD_1.Element$_1$ <Rel> BD_2.SET([1:N] Element$_{1'}$)

Pour résoudre ce type de conflit, suivant la stratégie d'intégration retenue, la méthode sera différente. Dans une optique mono-représentation, seule l'information la plus fragmentée et donc la plus riche sera conservée (si les spécifications de la nouvelle base l'impose). Dans une approche multi-représentations, le schéma intégré inclura également la classe présentant un niveau d'agrégation plus important.

A cette étape, les conflits entre schémas doivent donc être résolus en suivant une stratégie d'intégration. Il est également nécessaire de définir des mécanismes de traduction permettant de passer des schémas source au schéma unifié. Enfin, des modèles de données adaptés à la représentation de l'information géographique et supportant éventuellement la représentation multiple doivent être utilisés.

INTEGRATION DES DONNEES

La production du schéma intégré et la résolution des conflits à ce niveau doivent être accompagnées de l'intégration des données. Suivant la stratégie choisie, les données géométriques doivent être fusionnées ou reliées entre elles de manière cohérente. On peut ainsi décider de transférer les attributs d'une des bases sur la géométrie plus détaillée de l'autre base. On peut également agréger la géométrie de certains objets qui représentent des mêmes phénomènes dans la réalité. Si les jeux de données sont adjacents, leurs frontières doivent être raccordées et corrigées topologiquement [Laurini 1996].

Les conflits anormaux qui résultent d'erreurs de saisie ou de mises à jour doivent également être traités. Suivant le type d'erreur, des retouches interactives de la géométrie sont à appliquer, certains objets peuvent être éliminés, d'autres incorporés dans la base unifiée.

Ces différentes opérations sont guidées par les spécifications de la nouvelle base et nécessitent l'utilisation d'outils adaptés, spécifiques au traitement de l'information géographique numérique (techniques de fusion, algorithmes de généralisation, raccordements géométriques,…).

CONCLUSION

Les spécificités de l'intégration des BD géographiques que nous avons mises en avant demandent d'apporter des modifications à la description générale du processus proposé pour les bases de données classiques. Nous représentons ce processus modifié en figure 18 en situant notre travail de thèse.

Nous distinguons davantage l'intégration des schémas et des données car la déclaration des correspondances entre les schémas n'est pas suffisante pour retrouver les correspondances entre les données. Il est nécessaire de mettre en œuvre un processus d'appariement géométrique pour relier les données. L'intégration proprement dite peut également faire appel à des algorithmes géométriques pour fusionner les données, les agréger, les simplifier, etc.

Nous mettons également en évidence l'importance d'uiliser les spécifications dans le processus d'intégration. Ces métadonnées permettent de comprendre le contenu des bases, de donner un sens aux éléments des schémas et des données, de guider l'enrichissement des schémas et des données, et d'étudier les différences de représentation entre les données.

L'étape que nous traitons dans cette thèse se situe au niveau des données. Il s'agit de la *recherche des correspondances entre les données et l'étude de leurs différences.* Il ne s'agit donc que d'une étape du processus d'intégration. Néanmoins, pour mettre en œuvre cette étape, l'étape précédente doit être réalisée. Cela signifie que nous étudierons les spécifications, les schémas et les données au préalable (étape de la pré-intégration).

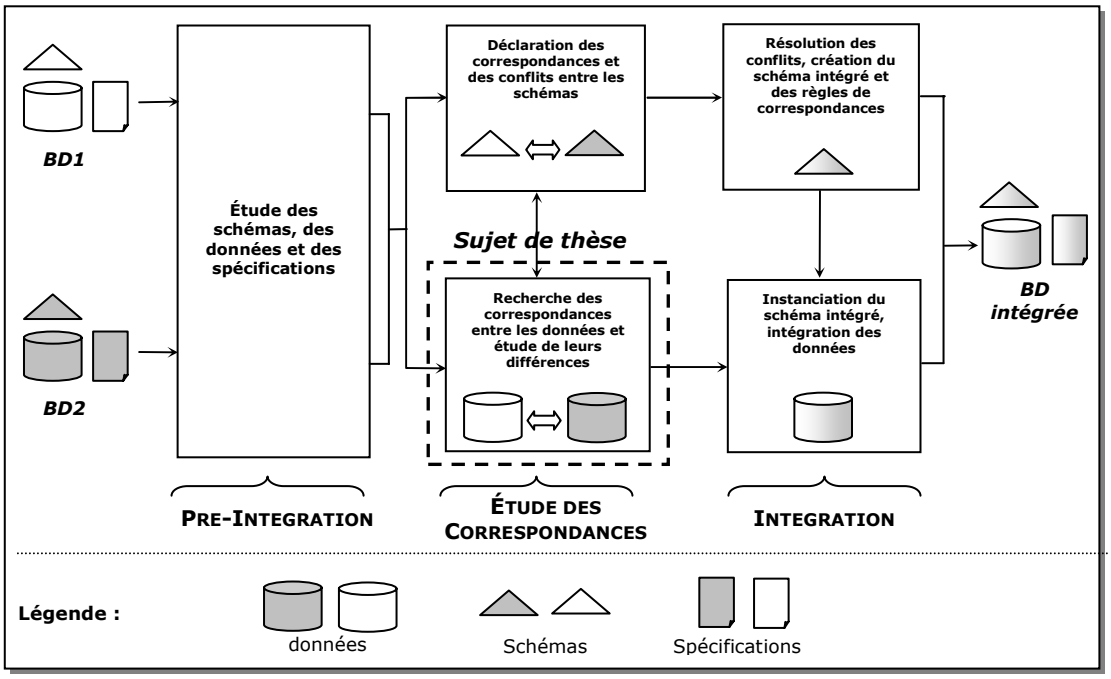


***Figure 18**. L'intégration de BD géographiques requiert un travail au niveau des schémas et des données.*

Dans présentons dans les sections suivantes les contributions proposées dans la littérature pour intégrer les bases de données géographiques.

A.4.2 TRAVAUX SUR L'INTEGRATION DES SCHEMAS DE BDG

Maintenant que nous venons de souligner certains traits caractéristiques de l'intégration des BD géographiques, nous pouvons présenter les travaux s'y rapportant. Il faut noter que le sujet a rarement été étudié dans sa globalité. Il existe peu de méthodologies complètes spécifiques aux BDG. Généralement, les efforts ont porté sur un problème particulier de l'intégration que ce soit au niveau des schémas ou des données. Nous nous sommes inspirés de ces différents travaux pour mener notre recherche. Nous verrons que les propositions concernant l'évaluation de la cohérence entre les données sont peu nombreuses et qu'aucune de ces propositions n'exploite formellement les spécifications des BDG. Dans cette partie, nous présentons les approches qui traitent des schémas (figure 19).

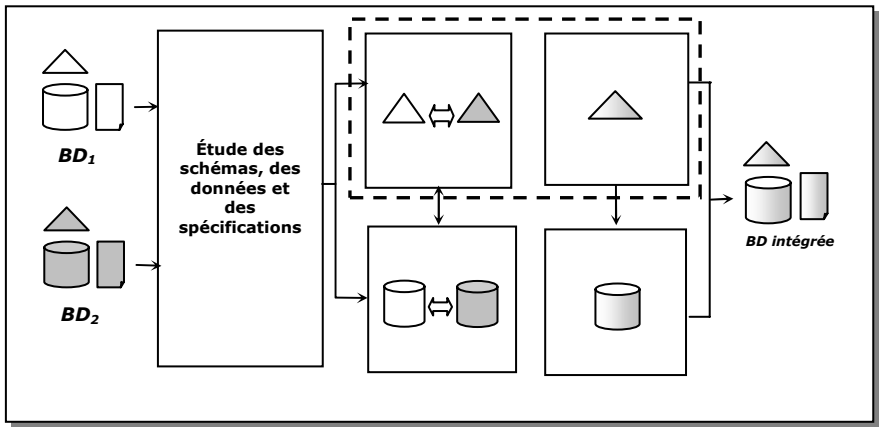


***Figure 19**. Illustration de la position des travaux présentés dans cette partie par rapport au processus d'intégration des BDG.*

A.4.2.1 APPROCHES METHODOLOGIQUES

APPROCHE DE [DEVOGELE 1997]

L'approche méthodologique sans doute la plus complète aujourd'hui pour intégrer des BD géographiques est celle définie par [Devogele 1997, Devogele et al. 1998]. L'auteur a étendu le processus d'intégration déclaratif proposé par [Spaccapietra et al. 1992] (présenté à la section A.3.2.) pour prendre en compte la dimension spatiale des éléments à intégrer. Plusieurs efforts préalables avaient déjà été fournis pour se raccrocher à une méthodologie d'intégration existante mais sans réellement tenir compte de la spécificité des BDG [Nyerges 1989]. Le processus de [Devogele 1997] comprend les trois phases de l'intégration : la pré-intégration, la recherche des correspondances et l'intégration. Ce travail apporte plusieurs contributions au problème d'intégration des BDG. D'abord, une taxonomie des conflits spécifiques a été proposée (figure 20). Parmi ces conflits, on peut citer [Parent et al. 1996, Devogele 1997] :

- Les *conflits de métadonnées géométriques* : conflits de résolution, de précision et d'exactitude qui sont susceptibles de provoquer des *conflits de données*.
- Les *conflits de définition des classes* : on distingue plusieurs catégories dont les *conflits de critère de spécification*, qui peuvent se traduire par des contraintes de sélection ou de décomposition des objets différentes. Il existe aussi les *conflits de fragmentation* qui peuvent se traduire par des *conflits de segmentation* (découpage des objets selon des attributs différents), *de granularité* (découpage des objets selon le même attribut mais en prenant en compte un critère d'homogénéité différent) ou de *décomposition* (un objet dans une base correspond à plusieurs objets dans l'autre base). Des exemples sont fournis à la figure 20.
- Les *conflits de structure* : en plus des conflits classiques que l'on peut trouver entre les schémas (comme la modélisation d'un concept sous forme de classe dans l'un et sous forme d'attribut dans l'autre), on trouve les *conflits de stockage de l'information*. Ils font référence aux informations implicites que l'on peut déduire des BD.
- Les *conflits de description sémantique et géométrique* : ces conflits résultent des différences de propriétés des classes en correspondance. Ils concernent notamment le nom de la classe, les attributs (domaine de valeur, type,…) et la dimension de la géométrie retenue (point, ligne, polygone).

En plus de cette taxonomie, le langage d'assertion de correspondance inter-schémas (ACI) a été étendu pour exprimer les relations entre les éléments des BDG et déclarer les conflits spécifiques. Cette extension se traduit notamment par l'ajout de la notion de direction dans les clauses AAC, la définition d'une clause relative à l'appariement géométrique (AGD), la définition d'une clause relative aux conflits de description de la géométrie (AGC - *Avec Géométrie Correspondante*) et différentes solutions pour traduire les conflits de critères de spécification, de fragmentation, etc. Plusieurs réponses ont également été apportées pour résoudre ces conflits lors de la phase d'intégration proprement dite.

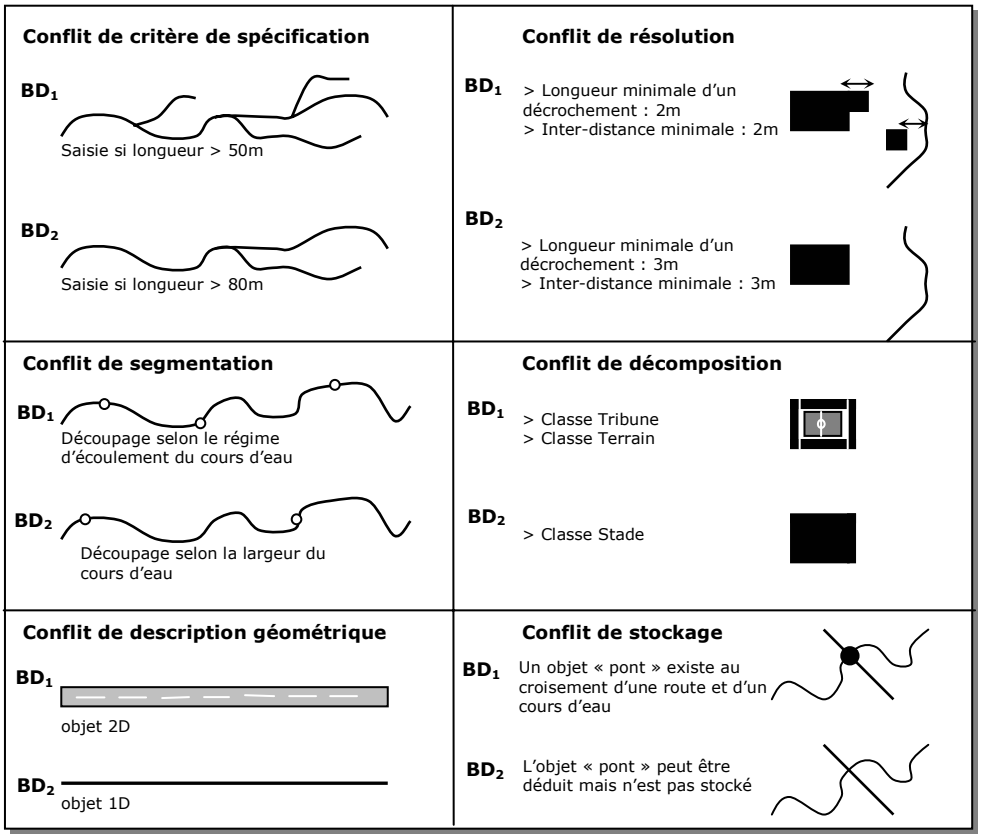


***Figure 20**. Quelques exemples de conflits d'intégration spécifiques aux bases de données géographiques. (D'après la classification de [Devogele 1997])*

Un processus d'appariement géométrique automatique a enfin été défini pour mettre en correspondances les données. Il permet de relier deux réseaux routiers de BDG ayant des échelles différentes.

Cette approche méthodologique est assez détaillée et nous nous sommes beaucoup inspiré de ce travail pour réaliser notre étude. Nous avons également utilisé les algorithmes d'appariement géométrique proposés. Néanmoins, la méthodologie de [Devogele 1997] n'apporte pas de solution pour évaluer la cohérence des données une fois celles-ci appariées. Notre thèse qui porte sur ce sujet permet donc de compléter cette approche.

APPROCHE DE [BRANKI ET DEFUDE 1998]

D'autres auteurs ont également apporté leur contribution à l'intégration des schémas. Ainsi, [Branki et Defude 1998] suggèrent une méthodologie d'intégration fondée sur l'utilisation d'une logique de description. Le processus se compose des étapes suivantes :

- La pré-intégration : les schémas initiaux des BD sont transformés dans un autre langage de représentation — le modèle sémantique *GeoCoopm* — en traduisant les différents éléments des schémas source en termes de concepts et de rôles (modèle dérivé des logiques de description).
- L'analyse des schémas : une fois la transformation réalisée dans le modèle canonique, les attributs (rôles) sont comparés et une hiérarchie d'attributs globale est définie. Cette hiérarchie est construite manuellement, par l'expert du domaine, en déduisant les correspondances.

- La construction d'une version préliminaire du schéma intégré : à partir de cette hiérarchie d'attributs, un graphe de concepts est construit automatiquement. Cette construction est facilitée par la fonction de subsomption propre aux logiques de description (déduction de liens *'is-a'* : liens de généralisation-spécialisation). Le graphe conceptuel fait apparaître différentes relations entre les concepts des schémas source.
- La restructuration et l'enrichissement du schéma intégré : pour tenir compte de certains conflits spécifiques aux BDG, plusieurs opérations de restructuration sont définies. Deux catégories d'opérateurs sont proposées : les opérateurs de restructuration des schémas et les opérateurs de restructuration des représentations spatiales. Les premiers permettent de créer de nouveaux concepts à partir des concepts existants (opérateurs de généralisation, de spécialisation). Les seconds sont destinés à redéfinir les concepts appartenant à une certaine représentation spatiale dans une autre représentation spatiale. Ces restructurations sont guidées par des métadonnées assez générales qui concernent notamment la résolution, le système de référence et la dimension des instances représentées par les concepts. On aboutit finalement au schéma intégré.

La proposition de [Branki et Defude 1998] concerne principalement les schémas. Leur méthodologie d'intégration fondée sur l'utilisation d'une logique de description s'inspire des travaux réalisés dans le cadre des BD classiques. Les auteurs tiennent compte des conflits spécifiques aux bases de données spatiales. L'intégration est guidée par des métadonnées générales.

APPROCHE DE [STRAUCH ET AL. 1998]

La méthodologie *MMultiGIS* proposée par [Strauch et al. 1998] comprend les étapes classiques d'intégration (pré-intégration, analyse des schémas, intégration) suivie de la création de schémas externes et de leur validation. Le modèle commun choisi (modèle pivot) correspond à une extension d'un format de stockage et d'échange de données (*SAIF - Spatial Archive and Interchange Format*). L'analyse des correspondances entre les éléments des schémas porte sur trois *contextes* différents : le contexte spatial (analyse des différences relatives aux systèmes de référence cartographique utilisés, à l'étendue de la région concernée), le contexte d'application (analyse des différences à partir de métadonnées décrivant le domaine d'application de la base) et le contexte sémantique (analyse de la proximité sémantique existant entre les classes et attributs des deux schémas). La résolution des conflits et la phase d'intégration sont ensuite réalisées, se traduisant par la création du schéma global et se poursuivant par la création de schémas externes.

Nous retenons de cette approche que des métadonnées sont utilisées pour décrire le domaine d'application des bases.

APPROCHE DE [PARK 2001]

La proposition de [Park 2001] montre clairement que l'intégration des BDG passe par un travail au niveau des schémas et des données. Une première étape consiste à analyser et déclarer les correspondances entre les éléments des schémas et ce, en utilisant des métadonnées sur la sémantique des éléments. Parallèlement, un processus de conversion des données permet de transformer les instances pour les rendre plus homogènes et les mettre en relation. Les fonctions de conversion incluent des méthodes d'appariement (fonctions de superposition), de transformation des

formats et des modes de représentation, d'analyse de la topologie, des réseaux, etc. Un prototype d'AGL (atelier de génie logiciel) a été défini. Il consiste en plusieurs modules : une module de conception de schémas, un module de traduction des schémas dans le modèle canonique défini *(« Unifying Semantic Model »)*, un module permettant de définir les relations entre les éléments des schémas, et une librairie de fonctions de conversion dédiées aux données géométriques.

Cette approche est intéressante. [Park 2001] distingue l'intégration des schémas et des données, de manière analogue à la nôtre. L'auteur utilise également des métadonnées pour comprendre la sémantique des schémas. La cohérence entre les données n'est pas étudiée.

APPROCHE DE [LASSOUED ET AL. 2004]

Une autre contribution récente fondée sur l'utilisation de l'apprentissage automatique multi-stratégies a récemment été proposée [Lassoued et al. 2004][8]. Les auteurs cherchent à établir les correspondances entre un schéma global défini dans un contexte de médiation et de nouveaux schémas source. Leur méthode est inspirée des travaux de [Doan et al. 2003], adaptés aux BD géographiques. Les schémas initiaux sont d'abord traduits dans le modèle de données préconisé par l'OpenGIS Consortium et enregistré dans le format GML (« *Geography Markup Language* »)[9]. Les schémas source sont ensuite raffinés et étendus de façon à faciliter l'intégration (on peut faire l'analogie avec l'étape d'enrichissement sémantique du processus d'intégration). Ce raffinement est réalisé en se fondant sur la notion d'*attribut discriminant* qui permet de spécialiser certaines classes des schémas (on décompose par exemple un attribut énuméré en plusieurs sous-classes). La recherche de ces propriétés discriminantes est facilitée par l'emploi d'apprenants (algorithmes d'apprentissage automatique) : le « *Name Learner* » et le « *Content Learner* ». A partir d'un ensemble d'exemples d'apprentissage composés des noms d'attributs et de leurs valeurs (avec leur classe correspondante fournie par l'expert : attribut discriminant ou non), les apprenants permettent d'associer des notes (*scores*) aux attributs, reflétant le degré auquel ils considèrent ces attributs comme discriminant ou non. Ces notes sont combinées par un méta-apprenant qui détermine des coefficients de confiance sur les apprenants respectifs. Une fois ce raffinement réalisé, les correspondances entre le schéma étendu et le schéma global sont déterminées. Plusieurs apprenants sont également utilisés à cette étape dont un apprenant géométrique. Celui-ci exploite cette fois les propriétés géométriques des objets du schéma source pour les classer. Le système d'apprentissage est fondé sur un réseau de neurones : à partir de propriétés géométriques calculées, le système détermine la classe de l'objet (route, bâtiment, cours d'eau, …).

Cette contribution est donc d'ordre méthodologique, mais vise aussi à trouver des solutions pour automatiser la mise en correspondance des schémas. C'est une approche qui exploite l'apprentissage automatique, comme celle que nous proposons dans cette thèse.

[8] Cette contribution s'inscrit dans le cadre du projet RNTL VirGIS. On peut trouver une description du projet sur le site : http://www.telecom.gouv.fr/rntl/FichesA/Virgis.htm

[9] On trouvera une description du format GML sur : http://www.opengis.net/gml/

A.4.2.2 MODELES SUPPORTANT LA REPRESENTATION MULTIPLE

La modélisation conceptuelle des BD géographiques requiert l'utilisation de formalismes[10] adaptés à l'information spatiale [Pantazis et Donnay 1996, Hadzilacos et Tryfona 1998]. Les deux propositions les plus abouties aujourd'hui à ce sujet sont celles de l'équipe de Stefano Spaccapietra de l'EPFL et Christine Parent, qui proposent le modèle MADS[11] [Parent et al. 1998], et celle de l'équipe d'Yvan Bédard de l'université Laval, fondée sur les PVL's [Bédard 1999].

Dans un contexte d'intégration, il est nécessaire d'utiliser des modèles suffisamment riches pour exprimer la sémantique des données dans les schémas et faciliter ainsi la comparaison des concepts (classes, attributs, relations, contraintes, etc.). Par ailleurs, suivant la stratégie adoptée pour l'intégration, les modèles doivent être capables de supporter des concepts permettant la représentation multiple [Vangenot 2001]. Une base de données multi-représentations est une base dans laquelle sont stockées plusieurs représentations d'une même entité géographique, ces représentations étant liées à des niveaux de détails et des points de vue qui leur sont propres. Les modèles conceptuels de données doivent permettre de repésenter cette multiplicité. Nous détaillons ci-dessous les solutions proposées par [Vangenot et al. 2002] pour le modèle MADS et celle de [Bédard et al. 2002, Proulx et al. 2002] dans le cadre des PVL's.

REPRESENTATION MULTIPLE DANS MADS

Le modèle MADS est un modèle conceptuel spatio-temporel entité-association doté des concepts principaux de l'approche orientée-objet. Il offre une large palette d'outils de modélisation des dimensions spatiale et temporelle. La structure supporte ainsi des types abstraits de données spatiaux et temporels, différents types d'associations (thématique, topologique, temporelle, héritage, composition, etc.) et plusieurs types d'attributs (attributs à valeurs variables dans l'espace notamment).

Deux approches complémentaires ont été proposées pour modéliser la représentation multiple dans MADS [Vangenot et al. 2002] :

- *L'approche par intégration* : les différentes représentations initiales sont regroupées dans un même type (figure 21). Le concept d'*estampille* est utilisé pour définir pour quels contextes une représentation a été élaborée et dans quelles conditions on a accès à la représentation. Une estampille est une paire (point de vue, résolution) ;
- *L'approche par mise en correspondance* : les différentes représentations sont reliées par des associations de correspondances (figure 21). Ces associations sont de plusieurs types : équivalence, agrégation et lien SetToSet.

Le modèle MADS et les concepts de multi-représentations ont été mis en œuvre dans le cadre du projet européen MurMur[12] [Spaccapietra et al. 1999, Balley et al. 2004]. Un AGL a été développé pour modéliser les schémas, créer les tables relationnelles correspondantes et interroger une base de données multi-

[10] Nous utilisons ici le terme *formalisme* comme synonyme de *langage de description de données* (UML par exemple) – cf. A.2.

[11] Une description détaillée du modèle MADS est fournie sur http://lbdwww.epfl.ch/e/research/mads/

[12] Le projet a réuni 6 partenaires : l'EPFL, l'ULB, l'UNIL, le CEMAGREF, Star Informatic et l'IGN (projet IST 10723). Une description peut être trouvée sur : http://lbdwww.epfl.ch/e/MurMur/

représentations (création de requêtes SQL en manipulant une représentation graphique des schémas).

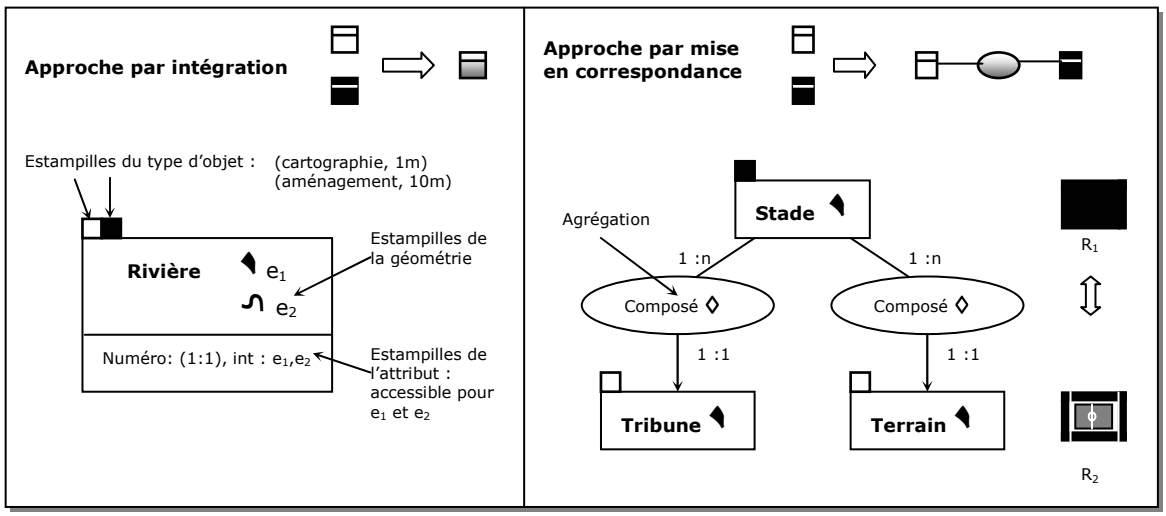


***Figure 21**. Les deux approches de modélisation de la représentation multiple dans MADS (D'après [Vangenot et al. 2002])*

MADS a également été utilisé dans le cadre du projet AMBER[13] récemment terminé [Sotnykova 2003]. L'objectif de ce projet consistait à développer un système fédéré destiné à la gestion de l'environnement à partir de plusieurs bases de données géographiques. Le schéma intégré a été réalisé en MADS. La déclaration des correspondances et des conflits sémantiques s'est faite à l'aide des ACI et un outil d'intégration semble avoir été développé pour exprimer et vérifier la syntaxe de ces ACI à partir des schémas source. Un travail concernant la formulation de contraintes d'intégrité dans MADS a également été mené. Ces contraintes sont liées aux conflits déclarés dans les ACI : elles font référence à des métadonnées concernant les conditions de représentation et d'existence des objets dans les BDG. Ces contraintes sont représentées à l'aide de la logique de premier ordre au niveau conceptuel, lorsque les éléments de base du modèle ne suffisent pas, et dans l'algèbre MADS pour l'implémentation. Nous donnons un exemple de formulation tiré de [Sotnykova 2003] ci-dessous :

Contrainte énoncée en langue naturelle :

« si la surface d'un parterre de fleurs est supérieure à 5m², alors certaines des fleurs devraient correspondre à des roses blanches et la valeur de l'attribut 'type de fleur' doit correspondre à 3 ».

Contrainte en logique des prédicats du 1er ordre :

$\forall x,\ x \in \mathrm{Pop}(\mathit{Parterre\_fleurs}) \wedge \mathrm{surface}(x) > 5 \rightarrow 3 \in \mathit{Type\_Fleur}(x)$

Cette contribution sur la formulation des contraintes d'intégrité est proche du travail que nous avons mené sur la représentation des spécifications des bases de données géographiques (voir chapitre D). Exprimer les spécifications de manière formelle est indispensable pour manipuler ces connaissances automatiquement et permettre ainsi une intégration sémantique des bases. La visualisation des contraintes

[13] Des informations plus détaillées sur le projet sont fournies sur : http://lbdwww.epfl.ch/e/research/amber/

d'intégrité spécifiques aux BDG dans un schéma de données est particulièrement intéressante. D'une manière générale, nous considérons que les spécifications doivent pouvoir être consultées interactivement, en même temps que les schémas [Mustière et al. 2003]. Elles devraient pouvoir être analysées automatiquement pour aider à comprendre les correspondances déclarées.

REPRESENTATION MULTIPLE A L'AIDE DES PVL'S ET DES VUELS

A l'origine des *Plug-in for Visual Language* (PVL's), le formalisme MODUL-R avait été défini pour représenter la dimension spatiale et temporelle des objets dans un modèle Entité-Association. Avec l'apparition du formalisme UML (modèle O.O.), la solution d'origine a évolué en utilisant l'extension prévue dans le méta-modèle d'UML : les stéréotypes. Les PVL's correspondent ainsi à des extensions de modélisation fondées sur des notations graphiques (des pictogrammes) et une grammaire (règles d'usage) qui permettent de représenter la composante spatiale (géométrie) et temporelle (existence et évolution) des objets. Les règles de modélisation sont relativement intuitives. C'est une approche qui privilégie la simplicité de modélisation. Elle préconise de documenter plus en détail les concepts représentés dans les schémas à l'aide du dictionnaire de données associé. Un AGL a également été défini pour aider à concevoir graphiquement les schémas et générer un squelette de code pour des SIG du marché. Il s'agit de *Perceptory*[14].

La représentation multiple en suivant cette approche est fondée sur le concept de *vuel* (*view element*). Celui-ci s'inspire des applications multidimensionnelles spatiales telles que les outils SOLAP (« *Spatial On-Line Analytical Processing* ») [Rivest et al. 2001]. Un *vuel* représente tout élément pouvant apparaître dans une vue SOLAP. Il combine des informations concernant la géométrie d'un objet (ou de sa classe), sa sémantique et sa représentation cartographique (figure 22). On peut donc définir plusieurs *vuels* pour un même objet afin de représenter différentes visions d'une même réalité. Pour une échelle donnée, un des *vuels* associés à chaque objet est sélectionné, l'ensemble formant une vue de base de données.

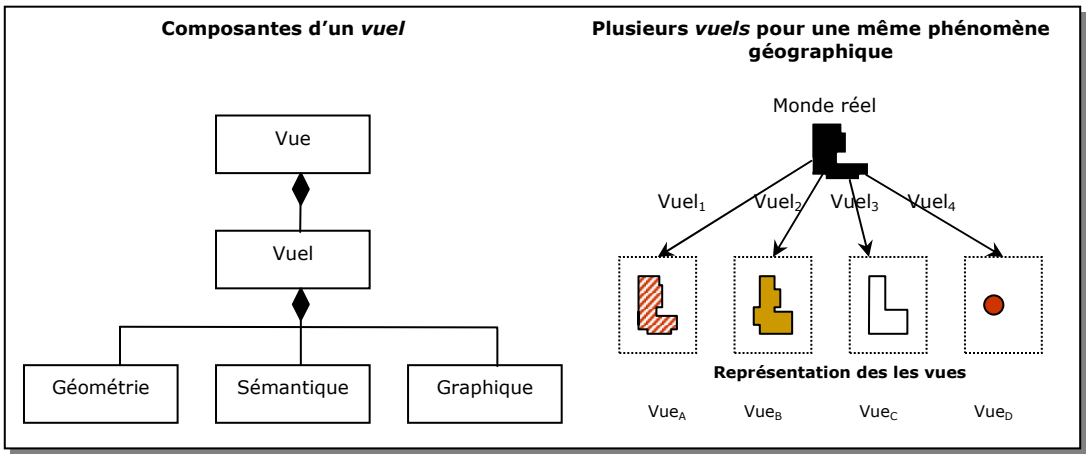


***Figure 22**. Le concept de Vuel pour la représentation multiple (Source : [Bédard et al. 2002])*

La représentation multiple fondée sur les *vuel*s est plus simple que celle proposée dans MADS. Elle est aussi moins complète. La possibilité de stocker des informations sur les schémas dans un dictionnaire de données associé est intéressante. Les

[14] Perceptory est documenté et peut être téléchargé sur le site http://sirs.scg.ulaval.ca/perceptory/

documents générés par ce dictionnaire se rapproche des spécifications des bases de données géographiques.

APPROCHE DE [FRIIS-CHRISTENSEN 2003] POUR LA REPRESENTATION MULTIPLE

Avant de clore cette partie sur les modèles conceptuels de données, mentionnons les travaux de [Friis-Christensen 2003]. L'auteur propose également une approche pour la modélisation de la représentation multiple qui est complémentaire aux précédentes. Plutôt que de définir comment les objets des différentes bases représentant un même phénomène se correspondent, ils décrivent plutôt les correspondances entre ces objets et le phénomène. De ce fait, un nouveau type d'objet est introduit : l'*objet d'intégration* (*i-objet*). Il représente une vision intégrée de l'entité géographique du monde réel (le phénomène). Les objets incarnant cette entité dans les différentes bases (*r-objets*) sont vus comme des *rôles*.

Ce sont d'abord les *r-classes* qui sont modélisées (issues des schémas source). Une *i-classe* est ensuite créée et reliée à ses *r-classes* à travers une association multi-représentations (*mr-association*). Les correspondances d'objets (OC) permettent de spécifier les dépendances d'existence entre l'instanciation d'une *i-class* et ses *r-objets* associés. Autrement dit, les OCs expriment quels sont les *r-objets* requis pour créer un *i-objet* (un objet intégré). Ce sont des règles d'intégration. Il est également possible de définir des contraintes sur les associations pour préciser les conditions de création des *i-objets*. La figure 23 illustre un exemple. Le langage OCL (« *Object Constraint Language* ») est utilisé à cette fin. Les contraintes représentées expriment qu'un *i-objet* n'est créé que si la surface des *r-objets* 'bâtiment' dans la base R_1 est supérieure à 25m² (contrainte), s'il existe au moins un objet de ce type dans R_1 (cardinalité 1), et un objet homologue dans la base R_2 (cardinalité 1 également). Le langage permet aussi de spécifier les correspondances entre les valeurs d'attributs (VC) et des règles permettant d'apparier les *r-objets* (condition pour créer un *i-objet*). A l'issue de cet appariement, il est possible qu'une OC ne soit pas satisfaite. En d'autres termes, l'appariement peut être incohérent : le *i-objet* est incomplet (les correspondances entre les *r-objets* ne sont pas valides). Dans ce cas, il existe des règles exprimant les actions à appliquer. Celles-ci peuvent être diverses : élimination ou insertion de *r-objets*, mises à jour ou transformations plus complexes. L'auteur ne précise pas les connaissances qu'il utilise pour composer ces règles.

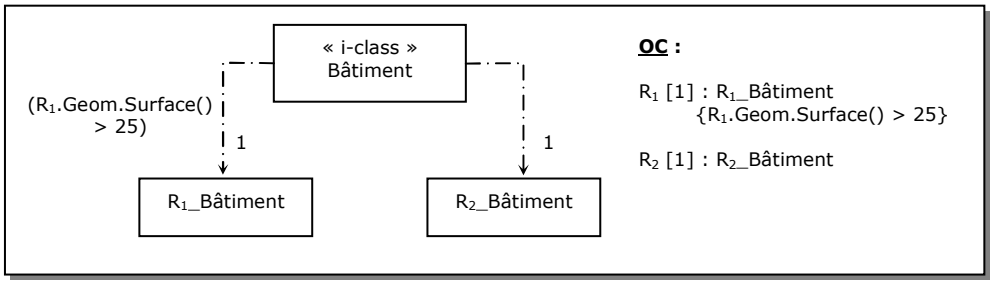


***Figure 23**. Exemple de correspondance d'objet (OC) (Source : Friis-Christensen 2003).*

Cette proposition concerne l'intégration des schémas. La solution qui consiste à définir un *objet intégré* représentant le phénomène géographique est proche de celle de [Gesbert et al. 2004] qui préconisent de créer un objet « Entité Géographique » pour représenter l'entité du monde réel. [Friis-Christensen 2003] définit des règles de cohérence entre *r-objets* mais la cohérence n'a pas le même sens que celui que nous adoptons dans cette thèse. Nous étudions les différences de représentation entre les objets et évaluons la cohérence en tenant compte des spécifications des bases.

L'auteur propose ici de garantir la cohérence entre les *r-objets* par rapport à l'objet intégré (*i-objet*).

A.4.2.3 INTEGRATION FONDEE SUR L'UTILISATION D'UNE ONTOLOGIE

Parmi les approches méthodologiques que nous avons présentées, la plupart traitent l'hétérogénéité sémantique (différence de signification entre concepts). La déclaration des ACI par exemple permet d'en tenir compte. Il existe d'autres contributions spécifiques à cette problématique dont celle de [Rodriguez 2000] notamment. L'auteur propose un modèle, le *Matching Distance*, qui permet d'évaluer la similarité sémantique entre classes d'objets à intégrer. Cette évaluation est réalisée à l'aide d'une « distance » sémantique en s'appuyant sur des ontologies. Le modèle *SFDS* (*Semantic Formal Data Structure*) proposé par [Bishr 1997] est un autre exemple. L'architecture de ce modèle est composée de trois niveaux : le premier comprend les BD et les schémas source, le second est constitué de vues externes sur ces schémas source avec une description du contexte associé, le troisième est le médiateur de contexte composé du schéma fédéré, d'une description de son contexte et d'une ontologie commune. L'évaluation de la similarité sémantique entre les classes est rendue possible grâce à une ontologie commune.

Les ontologies sont de plus en plus utilisées aujourd'hui pour traiter l'hétérogénéité sémantique [Kavouras et Kokla 2000, Ram et al. 2001, Cruz et al. 2002, Fonseca et al. 2002, Visser et al. 2002, Hakimpour 2003, Jaudoin et al. 2003, Morocho et al. 2003, Stoimenov et Đorđević-Kajan 2002, Brodeur 2004, Gesbert et al. 2004]. Nous expliquons ci-dessous à quoi fait référence une ontologie et le rôle qu'elle peut jouer pour l'intégration.

Il existe plusieurs définitions d'un point de vue informatique de la notion d'ontologie. En intelligence artificielle, [Gruber 1993] a définit l'ontologie comme « *la spécification explicite d'une conceptualisation* ». Cette conceptualisation est représentée par un ensemble de concepts, relations, objets et contraintes qui définissent un modèle sémantique d'un domaine [Guarino 1998]. Une ontologie est donc une description explicite de la sémantique des éléments d'un domaine considéré. De ce fait, l'utilisation d'une ontologie est particulièrement adaptée pour résoudre les conflits d'hétérogénéité sémantique puisqu'elle permet la compréhension d'un vocabulaire.

Différentes solutions existent pour identifier et associer les concepts communs des différentes sources en utilisant une ontologie. Trois approches peuvent être adoptées : l'approche globale, l'approche multiple et l'approche hybride [Wache et al. 2001]. Dans la première approche, une seule ontologie globale est définie (figure 24). Chaque source est reliée à cette ontologie globale et la similarité sémantique peut être évaluée en vérifiant que les éléments des sources sont reliés au même concept de l'ontologie. Dans la seconde approche, l'approche multiple, une ontologie locale est définie pour chaque source. Il n'existe pas de vocabulaire commun et un *mapping* entre les ontologies locales est nécessaire (correspondances entre termes égaux ou similaires). L'approche hybride mêle les deux solutions précédentes. Chaque source a sa propre ontologie définie à partir d'une ontologie globale (ou d'un vocabulaire commun). Les ontologies locales sont ainsi plus facilement comparables et l'ajout de nouvelles sources est aisément supporté. C'est l'approche notamment suivie par [Stoimenov et Đorđević-Kajan 2002].

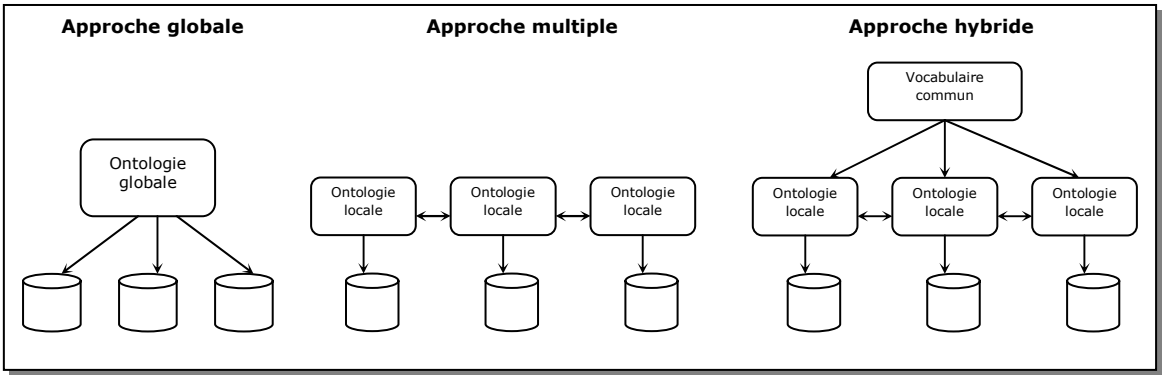


***Figure 24**. Différentes approches peuvent être adoptées pour gérer l'hétérogénéité sémantique à partir d'ontologies (Source : [Busse et al. 2000])*

Les ontologies peuvent être représentées sous différentes formes. Il peut s'agir de réseaux sémantiques qui modélisent un domaine ou une activité. Les logiques de description sont souvent utilisées comme formalisme pour modéliser les ontologies [Wache et al. 2001, Hakimpour 2003]. Ceci s'explique par l'existence de mécanismes d'inférences supportés par ces langages. A l'avenir, on peut imaginer que les schémas conceptuels de données seront suffisamment riches pour constituer une ontologie. Un débat existe à ce sujet, sur les différences existant entre les schémas conceptuels et les ontologies, de même que les liens qu'il est possible d'établir entre ces deux notions [Fonseca et al. 2003]. D'après [Cullot et al. 2003], « [les modèles conceptuels] *sont naturellement de bons candidats pour les ontologies mais doivent être étendus/enrichis afin d'offrir la possibilité de définir de nouvelles entités à l'aide d'axiomes et des outils d'inférence pour vérifier la cohérence des informations et la classifier (mécanisme de subsumption)* ».

A.4.3 TRAVAUX SUR L'INTEGRATION DES DONNEES DE BDG

Toutes ces approches que nous venons de décrire traitaient des schémas. Les travaux qui concernent plus spécifiquement les données se rapportent à l'appariement et au maintien de la cohérence entre représentations multiple (sujet de notre thèse). Nous les situons dans le processus d'intégration à la figure 25 et les présentons ci-dessous.

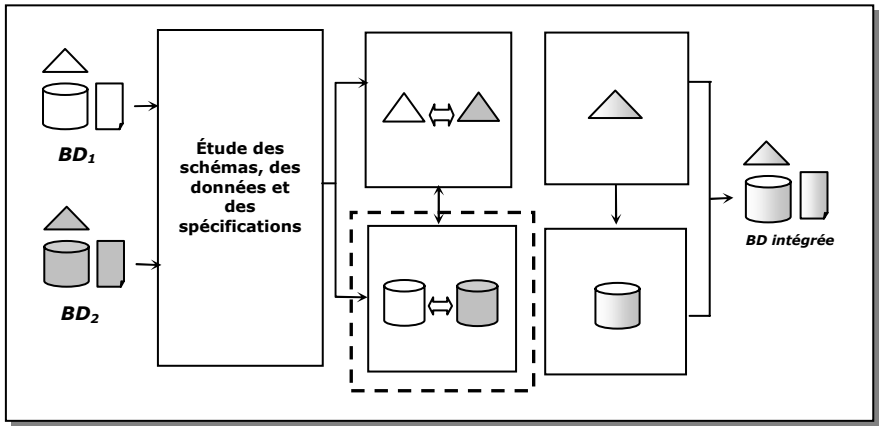


***Figure 25**. Illustration de la position des travaux présentés dans cette partie par rapport au processus d'intégration des BDG.*

A.4.3.1 APPARIEMENT AUTOMATIQUE DE DONNEES GEOGRAPHIQUES

L'appariement de données géographiques désigne le processus qui « *consiste à établir des liens de correspondance entre des ensembles d'entités géographiques symbolisant les mêmes phénomènes du monde réel dans deux représentations de celui-ci* » [Badard et Lemarié 2002, p. 163].

Les techniques d'appariement géométrique ont généralement été proposées dans trois contextes différents :

- Les *contrôles qualité des BDG* : l'appariement doit être mis en œuvre pour permettre la comparaison du jeu de données à contrôler et la référence. On peut citer à ce sujet le travail de [Bel Hadj Ali 2001].
- La *propagation des mises à jour* : l'appariement peut être utilisé dans un contexte de mise à jour lorsqu'aucune trace des modifications entre les différentes versions d'une BDG n'existe. La mise en correspondance des données permet la détection des différences entre les versions et facilite la déduction des évolutions subies [Badard 2000].
- L'*intégration* : comme nous l'avons déjà indiqué, l'appariement géométrique est nécessaire pour mettre en correspondance les données des différentes sources [Devogele et al. 1996, Laurini 1996, Sester et al. 1998, Walter et Fritsch 1999, Pendyala 2002, Dunkars 2003].

Le niveau de complexité du processus et des outils d'appariement est différent suivant le contexte d'utilisation. Dans le cadre des mises à jour, les objets sont définis d'après les mêmes spécifications. Les objets identiques seront facilement appariés et toute différence sera considérée comme une évolution. Pour l'intégration par contre, l'appariement est moins évident car les niveaux d'abstraction des BD sont généralement différents en plus des différences de mises à jour éventuelles. Les outils développés dans ce contexte sont plus complexes.

Les méthodes d'appariement automatique proposées dans la littérature suivent généralement une des stratégies suivantes : stratégie ascendante, descendante ou une combinaison des deux. Dans la première approche, les éléments de base sont d'abord appariés puis reliés en objets plus complexes. Les tronçons de routes par exemple peuvent être d'abord appariés pour ensuite être agrégés et former une route. L'agrégation peut se faire en une fois, lorsque tous les éléments de base ont été reliés indépendamment à leur homologue, ou de manière séquentielle, en appariant un élément et en traitant ensuite de proche en proche les éléments connectés [Gabay et Doytsher 2000]. L'approche descendante adopte la stratégie inverse. Ce sont les objets de haut niveau qui sont d'abord appariés puis les éléments les composant. Enfin, certains auteurs combinent les deux approches (ascendante et descendante), essentiellement pour établir des liens plus rapidement et augmenter la précision des résultats [Pendyala 2002].

En plus de ces différentes stratégies, les méthodes d'appariement peuvent être contextuelles. En effet, il est possible de tenir compte des résultats de l'appariement des éléments voisins pour confirmer ou infirmer l'appariement d'un élément en cours de traitement [Walter et Fritsch 1999].

Les processus auxquels nous faisons référence dans cette partie se fondent essentiellement sur la géométrie des objets. Des ressemblances géométriques et topologiques entre les jeux de données sont calculées de manière indépendante ou coordonnée. Différentes mesures de distance et de forme sont utilisées pour comparer les objets en tenant compte de leur mode d'implantation (point, ligne, polygone). Pour

les relations de proximité, il peut s'agir d'une simple distance euclidienne ou de distances plus spécifiques (Hausdorff, Fréchet, distance surfacique). Pour comparer les formes, différents caractères peuvent être retenus (longueur, sinuosité, compacité, etc.). Généralement, une étape de filtrage est ensuite nécessaire pour affiner le premier résultat de l'appariement et éliminer certains candidats. Des outils topologiques peuvent être utilisés à cet effet (nombre d'arcs entrants et sortants, plus court chemin, détection d'impasse, etc.). Les liens de correspondance sont finalement établis et validés. Leur cardinalité peut prendre les valeurs suivantes : 0-1, 1-0, 1-1, 1-n, n-1, n-m.

Nous donnons à la figure 26 un exemple de résultat d'appariement. On trouvera une description plus détaillée des outils généralement utilisés dans [Lemarié et Bucaille 1998, Badard et Lemarié 2002]. Nous exposerons d'autre part les processus mis en œuvre dans cette thèse dans le chapitre E. L'appariement est une étape centrale dans notre contexte d'évaluation de la cohérence puisque si les données ne sont pas appariées, il n'est pas possible d'analyser les différences.

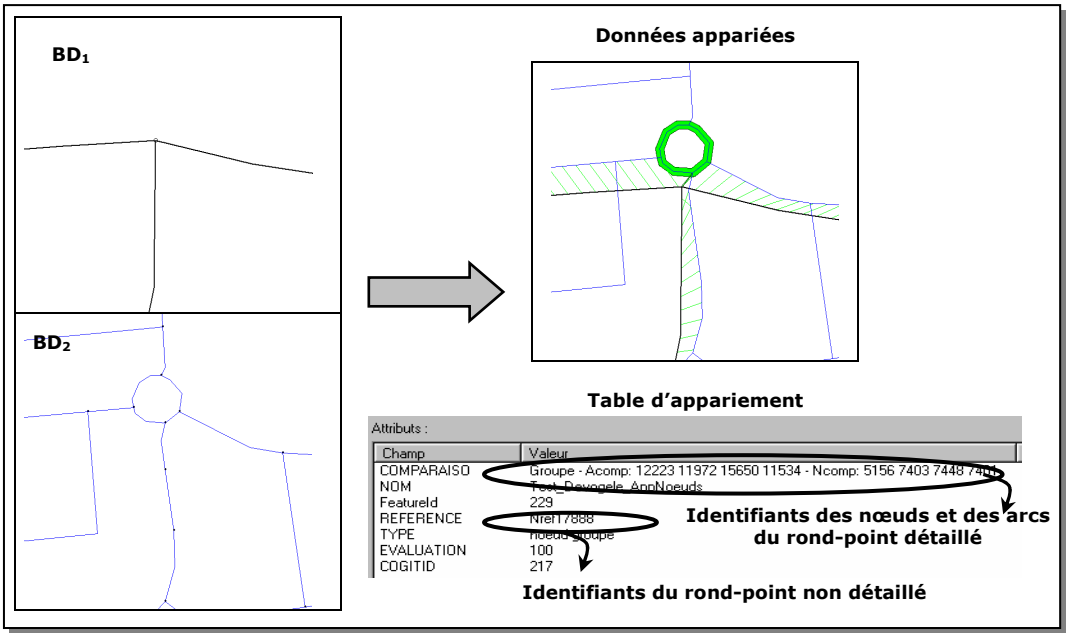


***Figure 26**. Résultat d'appariement de rond-points homologues appartenant à la BDCarto (BD_1) et Géoroute (BD_2). La table d'appariement permet permet de visualiser les identifiants des objets en correspondance. Cet exemple a été obtenu en utilisant les outils d'appariement existant au laboratoire COGIT de l'IGN.*

A.4.3.2 GESTION DES CONFLITS ET MAINTIEN DE LA COHERENCE ENTRE LES DONNEES

Cette partie décrit les contributions qui touchent précisément notre sujet de recherche. Nous avons vu en présentant les spécificités du processus d'intégration des BDG que l'appariement des données devait être suivi d'une étude des correspondances pour évaluer leur conformité. Les conflits de données (différences entre les données) doivent être détectés et justifiés pour garantir une intégration cohérente. Pour préciser à nouveau cette problématique, prenons l'exemple de l'AAC suivante :

$$\textbf{AAC}\ BD_1.\text{TRONCON.Nb\_Voies} = {}_{L(BD2.\text{TRONCON})>1000m}BD_2.\text{TRONCON.Nb\_Voies}$$

Cette assertion indique un conflit de granularité entre attributs des deux BD (d'après la classification de [Devogele 1997]). Le nombre de voies d'un tronçon de la première BD est égal au nombre de voie d'un tronçon équivalent dans la deuxième BD si la longueur du tronçon dans celle-ci est supérieure à 1000m. Il s'agit d'un conflit normal (puisqu'il est ici déclaré au niveau des schémas) et ce conflit est susceptible d'être rencontré dans les données.

Dans l'hypothèse où les longueurs des deux tronçons sont supérieures à 1000m mais qu'il existe une erreur de saisie dans une des deux bases, les valeurs seront différentes pour l'attribut « Nb_voies » alors que l'AAC indique que celles-ci doivent être égales. Il existera donc cette fois une incohérence entre les objets homologues (conflit anormal) qui pourrait apparaître à l'utilisateur lors de la formulation d'une requête impliquant ces objets. Il est donc nécessaire de les détecter et de les traiter pour mener à bien l'intégration.

Il existe assez peu de travaux qui se rapportent à la détection des conflits de données et en particulier, des incohérences, bien que la nécessité de résoudre cette problématique soit identifiée depuis longtemps [Buttenfield et Delotto 1989]. Les contributions les plus nombreuses concernent l'étude des équivalences entre relations spatiales entre objets décrits à différentes échelles. Ces travaux permettent de déterminer si les relations spatiales existant entre les objets d'une base sont cohérentes avec celles apparaissant entre les objets homologues d'une autre base, ces bases ayant leur propre niveau de détail. Nous les présentons ci-dessous.

Une proposition a été faite par [Egenhofer et al. 1994] pour assurer une cohérence topologique entre des données surfaciques représentées à différentes résolutions. Leur approche est fondée sur le modèle des *4-intersections* qui est largement répandu dans la communauté SIG et qui a d'ailleurs été étendu par la suite au modèle des *9-intersections* [Egenhofer et Franzosa 1991, Egenhofer et Herring 1991]. Ces modèles se fondent sur les concepts de topologie ensembliste basée sur les notions d'*intérieur* (noté A°) et de *frontière* (notée ∂A). Ainsi, les auteurs proposent de qualifier l'ensemble des relations topologiques entre deux régions A et B à partir d'une matrice (2x2), la matrice des *4-intersections*, qui est représentée de la manière suivante :

$$I_4(A,B) = \begin{pmatrix} A^\circ \cap B^\circ & A^\circ \cap \partial B \\ \partial A \cap B^\circ & \partial A \cap \partial B \end{pmatrix}$$

En analysant les intersections entre intérieurs et frontières des objets, on indique dans la matrice si le résultat est vide ($\varnothing$) ou non vide ($\neg\varnothing$). On distingue huit configurations topologiques différentes (figure 27).

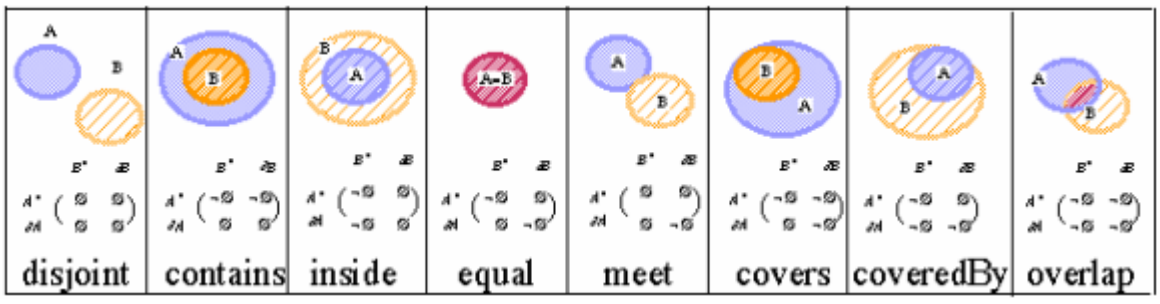


Figure 27. *Interprétation géométrique du modèle des 4-intersections entre deux polygones (Source : [Egenhofer et Herring 1990])*

A partir de ce modèle, [Egenhofer et al. 1994] ont proposé une approche pour maintenir la cohérence topologique entre objets surfaciques à différentes échelles, en introduisant la notion de similarité entre objets et relations. Ces similarités sont exprimées à partir de l'examen des différences d'*invariants*, qui font référence à des propriétés, les *composants*, lesquels permettent d'affiner la description des intersections non vides. Ainsi, différents composants peuvent être définis pour détailler les relations topologiques, notamment : la séquence des intersections entre objets, la dimension des intersections, le type d'intersection (« *touch* », « *cross* », …), le nombre d'enclaves pour les polygones [Egenhofer et Franzosa 1994]. Les degrés de similarité entre relations spatiales sont évalués en étudiant comment les invariants (nombre de composants, séquence des composants notamment) évoluent en passant d'une échelle à une autre, sachant que certaines évolutions ne sont pas permises. Des propriétés d'ordre sont ainsi introduites pour certains invariants. Par exemple, on peut considérer qu'un nombre plus grand de composants dans une scène à plus petite échelle n'est pas normal.

Cette proposition a été étendue aux relations métriques (orientation et distance) en se fondant sur le concept d'évolution graduelle [Bruns et Egenhofer 1996]. Le nombre minimum de transformations nécessaires (le nombre d'arcs) pour passer d'une configuration à une autre à travers un graphe conceptuel reliant les différentes configurations possibles, permet de fixer un degré de similarité qualitatif entre scènes (figure 28). Le concept d'évolution graduelle et le graphe associé sont identiques à la notion de *voisinage conceptuel* mentionné par [Euzenat 1999a] dans le cadre de la représentation des relations temporelles [Freska 1992].

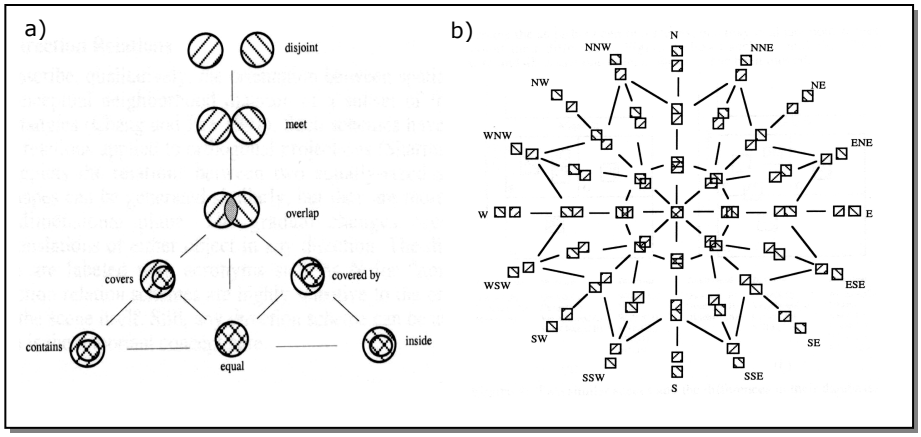


Figure 28. *Graphes conceptuels des relations topologiques (a) et directionnelles (b) entre polygones (Source : [Bruns et Egenhofer 1996])*

[Paiva 1998] propose aussi un modèle permettant de vérifier l'équivalence topologique entre deux scènes à différentes résolutions. Son travail s'inscrit dans la même lignée que les précédents. L'auteur expose le « *relation-based model* » qui se fonde sur la description de scènes sous forme de graphes et sur la recherche des configurations isomorphiques entre ceux-ci (scènes équivalentes). Il présente également une série d'indicateurs de similarité (déviation par rapport à la notion d'équivalence) qui concernent entre autre la dimension spatiale, le nombre d'objets adjacents à un autre et le nombre de niveaux hiérarchiques (nombre de graphes internes). Le degré de similarité du nombre d'éléments adjacents à un objet pour deux scènes homologues, peut ainsi être estimé à l'aide de l'indice suivant :

$$MeetSim_{i,j} = \frac{0-meetSim_{i,j} + 1-meetSim_{i,j}}{2}$$

Les éléments du numérateur sont définis par :

$$0-meetSim_{i,j} = \frac{\min(\#0-meets_i, \#0-meets_j)}{\max(\#0-meets_i, \#0-meets_j)}$$

$$1-meetSim_{i,j} = \frac{\min(\#1-meets_i, \#1-meets_j)}{\max(\#1-meets_i, \#1-meets_j)}$$

« 0-meet » correspond au nombre de relations d'adjacence de dimension 0 (un point) et « 1-meet » correspond au nombre de relations d'adjacence de dimension 1 (une ligne). Un exemple de calcul de cet indice pour deux scènes composées de polygones est donné à la figure 29.

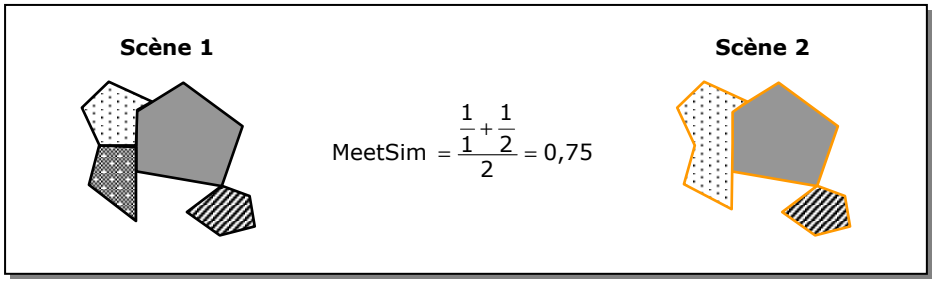


Figure 29. *Exemple d'indice de similarité du nombre d'éléments adjacents entre deux scènes composées de polygones.*

[Goyal 2000] s'est intéressé plus spécifiquement à évaluer la compatibilité des directions cardinales entre objets spatiaux représentés à différents niveaux de résolution, ainsi que leur similarité. Sa méthode d'évaluation de similarité des directions cardinales est fondée sur le calcul d'une distance entre matrices de directions cardinales, utilisant aussi la notion de voisinage conceptuel. Les directions cardinales entre objets sont évaluées qualitativement et représentées par une matrice dont la notation, qui peut être iconique, se rapproche de celle définie pour les relations topologiques (figure 30). Ainsi, pour chaque partition de la matrice, on regarde si l'intersection avec l'objet cible est vide ou non vide. L'auteur a ensuite enrichi cette matrice : chaque intersection non vide prend la valeur 1 tandis que les intersections vides prennent des valeurs calculées à partir de codes si l'objet cible présente des intersections avec les limites des partitions relatives aux directions. La valeur est égale à 0 dans les autres cas.

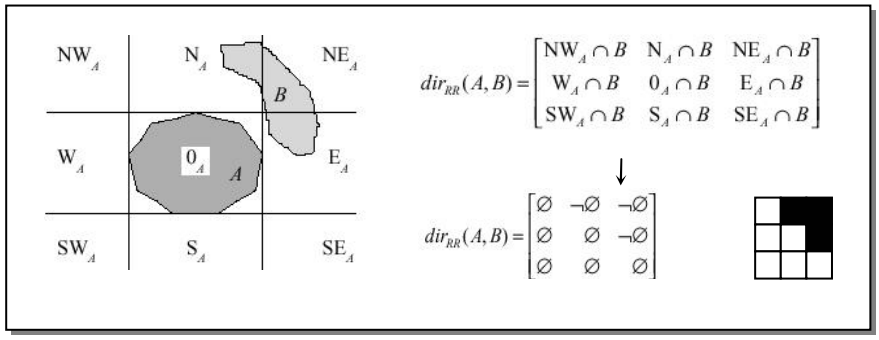


***Figure 30**. La matrice des relations cardinales et sa représentation iconique (Source : Goyal 2000)*

A partir de ce modèle, qui ne se limite pas à des objets surfaciques, l'auteur introduit la notion de compatibilité entre directions à des échelles différentes, sachant qu'un changement d'échelle peut introduire un changement de mode d'implantation des objets (une surface se transforme en un point par exemple). L'évaluation de la compatibilité des directions (et donc des matrices) pour des objets à différentes échelles, n'est valable que lorsque la réduction d'échelle est *significative*. Selon [Goyal 2000], une réduction d'échelle est considérée comme *significative* lorsque la dimension des objets (référence et/ou cible) change. De cette manière, on considère que la direction D^1 est *compatible* avec la direction D^0 (la direction à l'échelle la plus grande) si pour tout élément différent de zéro dans la matrice D^1, il existe un élément différent de zéro dans la matrice D^0.

En plus de cette compatibilité, une mesure de similarité entre directions cardinales à différentes échelles est également proposée. Il s'agit d'une distance qui est définie par le coût minimum de transformation pour passer d'une matrice à l'autre en utilisant le graphe des voisins conceptuels relatif aux neufs directions (figure 31). Pour cette distance, les valeurs numériques des matrices sont exploitées (les codes).

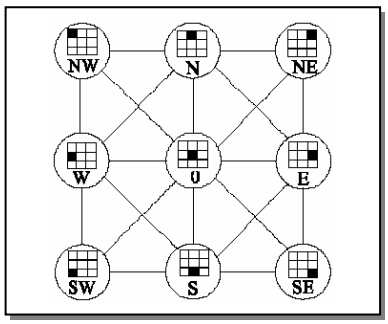


***Figure 31**. Graphe de voisinage conceptuel pour les directions cardinales (Source : Goyal 2000)*

Pour terminer cette partie, mentionnons les contributions de [Abdelmoty et Jones 1997, El-Geresy et Abdelmoty 1998]. Celles-ci sont définies dans le cadre spécifique de l'intégration des BDG. Les solutions proposées pour garantir la cohérence sont toutefois assez similaires aux précédentes. Ce sont les relations spatiales qui sont traitées de manière qualitative et la cohérence totale est supposée si les relations sont identiques.

CONCLUSION

Ces différentes propositions sont utiles pour évaluer la cohérence des relations spatiales mais elles ne précisent pas comment exploiter les valeurs de similarité calculées. En général, on obtient un degré de similarité entre deux scènes en terme de

« distance » qualitative par rapport à des configurations qui seraient identiques mais la difficulté est de pouvoir interpréter cette valeur (en plus de la calculer, ce qui n'est pas immédiat car les relations sont rarement stockées dans les bases). Pour évaluer la cohérence entre les relations spatiales entre objets homologues à différentes échelles, il semble nécessaire de fixer un seuil. Ce seuil correspondrait à la déviation maximale permise lorsqu'on passe d'une échelle à une autre. Mais comment fixer ce seuil ? D'autre part, ces approches introduisent généralement des contraintes d'ordre pour passer d'une échelle à une autre (par exemple, un objet surfacique dans une base détaillée peut être un objet surfacique ou ponctuel dans une base moins détaillée). Les auteurs n'envisagent d'ailleurs les différences de représentation que sous l'angle des différences d'échelle. Ils considèrent qu'il existe toujours des opérations de généralisation cartographique entre deux niveaux de détails différents et ne prennent pas en compte les différences liées à la notion de point de vue.

A.5 BILAN DES RECHERCHES ACTUELLES

Il existe de nombreuses contributions relatives à l'intégration des bases de données, qu'elles soient géographiques ou non. L'intégration des schémas a été davantage étudiée que celle des données. Pour les BD géographiques, les recherches ont généralement bénéficié des travaux effectués dans le cadre des BD traditionnelles, à la fois sur le plan technique et méthodologique. Ceci a d'ailleurs conduit à proposer directement des solutions d'intégration qui tiennent compte de la sémantique des éléments. Ce n'était pas le cas des premières approches d'intégration des BD classiques qui ne traitaient que les conflits structurels.

Du point de vue de l'intégration des données géographiques, le problème du maintien de la cohérence entre plusieurs représentations homologues a été peu étudié. Les contributions pour les données concernent surtout l'appariement. Il existe néanmoins quelques travaux qui traitent de la cohérence des relations spatiales entre objets à différentes échelles. Ceux-ci sont très utiles mais les solutions proposées se limitent généralement aux objets surfaciques et s'appliquent dans un contexte relativement restreint. Ces solutions ont été définies dans le cadre spécifique de la multi-représentations (les différences de points de vue ne sont pas prises en compte). A notre connaissance, il n'existe pas de travaux qui traitent de la cohérence entre représentations dans son ensemble (en tenant compte des différences de modélisation, des différences d'existence, des différences de position, des différences d'attributs, etc.). Notre thèse veut apporter une contribution à ce niveau.

Nous avons mentionné l'importance que nous accordions aux spécifications des BDG pour guider l'intégration et comprendre le contenu des bases. Notre proposition pour évaluer la cohérence repose sur l'utilisation de ces documents. Il semble aujourd'hui que les spécifications n'ont jamais été exploitées de manière formelle pour l'intégration de BDG. Beaucoup d'approches sont fondées sur l'utilisation de métadonnées mais ces métadonnées ne concernent pas les règles de saisie des objets. Il peut s'agir d'informations sur la résolution des bases, sur les systèmes de référence, les unités des données, leur qualité. Ces métadonnées ne permettent pas de comprendre les différences de représentation entre les objets. Au niveau des schémas, certains auteurs préconisent d'utiliser une ontologie. Il est cependant rarement indiqué comment construire cette ontologie. On peut supposer qu'une bonne part de la connaissance exploitée pour traiter l'hétérogénéité sémantique provient directement

des experts du domaine. Notre approche fondée sur l'utilisation des spécifications de manière formelle semble donc n'avoir jamais été adoptée.

Enfin, il existe quelques travaux qui proposent de recourir à l'apprentissage automatique pour faciliter l'intégration. Ceux-ci traitent généralement des schémas. L'apprentissage peut aider à définir automatiquement les correspondances entre les éléments des schémas. Dans cette thèse, nous utilisons aussi l'apprentissage mais pour aider à acquérir des connaissances permettant d'évaluer la cohérence entre les données.

Pour résumer, nous souhaitons apporter une contribution au problème d'intégration de données de BDG et en particulier, au problème de l'évaluation de la cohérence inter-représentations. Nous nous inscrivons dans la même lignée des travaux qui exploitent des métadonnées pour intégrer mais celles que nous utilisons correspondent aux spécifications des BDG. Nous nous inscrivons également dans la même lignée des contributions fondées sur l'utilisation de l'apprentissage automatique pour intégrer mais nous exploitons ces techniques dans le cadre de l'évaluation de la cohérence entre données.

Le chapitre suivant est consacré à la définition de la cohérence que nous adoptons dans cette thèse et à la présentation des spécifications des bases de données géographiques.

CHAPITRE B

REPRESENTATION DES CONNAISSANCES UTILES A L'EVALUATION DE LA COHERENCE

B.1 INTRODUCTION

Rappelons que cette thèse veut apporter une contribution au problème d'intégration des bases de données géographiques et en particulier, à l'intégration des données. Nous traitons l'étape de la mise en correspondance des données et l'étude de leurs différences dans le but de résoudre les conflits anormaux entre les données, à l'issue de leur appariement.

L'approche que nous adoptons repose sur le principe qu'une intégration sémantique des bases de données géographiques requiert à la fois une analyse des schémas et des données, en s'aidant des spécifications. Les spécifications constituent selon nous une source de connaissances essentielle pour réaliser l'intégration. Elles permettent de donner du sens au contenu des bases et constituent les métadonnées de référence pour juger si les représentations des bases qui se rapportent aux mêmes entités du monde réel sont cohérentes entre elles. Nous considérons que les représentations sont cohérentes, et donc que les différences de représentation sont normales, pourvu qu'elles soient justifiées par les spécifications de chacune des bases. Dans le cas contraire, les représentations sont jugées incohérentes.

Dans la première partie de ce chapitre, nous allons exposer plus formellement le problème de la cohérence en nous appuyant sur un modèle de représentation et d'abstraction de connaissances : le modèle *KRA* (« *Knowledge Representation / Abstraction model* ») [Saitta et Zucker 2001, Zucker 2001, Zucker 2003]. Dans la seconde partie, nous présenterons les spécifications. Nous décrirons comment sont représentées ces métadonnées aujourd'hui et exposerons les problèmes que leur représentation pose pour mener l'évaluation automatique de la cohérence. Nous conclurons la chapitre en montrant le besoin de mieux représenter les spécifications des bases pour faciliter leur acquisition et leur comparaison dans notre contexte. Nous montrerons également que les données des bases constituent une seconde source de connaissances intéressante pour mener l'évaluation.

B.2 DEFINITION DE LA NOTION DE COHERENCE ENTRE DONNEES DE BASES DE DONNEES GEOGRAPHIQUES

Nous abordons dans cette partie le problème de la cohérence entre les données de bases de données géographiques en adoptant le modèle *KRA* [Saitta et Zucker 2001, Zucker 2001, Zucker 2003].

B.2.1 CONTEXTE DE RAISONNEMENT ASSOCIE A UNE BASE DE DONNEES GEOGRAPHIQUES SELON LE MODELE *KRA*

NIVEAUX DE REPRESENTATION

Le modèle *KRA* a été initialement défini dans le cadre de l'apprentissage automatique, dans le but de modéliser la notion de *changement de représentation*. Il se prête bien à la formalisation du problème que nous traitons dans ce travail et a d'ailleurs déjà été utilisé pour décrire la tâche de généralisation cartographique [Mustière et al. 2000a].

Dans le modèle *KRA*, un contexte de raisonnement (*R*) distingue quatre niveaux de représentation différents : le niveau de perception (*P*), stocké dans une structure (*S*), dans laquelle sont symbolisées les données pour communiquer avec d'autres agents à travers un langage (*L*), sur lesquelles il est possible de raisonner au moyen de théories (*T*).

Le niveau *P* correspond aux stimulis perçus. Il spécifie la nature des éléments qui constituent le résultat d'une perception [Zucker 2001]. Dans notre contexte, nous l'assimilons au *terrain nominal* qui correspond à une image de l'univers géographique vu à travers le filtre des spécifications [David et Fasquel 1997]. C'est donc la réalité physique terrestre à cartographier, *reformulée* et *abstraite* dans les termes des spécifications.

La structure *S* est une représentation en extension de cette perception. Elle constitue le support à la mémorisation des stimulis perçus. Ce niveau *S* correspond donc à la base de données géographiques elle-même, dans laquelle les objets sont stockés selon une liste de coordonnées et décrits au moyen d'un ensemble d'attributs.

Pour communiquer et décrire de manière symbolique les éléments perçus, un langage *L* est requis. Dans notre contexte, il s'agit du langage cartographique, régi par les règles de sémiologie graphique [Bertin 1973].

Enfin, pour raisonner sur les éléments perçus exprimés dans le langage, une théorie *T* est nécessaire. Nous l'assimilons ici à l'analyse spatiale.

Ce contexte de raisonnement distinguant les quatre niveaux de représentation est illustré à la figure 32.

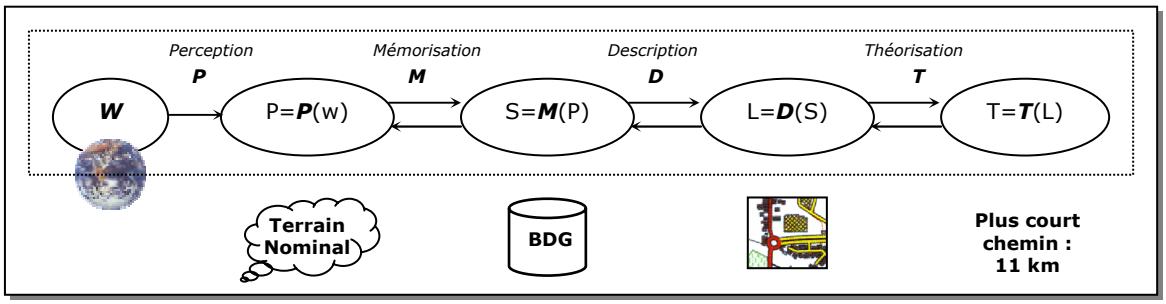


Figure 32. *Le contexte de raisonnement d'une base de données géographiques et ses quatre niveaux de représentation associés définis d'après le modèle KRA. (D'après [Saitta et Zucker 2001, Zucker 2003]).*

CORRESPONDANCES ENTRE LES NIVEAUX DE REPRESENTATION

Il existe une relation de dépendance entre chaque niveau de représentation et celle-ci est illustrée par les différents processus : *perception, mémorisation, description, théorisation* (figure 32).

Dans notre contexte de raisonnement, le processus de perception peut correspondre au processus mental suivi par l'expert du domaine pour conceptualiser l'univers à travers les spécifications. Le niveau *P* auquel il aboutit et qui correspond au terrain nominal est une vision théorique. Ce modèle n'est en effet pas directement accessible : mémorisé, il devrait être assimilé à une base de données au contenu virtuellement parfait, ce qui est impossible en pratique.

La mémorisation peut représenter l'opération de saisie de la base, autrement dit, l'enregistrement du terrain nominal. On pourrait considérer l'existence de deux

processus de mémorisation *M* : un processus théorique M_t, aboutissant à une base virtuelle parfaite (la base qu'on aurait dû produire), et un processus réel M_r, introduisant inévitablement des erreurs dans la base (celle qui est effectivement produite).

Le processus de description peut évoquer la représentation graphique des données stockées dans la base. Suivant le type de base que l'on souhaite créer — *géographique* ou *cartographique* — les opérations et les contraintes d'affichage ne sont pas les mêmes. Une base de données géographiques contient les données brutes. Leur symbolisation n'est pas dictée par les contraintes d'échelle et d'impression. Une base de données cartographiques stocke les données prêtes à être imprimées. Celles-ci ont généralement été transformées (généralisées) et symbolisées pour respecter les contraintes de lisibilité définie pour l'échelle d'impression. Le processus de description est donc différent.

Enfin, la théorisation correspond à la mise en œuvre d'une analyse géographique sur les données représentées, comme la découverte de dépendances spatiales ou la caractérisation d'un espace particulier.

B.2.2 DIFFERENCES ENTRE CONTEXTES DE RAISONNEMENT ASSOCIES A DES BASES DE DONNEES GEOGRAPHIQUES SELON LE MODELE *KRA*

Dans un contexte d'intégration, plusieurs bases de données sont impliquées. En général, ces bases présentent des différences et l'objectif de l'intégration est de les relier de manière cohérente pour tirer profit de leurs singularités respectives. Les bases que l'on intègre possèdent donc chacune leur propre contexte de raisonnement qui les distingue aux quatre niveaux de connaissances (figure 33) : au niveau du terrain nominal, puisque les spécifications sont généralement différentes, au niveau de la structure, puisque les objets mémorisés ne sont pas les mêmes et que les modèles de stockage peuvent différer, au niveau du langage, les objets n'étant pas représentés de la même manière, et au niveau des théories, puisque les réponses aux requêtes formulées sont différentes en raison des différences de contenu et de représentation.

Si les bases de données à intégrer ne possèdent pas la même résolution, des différences d'abstraction peuvent apparaître entre les données. Cette notion d'abstraction a été introduite dans le modèle *KRA*. Une abstraction entre deux contextes de raisonnement est vue comme « *un changement de représentation dans un même formalisme, qui en cachant des détails et en préservant des propriétés désirables, simplifie la représentation du problème initial* » [Zucker 2001, p. 45]. Le passage d'un contexte de raisonnement à un autre se fait par l'intermédiaire d'*opérateurs d'abstraction* qui représentent un type de transformation qui simplifie une représentation. Par exemple, il existe un ou plusieurs opérateurs d'abstraction pour passer du langage $L_1=D_1(S_1)$ au langage $L_2=D_2(S_2)$ si on peut considérer que la deuxième représentation est plus « simple » que la première (figure 33). Dans notre contexte, certains opérateurs d'abstraction correspondent aux opérations de transformation géométrique utilisées pour la *généralisation cartographique* [Mustière et al. 2000a]. Cependant, l'intégration ne se limite pas à des bases présentant des résolutions différentes. On peut vouloir unifier des données de même résolution qui possèdent des points de vue différents sur l'univers à représenter. Il n'existe donc pas toujours une représentation plus simple que l'autre et de différences d'abstraction entre les bases.

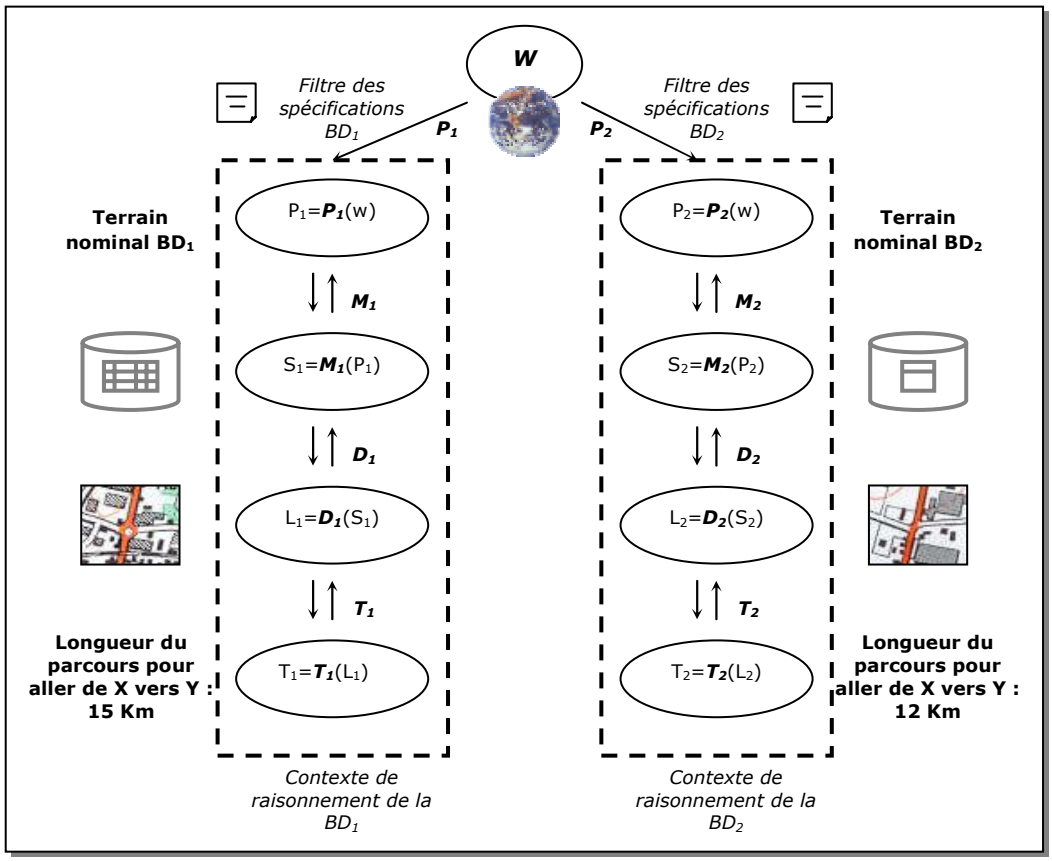


***Figure 33**. Deux bases de données géographiques sont associées à deux contextes de raisonnement différents dans le modèle KRA.*

B.2.3 DETECTION DES DIFFERENCES ET IDENTIFICATION DE LEURS ORIGINES

Les différences qui apparaissent entre les contextes de raisonnement des deux BDG correspondent aux *conflits* dont nous avons parlé précédemment (A.3.2.2.). La détection et l'identification de l'origine des conflits entre les données (différences) constituent précisément les tâches que nous traitons dans cette thèse.

Visuellement, ce sont les différences représentés au niveau du langage *L* qui sont perçus : les différences de représentation géographique. Ces différences sont également constatées chaque fois qu'une mesure est effectuée sur les objets : un calcul de superficie par exemple peut donner une réponse différente suivant la représentation utilisée (niveau *T*). Une détection automatique de ces différences suppose un travail au niveau de la structure *S*. C'est en effet la représentation des objets sous forme de coordonnées, associée à leurs attributs qui est manipulée pour cette détection. L'étude de ces différences requiert enfin une analyse au niveau de la perception *P* : elle met en évidence les différences de spécifications.

Alors que les différences pour les niveaux *S*, *L*, *T* concernent les extensions des bases (les données), les différences au niveau *P* touchent les éléments en intension. Elles font référence aux conflits que l'on déclare entre les schémas, dans les ACI.

B.2.3.1 ORIGINES DES DIFFERENCES DE REPRESENTATION

Dans l'hypothèse où le monde *W* est perçu au même moment (cas de la figure 33), les différences que l'on retrouve entre les niveaux *P* sont parfaitement justifiées : elles reflètent les conceptualisations différentes du monde. Par contre, les différences que l'on retrouve aux niveaux inférieurs (*S*,*L*,*T*) ne sont pas systématiquement justifiées. Des erreurs de saisie sont en effet introduites lors de la constitution de la base provoquant des conflits anormaux entre les données. Par ailleurs, les bases que l'on intègre présentent généralement des actualités différentes. L'hypothèse d'un monde *W* perçu au même instant est donc rarement vérifiée. Ce décalage temporel est aussi susceptible de provoquer des conflits anormaux entre les données.

Les différences de représentation ont donc trois origines : les différences de spécifications, les erreurs et les mises à jour.

REPRESENTATIONS EQUIVALENTES

La plupart des différences que l'on retrouve dans les données sont issus des différences de spécifications. Ces documents dictent le contenu et la représentation des objets dans les bases. Ils sont à l'origine des différences apparaissant entre les terrains nominaux respectifs et par conséquent, entre la plupart des données. Ces différences sont parfaitement justifiés et sont en adéquation avec les conflits déclarés au niveau des schémas.

Dans la suite de cet exposé, nous qualifierons les représentations se justifiant uniquement par leurs spécifications de *représentations équivalentes*. Nous en donnons la définition ci-dessous :

> **Définition 1 (représentations équivalentes).** *Soit O, l'ensemble des objets de la BD_1 et O', l'ensemble des objets de la BD_2. Soit (U,U'), un couple d'objets appariés, U étant un sous-ensemble de O et U' un sous-ensemble de O'. Les représentations du couple d'objets (U,U') sont dites équivalentes, si celles-ci modélisent un monde tel qu'au même instant, U et U' respectent les spécifications de leurs classes respectives et représentent la même entité du monde réel.*

Cette définition sous-entend que les processus de mémorisation mis en œuvre pour capturer les objets dans les bases n'introduisent aucune erreur. Autrement dit, nous pouvons assimiler le processus de saisie M_{1r} de la première base au processus M_{1t}, et le processus de saisie M_{2r} pour la seconde base au processus M_{2t}. Par ailleurs, la définition précise que les représentations du couple d'objets représentent la même entité du monde réel. Cela signifie que les objets constituant le couple ont été correctement appariés.

Nous justifions le choix du terme *équivalence* par le fait qu'entre les deux représentations du même phénomène, il n'y a pas une représentation qui est meilleure que l'autre (pour peu que les deux respectent leurs spécifications). Elles sont définies pour des contextes de raisonnement différents. Le sens d'égalité est donc ici appréhendé en terme de conformité et non en terme de similarité.

REPRESENTATIONS INCOHERENTES

Si des erreurs résident dans les BD géographiques, ce n'est pas faute d'un contrôle qualité. En effet, à l'issue de la saisie, on vérifie généralement à partir d'échantillons que les données mémorisées correspondent au terrain nominal. L'évaluation fait appel à différentes méthodes standardisées et plusieurs paramètres

sont vérifiés : précision et exactitude de position, précision et exactitude sémantique, exhaustivité, cohérence logique. On fournit également des informations relatives à la généalogie de la base et à son actualité [Guptill et Morrison 1995, Goodchild et Jeansoulin 1998].

Au terme de cette évaluation, on ne corrige pas systématiquement les erreurs détectées. En général, on fixe dans les spécifications des objectifs de qualité et on s'assure que ceux-ci sont respectés. Si c'est le cas, on considère que la base est conforme à la qualité attendue. Dans le cas contraire, on procède à une nouvelle saisie des données. De ce fait, il réside toujours des erreurs dans les BDG. On quantifie seulement leur proportion. Précisons qu'en pratique, on mesure la qualité sur des échantillons de données et on définit une unité de volume pour laquelle est estimée cette qualité. Si le taux d'erreur accepté n'est pas respecté, on remet en cause l'unité de volume et non la base entière.

Les erreurs que l'on est susceptible de rencontrer touchent à la fois la géométrie, les attributs et les relations des objets. Il peut ainsi s'agir d'erreurs de position, d'erreurs de forme (superficie, sinuosité,…), de confusions d'objets (on classe une route en chemin), de valeurs d'attributs erronées ou de défauts d'absence ou de présence (déficit ou excédent).

Les causes des erreurs sont également diverses. Elles peuvent être issues d'un systématisme (défaut de graduation d'un appareil de saisie par exemple) introduisant un biais dans une mesure (comme un écart systématique de position). Elles résultent également de fautes introduites par l'opérateur de saisie. Elles sont enfin liées à l'imprécision inhérente du processus de saisie lui-même et à celle des spécifications : il s'agit des erreurs aléatoires.

Ces différentes erreurs créent des conflits anormaux entre les données qu'il est nécessaire d'identifier avant de procéder à l'intégration effective des bases. Nous avons choisi de qualifier les représentations différant à cause d'erreurs de saisie de *représentations incohérentes*.

L'apparition d'incohérences dans les données n'est pas uniquement liée à la présence d'erreurs de saisie. Celles-ci peuvent également découler d'une différence d'actualité entre les BDG. Les bases de données que l'on intègre et les thèmes correspondant (hydrographie, routier,…) sont rarement mis à jour au même moment. La politique d'entretien des données peut être différente. On peut décider d'effectuer une mise à jour suite à un volume de changements trop important. On peut également mettre à jour les données de manière périodique ou continue. Il en résulte, lors de l'intégration des BD, la présence ou l'absence de certains éléments dans une des bases, une représentation différente ou des valeurs d'attributs différents.

Ces différences créent des conflits normaux ou anormaux. Tous les conflits entre les données liés à des différences de mises à jour ne sont effectivement pas toujours anormaux. Il est possible qu'à l'issue d'une mise à jour, les représentations homologues restent équivalentes. Imaginons par exemple deux jeux de données à intégrer créés au même moment et composés d'un ensemble de bâtiments. Les spécifications des deux jeux de données sont les suivantes : dans le premier jeu, les bâtiments sont saisis si leur superficie est supérieure à 100m², et dans le second, ils sont représentés à partir de 150m². Imaginons l'apparition de nouveaux bâtiments de 120m² sur le terrain. Si le premier jeu de données est mis à jour, ces bâtiments seront saisis. Cette mise à jour aura pour conséquence de faire apparaître de nouvelles différences entre les jeux de données (présence des nouveaux bâtiments dans l'un et absence dans l'autre) mais d'après les spécifications du second jeu, cette

absence sera considérée comme normale et les conflits de données seront justifiés. Les représentations seront considérées comme étant équivalentes. Par contre, si les nouveaux bâtiments avaient eu une taille de 180m² et que seul le deuxième jeu de données avait été mis à jour, les conflits de données n'auraient pas pu être justifiés par les spécifications. Les représentations auraient été dans ce cas incohérentes.

Ceci nous amène à définir les représentations incohérentes de la manière suivante :

> **Définition 2 (représentations incohérentes).** *Soit O, l'ensemble des objets de la BD_1 et O', l'ensemble des objets de la BD_2. Soit (U,U'), un couple d'objets appariés, U étant un sous-ensemble de O et U' un sous-ensemble de O'. Les représentations du couple d'objets (U,U') sont dites incohérentes, si celles-ci modélisent un monde tel qu'au même instant :*
> - *soit U ne respecte pas ses spécifications ;*
> - *soit U' ne respectent pas ses spécifications ;*
> - *soit U et U' ne représentent pas la même entité du monde réel.*

Ainsi, la définition prend en compte les erreurs de saisie, les différences anormales introduites par les rythmes de mise à jour différents, mais aussi les erreurs produites par le processus d'appariement. Ce processus fournit dans certains cas des correspondances erronées et il s'agit d'en tenir compte dans l'interprétation des différences.

B.2.3.2 APPROCHE SUIVIE POUR LA DETECTION DES INCOHERENCES

L'étude de la conformité des différences de représentation entre les données ne peut pas suivre une approche classique de contrôle qualité car les connaissances disponibles ne sont pas les mêmes et l'objectif est différent. Nous exposons ses caractéristiques ci-dessous.

DETECTION D'ERREURS DANS UN CONTEXTE DE CONTROLE QUALITE

Dans le cadre d'un contrôle qualité, deux jeux de données sont comparés : un jeu de référence (S_t) et un jeu à contrôler (S_r). Les deux jeux de données répondent aux mêmes spécifications. Il existe donc un terrain nominal commun P, mais on considère deux processus de saisie différents. On suppose que le jeu de données à contrôler contient des erreurs. Il suit un processus M_r. On considère par contre que le jeu de référence est juste, sans erreur. On assimile ce processus à M_t[15] (figure 34).

L'objectif du contrôle qualité est de détecter les erreurs dans le jeu S_r. Le principe suivi est de comparer S_r à S_t et de considérer toute différence comme une erreur, les spécifications des jeux de données étant les mêmes. De cette manière, aucune connaissance supplémentaire n'est nécessaire. La qualité du jeu S_r est ainsi estimée en mesurant l'écart de ses données à celles de la référence.

[15] Le jeu de référence constitue une estimation du terrain nominal mais il le représente plus fidèlement que le jeu de données à contrôler. Les données de référence sont saisies avec davantage de précision ou sont issues de sources possédant une résolution plus fine.

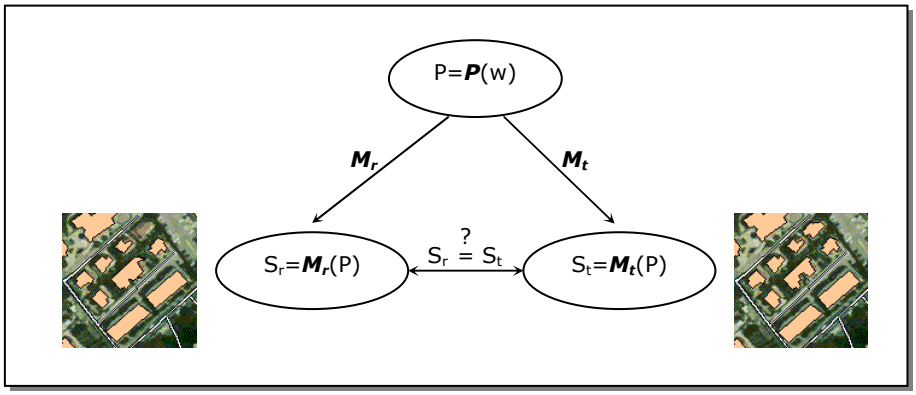


***Figure 34**. La détection d'erreurs dans un contexte de contrôle qualité traditionnel*

DETECTION D'INCOHERENCES DANS UN CONTEXTE D'INTEGRATION

La détection d'incohérences dans le cadre de l'intégration doit être abordée d'une manière différente. Les connaissances dont nous disposons ne sont effectivement pas les mêmes (figure 35). D'abord, les terrains nominaux sont différents (P_1 et P_2). Ensuite, on suppose que les processus de saisie des deux bases sont entachés d'erreurs (M_{1r} et M_{2r}). Une différence de représentation n'est donc pas systématiquement vue comme une erreur dans une des bases mais chacune de celles-ci peut en contenir. De ce fait, une comparaison des représentations ne suffit pas pour détecter les erreurs : il n'existe pas une base de référence (un terrain nominal mémorisé sans erreur). Des connaissances sont cette fois nécessaires pour évaluer la conformité des représentations : ce sont les spécifications.

Par conséquent, à l'issue de ce contrôle, nous ne pouvons pas certifier que les représentations des objets modélisent les bons phénomènes du monde réel et tous ceux qui doivent l'être, comme c'est le cas lors d'un contrôle qualité traditionnel. Nous n'avons pas accès au monde réel. Si les représentations respectent leurs spécifications, on peut supposer que le terrain nominal a bien été mémorisé mais ce n'est qu'une hypothèse.

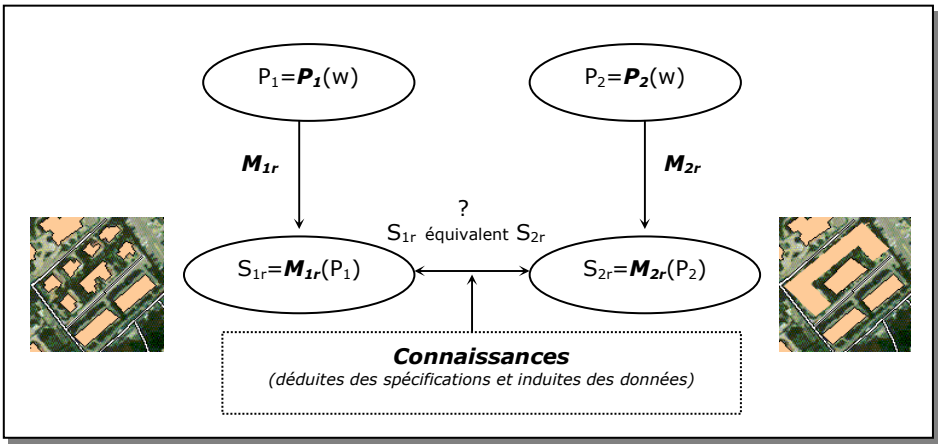


***Figure 35**. La détection d'incohérences dans un contexte d'intégration*

A l'issue du processus d'évaluation, chaque correspondance peut être qualifiée d'incohérence ou d'équivalence. Dans le cas d'une incohérence, il est possible d'affiner l'interprétation et de préciser la représentation erronée mais il est généralement nécessaire de faire l'hypothèse qu'une des deux bases est meilleure que l'autre (sans

pour autant prendre toujours la même base comme référence). Nous y reviendrons plus en détail au chapitre C.

Les spécifications fournies par les producteurs de données sont une source importante de connaissances pour l'analyse des différences de représentation. Mais les connaissances que nous préconisons d'utiliser ne sont pas toutes issues de documents existants. En effet, les spécifications papier ne contiennent pas toujours toute l'information nécessaire pour mener à bien l'évaluation. Il existe également des connaissances implicites dans les données. Il s'agit de connaissances qui n'ont pas été formalisées dans les documents mais qui sont utilisées par les experts lors de la saisie des éléments dans la base. Nous considérons que les données constituent également une autre source de connaissances importante pour étudier la cohérence des représentations. Nous discutons de ces connaissances dans la partie suivante.

B.3 CONNAISSANCES POUR L'EVALUATION DE LA COHERENCE

B.3.1 CONNAISSANCES DEDUITES DES SPECIFICATIONS DES BDG

B.3.1.1 REPRESENTATION INFORMELLE DES SPECIFICATIONS

Les spécifications des bases de données géographiques — nous faisons référence ici à celles décrites dans les documents — constituent la description détaillée du contenu d'un produit. Ainsi, les spécifications décrivent les règles de sélection des objets dans la base et la manière de les représenter. Elles sont destinées aux opérateurs chargés de la production de la base, c'est-à-dire les restituteurs qui saisissent les données à partir de photographies aériennes et les géomètres qui complètent cette saisie en récoltant des informations supplémentaires sur le terrain. Une version simplifiée des spécifications est également fournie aux utilisateurs de la base. Elles leurs permettent d'évaluer en partie l'adéquation du produit à leur besoin, et constituent des métadonnées sur ce produit.

Les spécifications des BDG, dans le cas de l'IGN, se caractérisent par des documents volumineux (plusieurs centaines de pages) comprenant des connaissances déclaratives et procédurales, sous forme de texte. Déclaratives parce qu'elles précisent ce que sont les objets de la base (le « *quoi* »). Procédurales car elles indiquent aussi la manière de saisir ces objets (le « *comment* »). Les spécifications sont découpées selon les classes du schéma conceptuel de la base qui fait d'ailleurs partie intégrante de ces spécifications[16]. A chaque classe de la base correspond ainsi une fiche de spécifications qui présente une certaine structuration.

Chaque fiche relative à une classe est composée de plusieurs parties, c'est du moins le cas pour la BDPays de l'IGN (figure 36).

[16] Dans cette thèse, nous séparons les spécifications et les schémas des BDG.

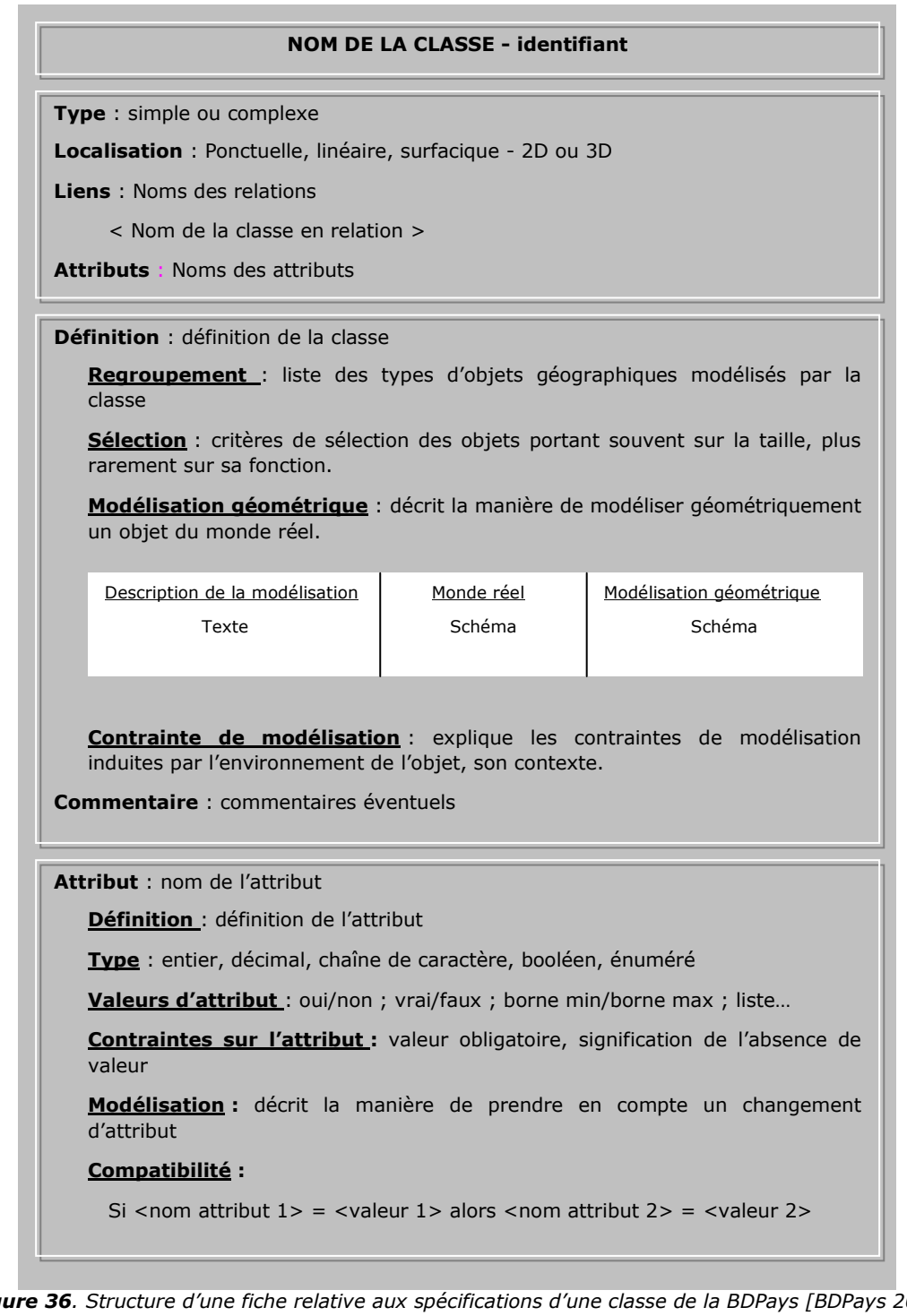
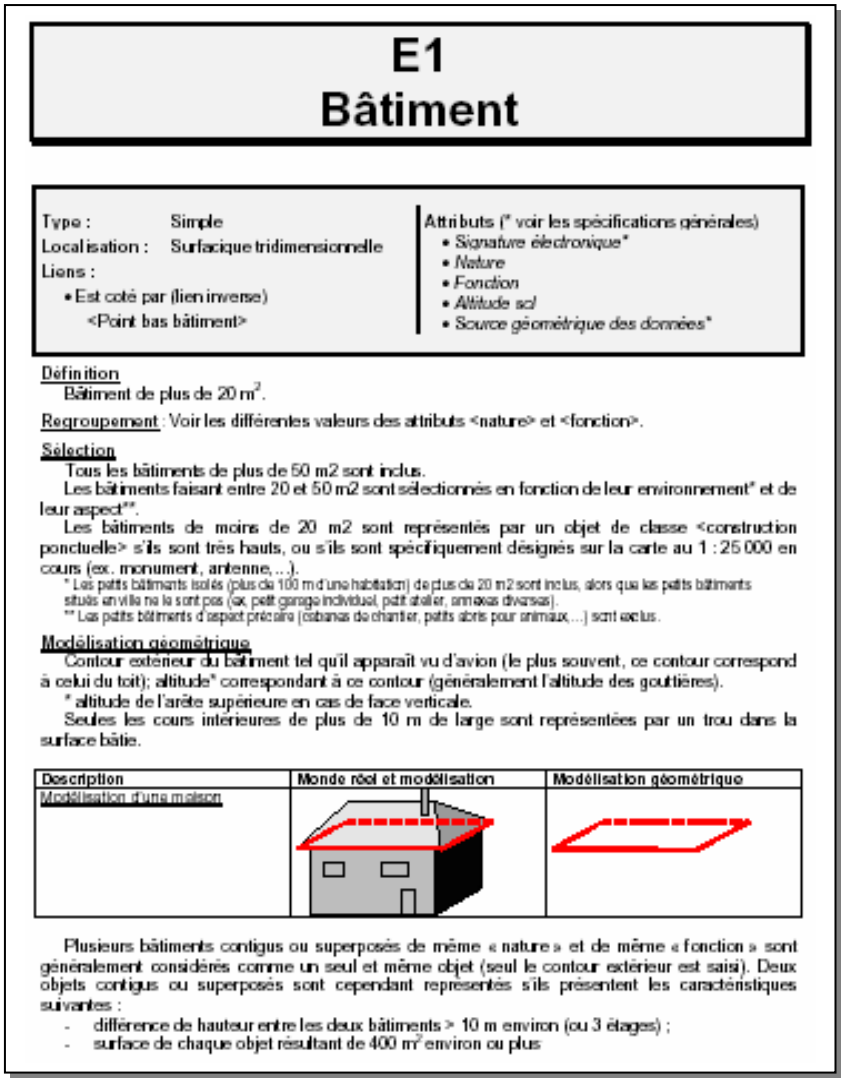


***Figure 36**. Structure d'une fiche relative aux spécifications d'une classe de la BDPays [BDPays 2001].*

La première première partie de la fiche correspond au nom de la classe avec son identifiant. La deuxième reprend les informations que l'on retrouve dans le schéma conceptuel : on mentionne le type de géométrie de l'objet, la liste des attributs que possède la classe et les relations avec les autres classes de la base. Les parties suivantes sont consacrées à une description plus précise des objets de la classe et leurs attributs. Une définition de la classe est d'abord mentionnée. Les conditions de sélection des objets, la manière de les représenter et les contraintes relatives à cette modélisation sont ensuite fournies. On trouve par exemple des contraintes concernant l'existence des objets dans les termes suivants : « *un poste de transformation est saisi s'il est situé sur le réseau de lignes à haute ou très haute tension* » ; « *tous les*

bâtiments de plus de 50 m² sont saisis ». Les contraintes relatives à leur modélisation peuvent être exprimées de la manière suivante : « *Le mode d'implantation du poste est surfacique. On saisit le contour du poste, au sol lorsque le poste est délimité par un grillage, ou en haut des bâtiments lorsque ceux-ci constituent la limite du poste* » ; « *Le mode d'implantation des bâtiments est surfacique tridimensionnel. On saisit le contour extérieur du bâtiment tel qu'il apparaît vu d'avion (le plus souvent, ce contour correspond à celui du toit). Seuls les contours intérieurs de plus de 10 mètres de large sont représentés par un trou dans la surface bâtie.* ». Les attributs sont enfin définis avec leur domaine de valeur, leur sélection et leur regroupement éventuel : « *l'attribut 'nature' du bâtiment peut prendre la valeur 'serre' qui regroupe les serres et les jardineries. Seules les serres de 20m de long sont concernées ».* Des contraintes de comptabilités sont parfois formulées : « *si l'attribut 'nature' du bâtiment a pour valeur 'église', alors l'attribut 'fonction' prend la valeur 'religieuse'* ». Nous donnons un exemple de fiche remplie concernant la classe « Bâtiment » de la *BDPays* de l'IGN en figure 37.



E1
Bâtiment

Type : Simple
Localisation : Surfacique tridimensionnelle
Liens :
• Est coté par (lien inverse)
<Point bas bâtiment>

Attributs (* voir les spécifications générales)
• *Signature électronique**
• *Nature*
• *Fonction*
• *Altitude sol*
• *Source géométrique des données**

Définition
Bâtiment de plus de 20 m².

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 m² environ ou plus

Figure 37. *Extrait des spécifications d'une classe de la BDPays.*
(Source : [BDPays 2001])

En plus de ces fiches relatives aux classes de la BDG, les documents contiennent en préambule les *spécifications générales de contenu* et les *spécifications générales de qualité*. Les premières exposent le contexte de la base (pourquoi constituer une telle

base, quel est son objectif, à qui s'adresse-t-elle,…). Elles précisent également son extension géographique et son découpage (par région, par département,…). Elles mentionnent enfin le référentiel géodésique (ellipsoïde et système géodésique national, projection,…), la référence temporelle (date de prise de vues, date du passage sur le terrain,…) et éventuellement la politique de sa mise à jour (par département, cycle unique pour tous les thèmes de la base,…). Les spécifications de qualité fixent quant à elles les exigences de qualité que la base doit atteindre. On précise par exemple que le taux d'erreurs lié aux confusions des objets de telle classe avec tout autre objet doit être nul ou ne pas dépasser 5%. On indique encore que l'écart moyen quadratique (indicateur de l'exactitude de position) ne peut pas dépasser 2m. Ces exigences ne concernent généralement pas toutes les composantes de qualité [Guptill et Morrison 1995]. Elles se limitent souvent à l'exactitude de position, l'exactitude sémantique et la complétude.

Les spécifications constituent des métadonnées relatives à la base mais elles ne prennent pas en compte tous les éléments prévus dans les standards de métadonnées (comme la norme ISO/TC211 par exemple ou la norme américaine du *Federal Geographic Data Committee*). Ceci s'explique par le fait que le rôle premier des spécifications n'est pas de documenter la base pour les utilisateurs (même si une partie est fournie) mais plutôt de définir les règles pour la constituer. Elles concernent avant tout les producteurs de données et sont fixées avant la saisie de la base. Les métadonnées définies dans les standards visent plutôt à fournir aux utilisateurs des informations sur le produit qu'ils achètent. Elles sont élaborées à l'issue de la saisie. Les spécifications se rapprochent davantage de la notion de *dictionnaire de données*[17] bien que les dictionnaires intégrés que l'on retrouve dans les SGBD relationnels ont une portée beaucoup moins grande. Le contenu est plus proche de celui qui peut être généré par un outil comme PERCEPTORY, l'AGL supportant la modélisation des BD spatiales à l'aide des PVL's (cf. chapitre précédent). On pourrait enfin considérer les spécifications comme une *ontologie* puisque qu'elles décrivent la manière d'aboutir à une conceptualisation du monde, le terrain nominal.

B.3.1.2 DIFFICULTES D'UTILISATION DES SPECIFICATIONS

L'utilisation des spécifications dans notre contexte semble relativement naturelle. Les documents sont particulièrement riches et constituent une source d'informations importante sur la base. Néanmoins, en pratique, leur exploitation n'est pas immédiate et aisée. Plusieurs difficultés doivent être surmontées pour en tirer profit dans le cadre de l'intégration. Du point de vue de l'automatisation, le facteur le plus limitant est lié au fait qu'elles sont décrites en langue naturelle dans des documents papier. Bien que la langue naturelle constitue un moyen très simple et très puissant pour les exprimer, elle rend difficile leur traitement automatique. Mais les spécifications présentent d'autres caractéristiques rendant difficile leur utilisation. Nous en discutons ci-dessous.

Précisons d'abord un point important : les spécifications sous forme de document n'existent pas systématiquement. Les règles d'acquisition des données ne sont effectivement pas toujours formulées explicitement. Il arrive que des BDG soient constituées sans définir de documents relatifs à leur saisie (ou alors de manière très sommaire) ni même de schémas conceptuels [Pantazis et al. 2002]. Seuls les opérateurs de saisie détiennent dans ce cas le savoir et savoir-faire sur la base.

[17] Le dictionnaire de données regroupe un ensemble d'informations relatives aux données, à leur type, format, les droits d'accès, etc. Au niveau logique, le dictionnaire peut correspondre à une méta-base.

Toutes les connaissances sont alors implicites. Cette conception sans la définition explicite de spécifications est possible lorsque la base n'est saisie que par un petit nombre de personnes.

SPECIFICATIONS IMPRECISES ET IMPLICITES

Les spécifications renferment de nombreuses déclarations imprécises et implicites. Cette particularité est liée à plusieurs aspects : l'usage de la langue naturelle, la nature des données à laquelle se réfèrent les spécifications (l'espace), et le contexte d'utilisation pour lequel ces spécifications sont destinées.

Ainsi, le caractère vague de certaines spécifications réside d'abord dans le fait qu'elles sont énoncées sous forme textuelle. De nombreux de qualificatifs du langage naturel sont flous et présentent des caractères de généralités. Qu'entend-on par « près », « nombreux », « trop », « principaux », « petit » ? Ces termes comme bien d'autres sont fréquemment utilisés dans les spécifications. On trouve par exemple des déclarations du type : « *les petits bâtiments d'aspect précaire sont exclus* » ou « *si les sentiers sont trop nombreux, seuls les principaux sont retenus* » ou encore « *les points d'eau sont sélectionnés s'ils sont hors d'une ville importante* ». Dans certains cas, une représentation schématique de la réalité est fournie mais elle laisse également une bonne part d'ambiguïté. Le schéma représenté en figure 38 concerne les ronds-points. On peut comprendre qu'il y a deux représentations pour cet objet (en fonction du diamètre), mais faut-il considérer que cette contrainte ne s'applique qu'à des ronds-points *circulaires* ?

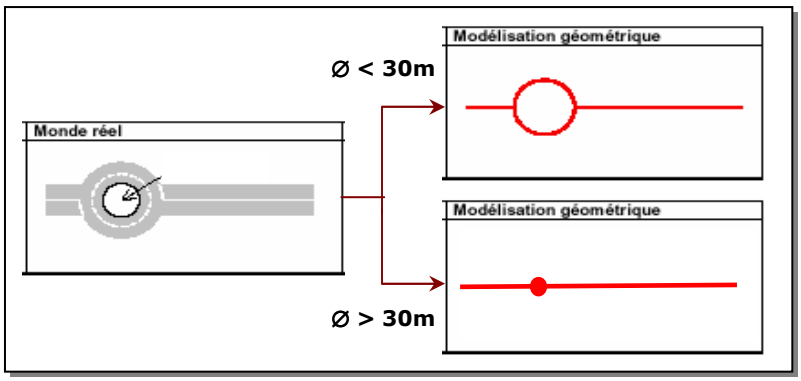


Figure 38. *Une représentation schématique des critères de saisie peut introduire des ambiguïtés dans les spécifications : la contrainte s'applique-t-elle uniquement aux ronds-points circulaires ? (Source : [BDPays 2001])*

Si les spécifications présentent ces imprécisions, c'est aussi parce qu'elles sont destinées à des personnes possédant une expertise importante du domaine. Il n'est pas toujours nécessaire d'être plus précis dans les documents pour la production des données. Les opérateurs de saisie ont acquis au fil du temps un tel savoir-faire qu'il leur est facile d'interpréter les règles mentionnées. Ces règles les guident dans leur choix mais une part importante d'interprétation leur est laissée : « *toutes les lignes de transport par câbles de plus de 100m de long sont saisies, exception faite des installations sommaires servant uniquement à transporter du matériel* » ; « *les cours d'eau temporaires artificiels ou artificialisés sont saisis en fonction de leur importance et de leur environnement* » ; « *les constructions de moins de 20m² et de moins de 50m de haut sont incluses lorsque leur taille ou leur forme font d'elles des constructions à la fois bien identifiables et caractéristiques dans le paysage* ». La sélection de ces objets dans la base ne sera pas la même suivant la personne chargée de la saisie, son expérience et éventuellement la connaissance qu'elle a du terrain.

Une imprécision est donc tolérée. C'est même le cas des spécifications pour lesquelles un seuil précis est fixé. Le seuil de 100m relatif aux lignes de transport par câbles par exemple ne sera pas rigoureusement respecté. Un câble de 98 m sera peut-être introduit dans la base car les opérateurs ne mesurent pas précisément les objets. Doit-on alors considérer que ces objets ne respectent pas les spécifications ? Il s'agit là d'un choix d'interprétation.

> *Nous considérons qu'il est nécessaire de tenir compte de cette imprécision et d'interpréter les différences de représentation avec des spécifications qui reflètent les connaissances implicites utilisées lors de la saisie. Il y a dans les données des représentations qui ne respectent pas rigoureusement les spécifications papiers mais qu'on peut considérer comme exactes en raison de l'imprécision tolérée au moment de leur création. Un travail s'impose sur les données pour cette raison : il faut quantifier l'imprécision pour mener une interprétation plus juste, avec des spécifications qui traduisent plus fidèlement le contenu des bases.*

L'imprécision des spécifications s'illustre assez clairement en figure 39. Elle représente le résultat de la saisie de l'occupation du sol par deux producteurs différents, à partir des mêmes sources et en suivant les mêmes spécifications (celles de la BDTopo de l'IGN). Les différences de saisie entre les deux extraits sont évidentes et sont d'ailleurs particulièrement importantes mais la nature des données traitées accentue cette imprécision. L'occupation du sol est un thème particulier car les limites des zones sont par nature assez floues. Comment fixer précisément la limite d'une forêt ou d'une zone de broussailles ?

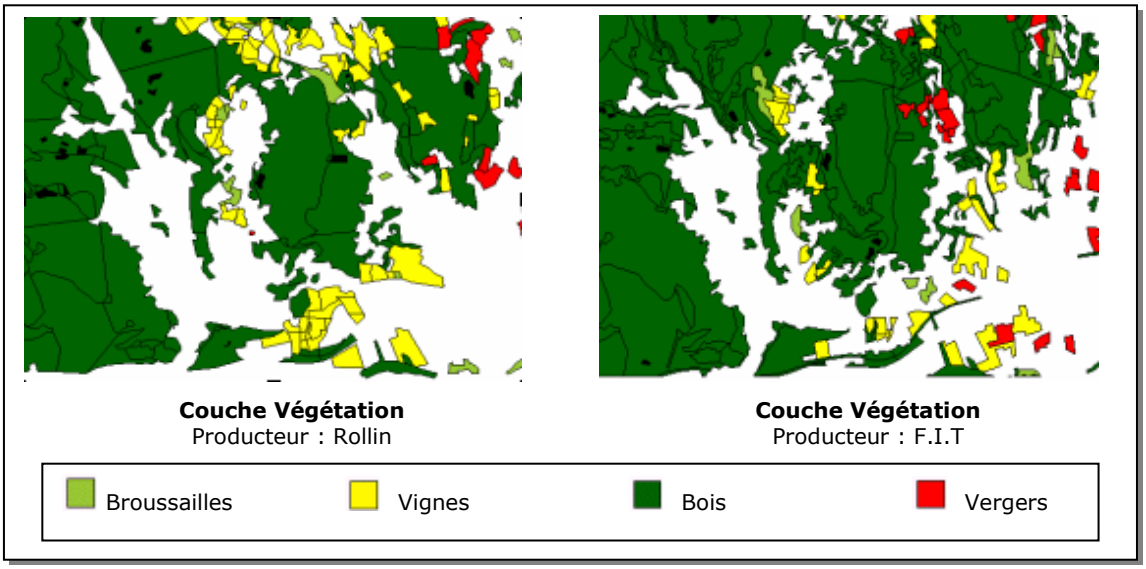


***Figure 39**. Résultats de la saisie de l'occupation du sol par des producteurs différents, en suivant les mêmes spécifications (les données sont ici à la même échelle) (Source : [Vauglin et Bel Hadj Ali 1998])*

L'imprécision peut donc venir du langage utilisé, du manque d'information dans les spécifications mais aussi des phénomènes géographiques eux-mêmes. L'espace géographique est complexe. Il est parfois difficile de le décrire parfaitement. Il contient beaucoup de concepts aux limites floues (une ville, une agglomération, un talus,…) et il n'est pas simple d'en définir précisément le contour de manière univoque (voir à ce sujet l'ouvrage de [Burrough et Franck 1996]). La réalité géographique est également peuplée de nombreux cas particuliers. Des spécifications, aussi exhaustives soient-elles, ne pourront jamais tenir compte de tous ces cas particuliers. Une part de liberté et d'interprétation sera immanquablement laissée aux opérateurs lors de la sélection.

Ces imprécisions vont naturellement apporter des difficultés dans la justification des différences de représentation mais un des enjeux de cette thèse est aussi de clarifier les critères de saisie peu précis. Un enrichissement peut être envisagé en « fouillant » les données de chaque base indépendamment les unes des autres mais surtout, en analysant les correspondances une fois les données appariées. En plus de l'étude de la cohérence des représentations, il devrait être possible d'éliciter certaines spécifications d'une des bases en s'aidant de l'autre base. Si on imagine par exemple que dans la première, les spécifications indiquent que toutes les cours intérieures des bâtiments sont saisies et que pour la deuxième, les contraintes relatives à ces objets sont absentes, on pourrait envisager d'utiliser la première base pour enrichir les spécifications de la deuxième. En analysant les données, on découvrirait par exemple que les cours intérieures ne sont saisies que sous certaines conditions (pour une longueur > 10 m et une largeur > 5 m par exemple).

STRUCTURATION HETEROGENE

Un second aspect important des spécifications est le manque d'homogénéité dans leur description. La fiche que nous avons présentée en figure 36 est propre aux spécifications de la BDPays de l'IGN. Il n'existe pas aujourd'hui une structure type pour élaborer les documents. Chaque ligne de production définit son propre document relatif à sa base de données, qui répond à un modèle spécifique trop peu formalisé pour un contexte d'utilisation automatique comme le nôtre. Les spécifications relatives à la BDTopo (version 3.1.) par exemple ne sont pas composées des mêmes parties que celles présentées pour la BDPays : il existe un cadre réservé au nom de la classe, un autre réservé aux spécifications de contenu dans lequel on précise la définition de la classe, les relations et les attributs, un autre qui s'intitule « spécifications complémentaires » dans lequel les contraintes de saisie sont mentionnées sans organisation particulière, et enfin, un dernier cadre est consacré aux critères de qualité.

La présentation des spécifications laisse également supposer que chaque fiche ne contient que des informations relatives à la classe qu'elle représente, mais comme le soulignent très justement [Gesbert et al. 2004], il n'en est rien. Les informations concernant les objets d'une classe se retrouvent parfois disséminées dans d'autres classes. On retrouve par exemple la relation entre le poste de transformation et la ligne électrique de la *BDPays* uniquement dans la spécification des lignes électriques : « *le dernier point d'une ligne électrique aérienne arrivant sur un poste de transformation est situé à l'intérieur de la surface du poste* ». De même, les spécifications de la BDTopo précisent que « *les bassins ou étangs en bord de mer subissant l'influence des marées sont traités en 'eau marine', seulement si des laisses de plus basses mers et de plus hautes mers sont saisies sur leur contour, dans le cas contraire la surface d'eau est saisie en 'surface hydrographique'* ». Cette information n'est mentionnée que dans la classe « Eau Marine » et pas dans la classe « Surface Hydrographique ».

Ce manque d'homogénéité et de structuration est une cause supplémentaire de l'imprécision des spécifications. Des contraintes sont parfois oubliées ou mélangées parce qu'elles sont peu formalisées. Ceci a pour conséquence de compliquer leur recensement et leur comparaison dans le cas de plusieurs bases. C'est pourtant nécessaire pour étudier les différences entre les données.

Les spécifications renferment un grand nombre d'informations très utiles pour mener l'évaluation de la cohérence automatiquement. Néanmoins, si nous voulons les exploiter dans ce cadre, il est nécessaire d'adopter une autre structuration et un autre

langage de représentation. Par ailleurs, puisque les spécifications peuvent être incomplètes et manquer de précision, nous devons également trouver un moyen de rendre certaines spécifications plus explicites et acquérir l'information indisponible. A cette fin, nous considérons que les données constituent une seconde source de connaissances intéressante à analyser.

B.3.2 CONNAISSANCES INDUITES DES DONNEES

Une partie de l'information que nous devons exploiter pour étudier les différences de représentation se retrouve dans les données. C'est une conséquence de l'imprécision des spécifications mais c'est aussi lié à une caractéristique bien particulière des bases de données géographiques : la présence d'informations implicites.

Il existe un décalage entre ce qui est perçu lorsqu'on observe des données géographiques représentées graphiquement et ce qui est effectivement stocké dans la base de données. La quantité d'informations véhiculées par la géométrie est beaucoup plus importante que celle qui est mémorisée. Par exemple, si on visualise les données de la figure 40, on peut voir qu'une route droite mène à des maisons isolées, qu'une route sinueuse traverse le village ou encore, que la densité de l'espace bâti dans le centre ville est élevée. Dans la base, seules les routes et les maisons sont stockées, avec des coordonnées bien précises. Les caractères droit et sinueux des routes ne sont pas directement accessibles, de même que la notion d'isolement des bâtiments ou de densité des îlots. C'est également le cas de la plupart des relations spatiales entre les objets (la route qui *mène* aux bâtiments). Certaines structures de données prennent en compte la topologie mais les relations métriques par exemple (distance entre les objets, orientation) sont rarement stockées.

Figure 40. *La quantité d'information véhiculée par la géométrie est beaucoup plus importante que celle à laquelle on peut accéder directement en interrogeant une BDG.*

Ce décalage existe également entre ce qui est décrit dans les spécifications et les données elles-mêmes. Les spécifications font souvent référence à des objets du monde réel qui sont implicitement présents dans la base. Par exemple, on retrouve pour la BDTopo des contraintes du type : « *Pour les carrefours en patte d'oie, deux branches ne sont individualisées par la saisie de leurs deux axes qu'à partir du moment où leur écartement au débouché sur l'autre route est au moins égal à 50 mètres* ». Les spécifications décrivent les règles de sélection relatives aux ronds-points de manière analogue (en fixant un seuil), mais les classes de ces objets (« Rond-

point », « Patte d'oie ») n'existent pas dans la BD. Ils sont constitués d'un ensemble de tronçons de route (des arcs) et de carrefours simples (des nœuds). Pour vérifier les spécifications, il faut donc d'abord reconstituer ces objets. Dans le cas des pattes d'oie et des ronds-points, la limite des objets est précise mais qu'en est-il des agglomérations par exemple ou des villes : « *en ville, les grands boulevards n'ont pas toujours leurs voies matérialisées, le nombre de voies saisi est celui qui est réellement utilisé* » ; « *en milieu rural, on indique toutes les pistes cyclables* ». Ces phénomènes (« ville », « milieu rural ») existent aussi tacitement dans la base mais sont bien plus complexes à reconstituer [Boffet 2001].

L'implicite domine encore dans les données en raison de l'espace géographique lui-même. En particulier, l'existence de phénomènes (naturels ou construits par l'homme) est généralement liée à la présence ou l'absence d'autres phénomènes, à leur contexte. Ces dépendances spatiales peuvent être évidentes : s'il y a un cours d'eau, il y a forcément une source ou s'il existe une gare, on s'attend à la présence de voies ferrées. Certaines corrélations spatiales sont moins triviales : la localisation de l'habitat des merles à ailes rouges est davantage liée à la présence de plantes robustes (résistant à l'action du vent) qu'à l'espèce de la plante elle-même [Chawla et al. 2001].

La présence de connaissances implicites rend la vérification des spécifications plus complexe puisqu'il est nécessaire d'extraire une partie de ces connaissances et d'enrichir les données avant d'étudier la cohérence. L'évaluation automatique de la cohérence en suivant une approche fondée sur l'utilisation des spécifications implique donc non seulement d'adapter la représentation des spécifications mais aussi, de trouver une solution pour extraire les connaissances contenues dans les données des bases.

B.3.3 CONNAISSANCES EXTERNES

Il existe une troisième source de connaissances qui peut aider à raffiner l'interprétation des différences de représentation : ce sont les experts du domaine.

Les connaissances de l'expert sur l'espace sont particulièrement requises pour distinguer les incohérences liées à des mises à jour de celles produites par des saisies erronées. En effet, sans connaissance supplémentaire, une incohérence liée à une mise à jour est considérée comme une incohérence produite par une mauvaise saisie. Pour éviter cette confusion, des connaissances relatives à l'évolution des données devraient être utilisées : un nouveau lotissement a été construit dans tel secteur, un ancien site industriel a été démoli à tel endroit, etc. En pratique, on dispose rarement de ces informations. La gestion de l'historique et de l'évolution spatio-temporelle des objets est encore relativement peu développée dans les BD spatiales (on peut se référer à l'article de [Peuquet 2001] pour davantage de détails à ce sujet). On peut toutefois raisonner sur les correspondances incohérentes et dans certains cas, préciser qu'il s'agit probalement d'une erreur ou d'une mise à jour. Pour ce raisonnement, on fait appel à des connaissances géographiques générales en tenant compte des dépendances spatiales et on procède à des suppositions logiques.

Imaginons par exemple (figure 41) la présence d'une série de bâtiments dans une base et leur absence dans l'autre (les bâtiments dans cette deuxième formant une zone d'espace bâti). Au regard des spécifications (on ne les détaille pas ici), certaines absences de bâtiment sont injustifiées. Un quartier entier n'a pas été saisi. Il s'agit d'une incohérence.

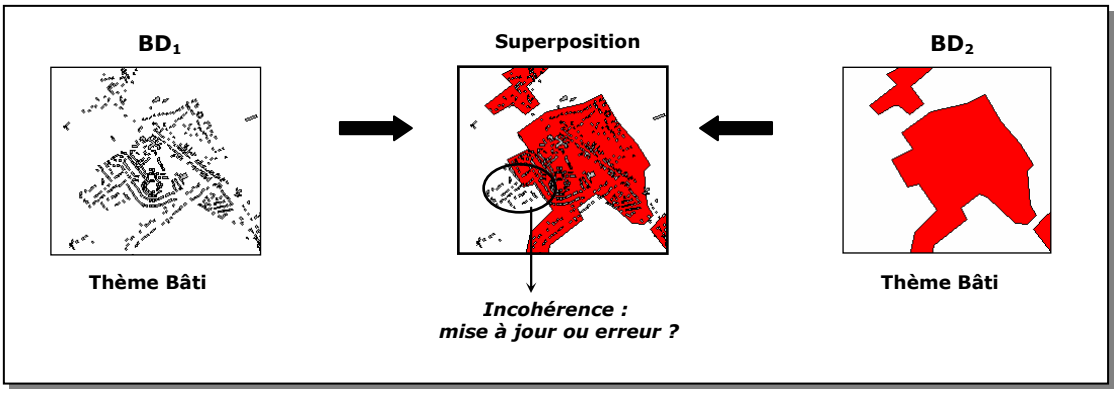


***Figure 41**. Différencier les mises à jour des erreurs requiert l'utilisation de connaissances externes.*

Avec les connaissances dont on dispose sur les données, il n'est pas possible d'approfondir cette interprétation. On ne peut pas savoir si ces incohérences sont liées à des erreurs de saisie ou des mises à jour. Pourtant, si on regarde les données, on peut objectivement penser qu'il s'agit très certainement de mises à jour. Il est difficile d'imaginer que l'opérateur de saisie ait oublié ces bâtiments lors de la construction de la zone d'habitat, ceux-ci formant un bloc attenant avec les autres bâtiments eux-mêmes représentés. Vu la forme du bloc, on pourrait songer à une extension d'un lotissement. Par ailleurs, il existe également un certain nombre d'incohérences concernant les routes à cet endroit. Il paraît peu probable que celles-ci soient également le fruit du hasard, conséquence d'oublis successifs. On peut logiquement penser qu'une des bases a été mise à jour et l'autre pas.

On émet donc une hypothèse après un raisonnement qui requiert des connaissances très générales et qui n'est ici applicable qu'en présence de nombreux changements groupés, en exploitant l'information contextuelle des objets. Les correspondances étudiées individuellement ne permettraient pas ici de formuler cette hypothèse.

La capacité de raisonnement de l'expert est également très utile pour préciser dans quelle base réside une erreur, lorsque l'information fournie par les documents ou les données ne suffit pas. Il n'est effectivement pas toujours possible de préciser la représentation erronée pour une incohérence détectée. La figure 42 illustre ainsi une correspondance entre un rond-point de représentation ponctuelle et un rond-point de représentation détaillée.

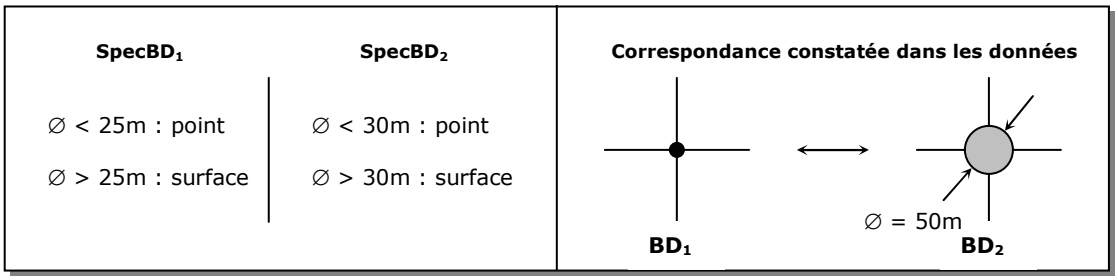


***Figure 42.** Correspondance incohérente*

En se référant aux spécifications, on peut constater que la correspondance est incohérente mais on ne peut pas identifier dans quelle base existe l'erreur. Peut-être que le diamètre sur le terrain est effectivement plus petit que ce que laisse supposer la représentation détaillée dans la BD_2. Dans ce cas, il y a une erreur dans la base BD_2. Au contraire, ce diamètre a peut-être été sous-estimé pour la première base ou

l'objet sur le terrain a été mal interprété. Mais objectivement, on peut supposer que c'est la première base qui contient l'erreur. Il est difficile de penser que l'opérateur de saisie ait inventé le pourtour du rond-point et exagéré ainsi sa taille, la référence sur laquelle il s'appuie étant une photographie aérienne. Cette erreur est possible mais elle est moins probable que la saisie inexacte de la représentation ponctuelle. On peut donc faire l'hypothèse ici que c'est la première base qui contient l'erreur, sans toutefois en être sûr.

B.4 CONCLUSION

Pour juger de la conformité des représentations, nous nous référons aux spécifications. Ce sont les spécifications qui constituent la référence pour déterminer si les représentations sont *incohérentes* ou *équivalentes*. Nous ne considérons pas qu'entre deux bases de niveaux de détails différents, celle qui possède le niveau de détail le plus élevé est meilleure que l'autre. C'est uniquement le respect aux spécifications qui importe pour juger la cohérence. Une telle approche rend l'étude de la cohérence relativement objective puisqu'on se réfère aux critères de saisie des bases. Elle peut permettre également d'améliorer la qualité des bases. Cependant, elle implique de faire face à plusieurs difficultés.

La première difficulté est liée à la représentation actuelle des spécifications. En l'état, les spécifications ne peuvent pas être exploitées pour mener l'évaluation de la cohérence automatiquement. Elles sont décrites en langue naturelle dans des documents papier ce qui empêche leur manipulation automatique. Nous devons donc les représenter en adoptant un autre langage de représentation pour pouvoir les utiliser dans un système informatique.

Ensuite, la structure même des spécifications rend leur analyse et leur comparaison difficile dans notre contexte d'intégration. Les informations peuvent être disséminées à plusieurs endroits dans les documents et la description des règles de saisie n'est pas normalisée. Nous devons donc adapter et formaliser la représentation des spécifications pour rendre leur acquisition plus aisée.

Enfin, la troisième difficulté est liée au contenu des spécifications. Bien que les documents soient très riches et très utiles, ils manquent parfois d'exhaustivité et de précision pour juger la conformité des représentations. Par conséquent, nous devons utiliser d'autres sources de connaissances — nous avons fait le choix d'exploiter les données des bases — et trouver un moyen de les acquérir.

Les deux premières difficultés correspondent à un problème de représentation de connaissances [Kayser 1997]. En intelligence artificielle, la question de la modélisation des connaissances et celle relative au choix du langage à adopter pour représenter des connaissances symboliquement et les rendre manipulables par une machine est un sujet de recherche très important.

Dans cette thèse, nous répondons à ce problème de deux manières. Nous proposons d'abord de représenter les spécifications selon un modèle que nous avons défini. Nous le considérons comme un outil qui facilite l'acquisition des connaissances. Nous proposons également de manipuler ces spécifications automatiquement, grâce à un système-expert. Nous adoptons de ce fait un langage à base de règles de type « *Si…Alors* » comme langage de représentation des spécifications.

La troisième difficulté, le problème de l'extraction de connaissances à partir de données, est un problème d'acquisition de connaissances. Il fait également l'objet de nombreux travaux en intelligence artificielle [Russell et Norvig 2003]. Plusieurs approches existent pour y répondre. Celle que nous avons adoptée est fondée sur l'utilisation de l'apprentissage automatique supervisé [Mitchell 1997].

Les problèmes de représentation et d'acquisition de connaissances sont traités dans le chapitre D. Celui-ci est consacré à la méthode *MACO*.

Il reste à savoir *comment* réaliser l'évaluation de la cohérence à l'aide de ces connaissances. Quelles sont les étapes à suivre pour déterminer si les représentations sont conformes ? Cette question est traitée dans le chapitre suivant qui présente la méthode *MECO*.

CHAPITRE C

MECO : METHODE D'EVALUATION DE LA COHERENCE

C.1 PRESENTATION GENERALE DE LA METHODE *MECO*

C.1.1 INTRODUCTION

Ce chapitre constitue la première partie de notre contribution méthodologique pour l'évaluation de la cohérence inter-représentations. Il est consacré à la description de *MECO*, méthode d'évaluation de la cohérence (figure 43). Cette méthode se veut générique. Elle devrait pouvoir être appliquée quelque soit le type de différences rencontrées (géométrique, attributaire ou relationnel). Elle peut être mis en œuvre pour étudier la cohérence des représentations appartenant à des bases de résolutions et de points de vue similaires ou différents.

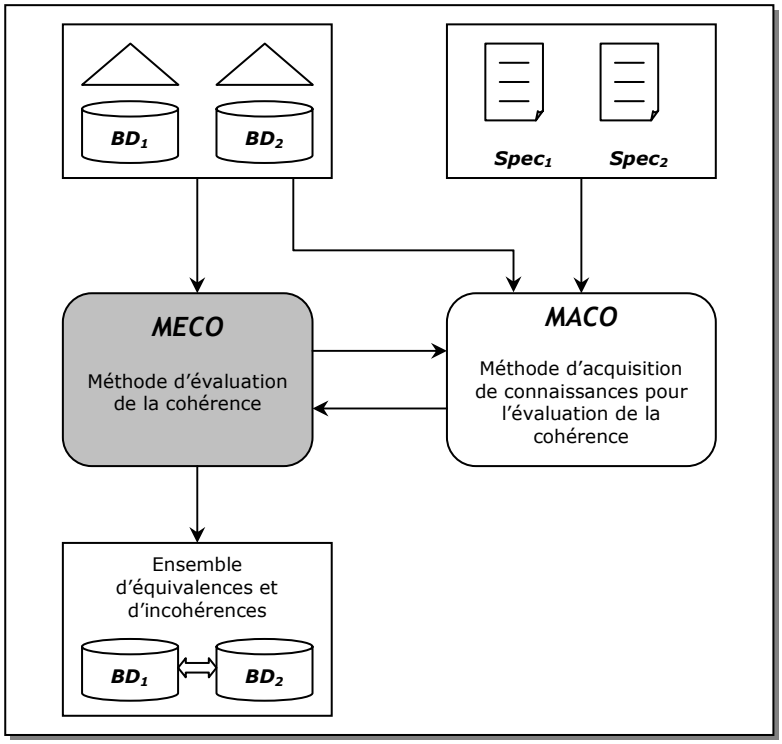


***Figure 43**. MECO : première méthode proposée dans la méthodologie générale d'évaluation de la cohérence inter-représentations.*

Nous supposons que les deux bases à intégrer sont des bases de données géographiques vectorielles. Nous faisons également l'hypothèse que les données sont décrites dans le même système de référence. Une transformation des données dans un système de projection commun peut donc s'imposer avant d'étudier les différences de représentation.

Cette méthode s'applique sur chaque ensemble d'objets des deux bases correspondant aux mêmes phénomènes du monde réel. Autrement dit, toutes les différences ne sont pas traitées en même temps. On choisit par exemple d'étudier les différences entre les routes, puis entre les rivières, ensuite les bâtiments, et ainsi de suite. Nous supposons que les catégories d'objets à mettre en correspondance ont été identifiées au préalable. Nous savons par exemple que la vérification de la conformité des routes entre deux bases à intégrer doit être réalisée en sélectionnant les objets de la classe « Route » de la première et les objets des classes « Route » et « Chemin »

de la seconde. Nous nous appuyons donc sur les assertions de correspondance inter-schémas pour mettre en œuvre la méthode.

C.1.2 LES ETAPES DE *MECO*

La méthode *MECO* est constitué de cinq étapes : l'*enrichissement*, le *contrôle intra-base*, l'*appariement*, le *contrôle inter-bases* et l'*évaluation globale.* Chaque étape de la méthode utilise des outils qui reposent sur des connaissances acquises grâce à l'application de *MACO* (chapitre D)*.* Nous préciserons ces connaissances et leur origine lors de la présentation de chaque étape sans toutefois préciser leur mode d'acquisition.

Si le lecteur souhaite avoir directement une illustration de la mise en pratique de cette méthode, il peut lire parallèlement à ce chapitre les applications décrites au chapitre E.

C.2 ENRICHISSEMENT DES BASES

L'enrichissement constitue la première étape de *MECO*. Il touche à la fois les schémas et les données. Il consiste principalement à extraire des données des informations le plus souvent implicites à travers la géométrie, ceci afin de rendre possible la vérification des spécifications (contrôle intra-base et inter-bases).

Nous considérons l'enrichissement comme une phase de préparation au contrôle de la cohérence des représentations. De ce fait, l'enrichissement est à mettre en relation avec l'étape de pré-intégration (cf. A.3.2.1).

C.2.1 ENRICHISSEMENT ET RESTRUCTURATION DES SCHEMAS

Bien que nous nous intéressions essentiellement aux données dans ce travail, nous devons signaler l'existence d'une phase d'enrichissement et de restructuration des schémas dans la méthode *MECO*. Cette tâche a pour objectif de rapprocher les schémas, préparer la mise en correspondance des données et les contrôles de la cohérence, et réduire l'hétérogénéité des représentations.

La restructuration des schémas peut se traduire par l'éclatement d'une classe en plusieurs classes selon la valeur d'un attribut en cas de conflit de structure classe/attribut [Kim et al. 1993]. Par exemple, on peut imaginer que la classe « Route » dans une des bases se compose de routes nationales et départementales ainsi que des chemins et des allées (différenciation selon la valeur d'un attribut « catégorie »). Si la seconde base est dotée de trois classes (« Route », « Chemin », « Allée »), une restructuration peut être envisagée dans la première BD pour faciliter le travail d'évaluation de la cohérence. On peut ainsi créer trois classes (« Route », « Chemin », « Allée ») et calculer les correspondances entre les éléments de ces classes respectives (figure 44).

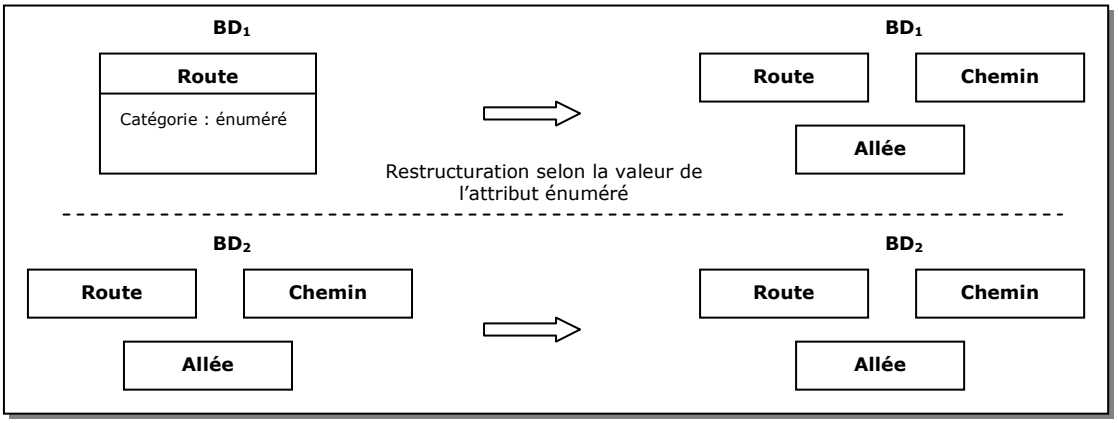


***Figure 44**. Rapprochement des schémas des bases à intégrer par restructuration.*

En plus de la restructuration, un travail d'enrichissement des schémas s'impose également. Cet enrichissement est destiné à préparer les bases à recevoir de nouveaux objets (dans des nouvelles classes) ou de nouveaux attributs qui seront utiles aux étapes de contrôles des représentations. Des attributs portant sur la géométrie des objets peuvent ainsi être ajoutés car l'évaluation de la conformité des représentation nécessite souvent de *caractériser* la géométrie (cf. section suivante). De nouvelles classes peuvent aussi être définies car des concepts existant implicitement dans les données à travers la géométrie peuvent être explicités pour permettre les contrôles. La création de nouvelles classes permet aussi de réduire les *conflits de stockage* susceptibles d'exister entre les données des bases [Devogele 1997]. Un conflit de stockage apparaît lorsqu'une information stockée dans une des bases correspond à une information qui doit être déduite dans l'autre base (figure 45). C'est un conflit très fréquent entre bases de données géographiques. Il est lié à la présence d'informations implicites dans les données. L'enrichissement réalisé à ce niveau est guidé par les spécifications des BDG.

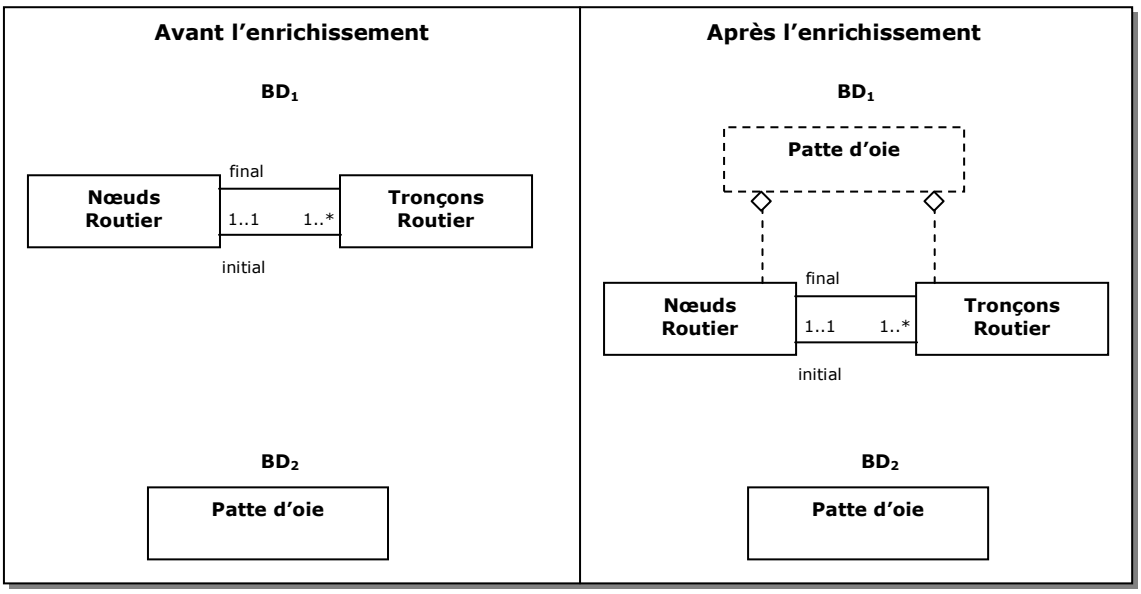


***Figure 45.** Enrichissement du schéma de la première BD se traduisant par la création d'une nouvelle classe « Patte d'oie » pour éliminer un conflit de stockage et rendre possible l'évaluation de la cohérence inter-représentations.*

L'enrichissement des schémas est donc réalisé pour accueillir les nouveaux éléments qui vont servir aux contrôles. Il rend généralement les bases plus homogènes et simplifie les assertions de correspondances inter-schémas. Cet enrichissement s'ajoute à l'enrichissement sémantique préconisé dans le processus d'intégration de [Parent et Spaccapietra 2001] (cf. A.3.2.1.).

C.2.2 ENRICHISSEMENT DES DONNEES

EXTRACTION D'OBJETS IMPLICITES

Nous venons de rappeler l'existence de conflits de stockage susceptibles d'apparaître entre les données. Nous avons également discuté dans le chapitre précédent du décalage existant entre ce qui est décrit dans les spécifications et les données elle-mêmes. Nous avons vu que les spécifications faisaient souvent référence à des objets qui n'étaient pas directement stockés dans la base. Les carrefours en pattes d'oie par exemple de la BDTopo n'existent pas sous forme d'objet dans les données bien que des règles de saisie les concernant soient évoquées dans les spécifications. Si nous voulons vérifier ces spécifications, il est nécessaire d'extraire ces objets. Cette extraction est l'objectif de cette tâche.

L'extraction peut concerner une entité ou un groupe d'entités, ce qui a pour conséquence de créer un objet dans la base dont la géométrie peut être simple (point, ligne, surface) ou composée (agrégat de points, lignes ou surfaces) voire complexe (agrégat de différentes primitives). Les pattes sont un exemple de géométrie simple surfacique. Un objet de géométrie composée pourrait correspondre à un groupe d'arbres. Par exemple, on pourrait imaginer une BDG dans laquelle figurent des arbres, de géométrie ponctuelle. Les spécifications pourraient indiquer que la présence de ces arbres est liée à leur nombre et leur existence dans un alignement : un arbre un saisi s'il fait partie d'un alignement composé d'au moins 5 arbres. Pour vérifier cette règle de saisie, un objet « groupe d'arbres » doit être créé. Sa création permet de contrôler que le nombre d'objets dans le groupe vérifie bien le nombre minimum fixé dans les spécifications. Elle permet également d'étudier la forme de ce groupe pour savoir si les arbres sont alignés ou non (figure 46).

La construction automatique de ces objets n'est pas évidente malgré leur contour bien délimité [Regnauld 1998, Christophe et Ruas 2002, Grosso 2004]. Il est fréquent de devoir mener un travail d'analyse pour comprendre comment extraire l'information implicite. Ainsi, les spécifications de la BDCarto relatives aux tronçons hydrographiques mentionnent que « *tous les axes principaux sont saisis, […], à l'exception des culs-de-sac d'une longueur inférieure à 1km sauf s'ils appartiennent à un cours d'eau d'une longueur supérieur à un kilomètre. Outre l'axe principal, les axes des bras secondaires d'une longueur supérieure à un kilomètre […] sont également saisis* » [BDCarto 2001]. Pour vérifier ces spécifications, il faut distinguer les tronçons principaux et les tronçons secondaires dans la base. Les culs-de-sac doivent également être identifiés. Mais doit-on considérer un cul-de-sac comme un tronçon secondaire (figure 47) ? Et comment distinguer les tronçons principaux des secondaires ? Une étude assez complexe du réseau hydrographique doit être effectuée pour répondre à ces questions. Cette étude doit notamment passer par la création d'un graphe topologique à partir du réseau hydrographique existant. Chaque arête de ce graphe doit ensuite être qualifiée. La tâche d'enrichissement n'est donc pas triviale.

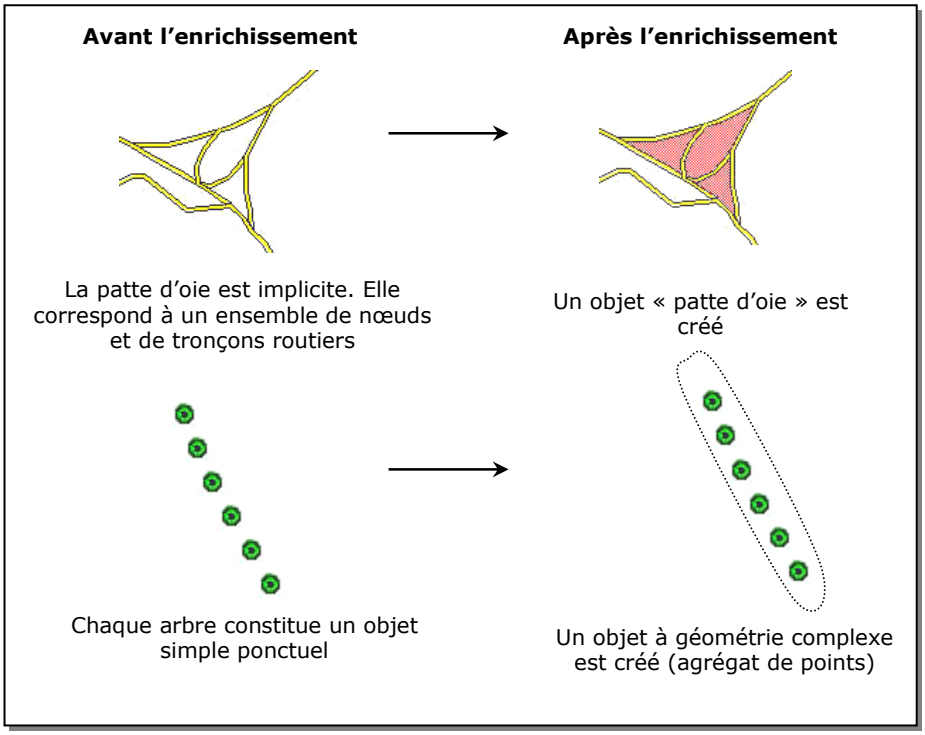


***Figure 46**. Exemples d'enrichissement de données*

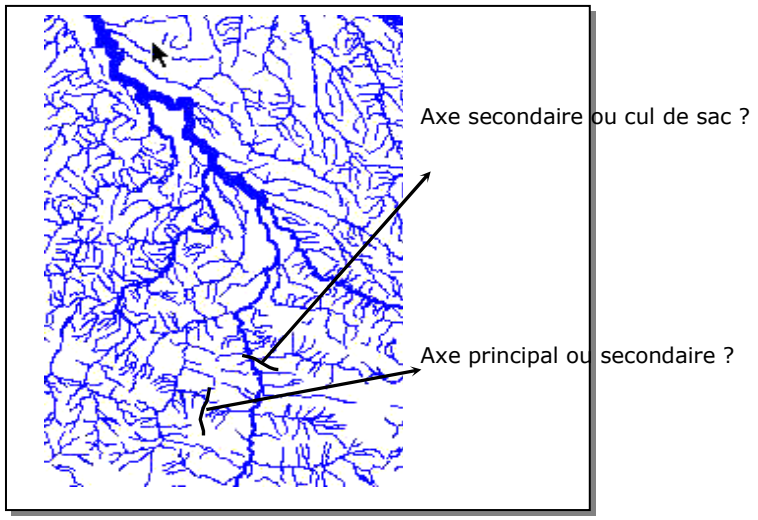


***Figure 47**. L'enrichissement des données pose fréquemment de nombreuses questions et peut nécessiter une analyse complexe des représentations.*

CARACTERISATION DES OBJETS

L'extraction d'objets implicites n'est pas la seule opération à mener lors de l'enrichissement. L'analyse de la conformité des représentations nécessite aussi de *caractériser* les objets.

La caractérisation est une tâche récurrente pour les objets contenu dans les bases de données géographiques. Elle s'impose chaque fois qu'il est nécessaire d'effectuer une analyse en exploitant l'information géométrique des objets. C'est une étape essentielle pour la généralisation cartographique automatique par exemple, et son évaluation [Ruas 1999, Bard 2004]. Elle s'impose également dans notre méthode.

Caractériser un objet signifie qualifier ses propriétés spatiales ou préciser sa sémantique. On fait référence ici aux propriétés spatiales relatives à l'objet lui-même (sa taille, sa forme, son orientation et sa position absolue) ou à celles qu'il entretient avec d'autres objets (relations métriques et topologiques). Mais cette caractérisation ne se fait pas au hasard. Elle est principalement guidée par les spécifications des bases à vérifier. Si on reprend l'exemple des pattes d'oie, seule la base de l'objet sera mesurée, sa représentation étant conditionnée par la longueur de cette base selon les spécifications. De manière similaire, on affectera seulement deux attributs à l'objet « groupe d'arbres », l'un portant sur le nombre d'éléments dans le groupe et l'autre portant sur le caractère aligné de ces éléments (figure 48).

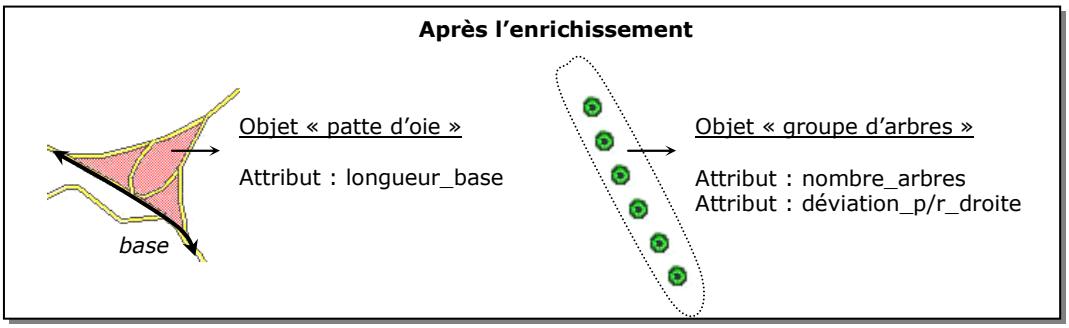


***Figure 48**. Objets enrichis d'attributs caractérisant leur géométrie.*

Les propriétés à mesurer sont souvent faciles à identifier lorsque les spécifications font référence à des critères géométriques. Cela ne veut pas dire que l'opération est toujours simple, l'identification automatique de la base d'une patte d'oie n'est pas immédiate par exemple, mais on sait ce qu'on doit mesurer. Naturellement, il existe des cas où les propriétés à mesurer n'apparaissent pas clairement. Il est par exemple difficile de formaliser une notion comme « *être remarquable dans le paysage* ». On peut supposer que l'environnement de l'objet ait une importance (l'objet doit sans doute être isolé), mais la taille de l'objet peut également intervenir. Pour ce type de cas, une bonne expertise du domaine est indispensable.

La difficulté se pose également lorsqu'on doit qualifier la forme d'un objet (figure 49). Il n'est jamais facile de définir les critères de qualification globale d'une forme. Cet exercice consiste en effet à fournir une description symbolique de la géométrie à l'aide d'une série de mesures censées refléter au mieux la représentation graphique de l'objet. Il n'est pas évident de traduire toutes les propriétés identifiables visuellement et il peut exister plusieurs manières de les exprimer [Barillot et Plazanet 2002].

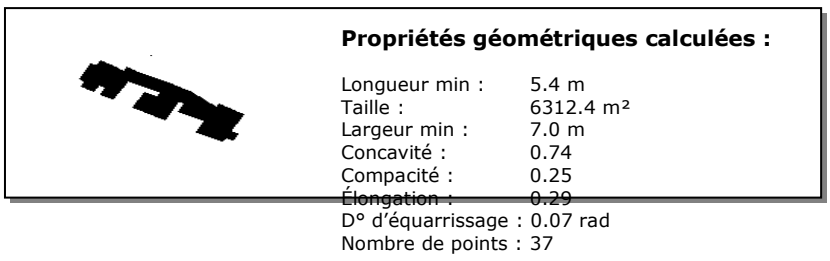


***Figure 49**. Une caractérisation possible de la forme d'un bâtiment (Source : Mustière 2001)*

Le problème d'identification des mesures pertinentes apparaît aussi lorsque les spécifications sont inexistantes. Dans un tel contexte, le choix des mesures n'est plus

guidé par les spécifications mais dépend des représentations homologues et des connaissances du domaine. La caractérisation se fait dans ce cas lors la mise en œuvre du contrôle inter-bases (cf. C.5.).

La caractérisation peut également se traduire par l'ajout d'une nouvelle géométrie à l'objet, dérivée de la première. Cet ajout est fréquemment requis pour des bases de résolutions différentes. Il pourrait se présenter dans le cas des arbres par exemple. Si un alignement d'arbres était représenté par un objet linéaire dans la seconde base, il serait utile d'enrichir la représentation de l'objet « groupe d'arbres » de la première base d'une géométrie linéaire représentative du groupe. Cette géométrie linéaire permettrait de comparer plus facilement les représentations lors de leur mise en correspondance. Elle faciliterait également l'appariement et le rendrait plus fiable. On pourrait en effet s'appuyer sur davantage de critères géométriques que la seule position des objets pour calculer les correspondances, lesquelles seraient nettement simplifiées (figure 50).

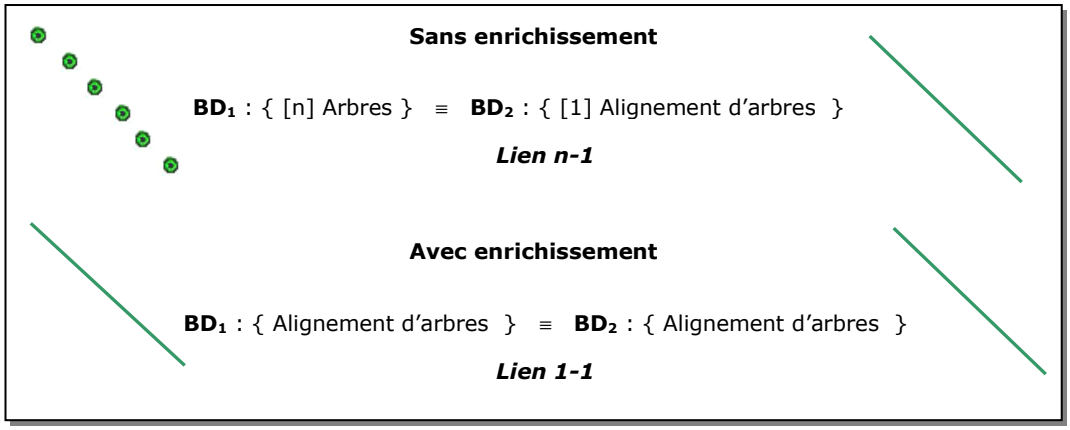


***Figure 50**. L'enrichissement peut rendre le contenu des bases plus homogène et faciliter ainsi le processus d'appariement*

La caractérisation des objets se traduisant par l'ajout d'une nouvelle géométrie peut donc parfois s'imposer pour rendre les représentations comparables et faciliter le calcul des correspondances.

C.2.3 OUTILS D'ENRICHISSEMENT DES DONNEES : L'ANALYSE SPATIALE

Pour enrichir les données, que ce soit pour l'extraction, la caractérisation ou le changement de niveau de détail, nous avons recours à l'analyse spatiale.

L'analyse spatiale est un domaine à part entière en géographie. Elle s'attache à étudier et à formaliser la configuration et les propriétés de l'espace géographique, tel qu'il est produit et vécu par les sociétés humaines [Pumain et Saint-Julien 1997].

Ainsi, l'objet de l'analyse spatiale est d'identifier les régularités qui peuvent apparaître dans l'espace et d'expliquer la localisation des phénomènes présents, en se fondant sur les lois ou règles de la spatialité. La découverte de ces formes d'organisation spatiale peut aboutir à la production de modèles et de théories qui représentent le fonctionnement des systèmes spatiaux. A titre d'exemple, on peut citer les modèles spatiaux de hiérarchies des lieux centraux de W. Christaller (1933) proposés pour rendre compte de l'organisation hiérarchique des villes selon le niveau des biens et des services qu'elles offrent (voir l'ouvrage de [Mérenne-Schoumaker 1996] à ce sujet).

L'analyse spatiale s'appuie sur différents outils pour mettre en évidence les formes d'organisation des objets. Ceux-ci sont empruntés au domaine de la statistique (spatiale ou non), des mathématiques et du traitement d'images.

Dans le cadre de ce travail, nous ne cherchons pas à étudier la forme d'organisation spatiale des objets et en expliquer la cause, mais nous souhaitons exploiter les outils qui sont traditionnellement utilisés.

MESURES ET STRUCTURES D'ANALYSE EN ANALYSE SPATIALE

L'intérêt que nous portons aux méthodes d'analyse spatiale concerne plus spécifiquement ce qu'on a coutume d'appeler les *mesures*. Cette notion de mesure porte ici le sens d'une description d'un concept sous-jacent comme la mesure d'une distance ou d'une forme [Barillot 2002].

L'utilisation de mesures répond à plusieurs objectifs dans notre contexte :

- Elles doivent permettre d'enrichir les données d'attributs relatifs à leur géométrie pour qualifier la représentation de chaque objet à comparer et vérifier le respect des spécifications (*caractérisation*) ;
- Elles doivent permettre d'enrichir les bases de données de structures et d'objets qui ne sont pas explicitement stockés pour rendre les représentations à apparier plus homogènes, plus facilement comparables et rendre possible l'évaluation (*extraction d'objets géométriques implicites* et *changement de niveaux de détails*).
- Elles doivent enfin permettre de créer les couples d'objets appariés (étape d'appariement).

Plusieurs types de mesures sont exploités : des mesures relatives à un objet (superficie, longueur, indicateurs de formes,…), à un groupe d'objets (densité, alignement, indices de dispersion,…), entre deux objets (distance euclidienne, distance de Hausdorff, mesure de parallélisme, orientation relative, mesure d'adjacence…), ou deux groupes d'objets (différence de cardinalité, de densité, d'organisation spatiale,…). Certaines mesures auxquelles nous faisons référence peuvent être trouvées dans [Agent 1999a, Agent 1999b].

Les mesures sont effectuées à plusieurs niveaux d'analyse, en fonction de l'existence ou non d'un groupe d'objets. On distingue classiquement trois niveaux : microscopique, mésoscopique et macroscopique [Ruas 1999]. Le premier niveau correspond à l'objet lui-même (ex : une maison, une route, un champ). Le second niveau fait référence à une collection d'objets proches ayant un sens géographique (ex : un alignement de maisons, un quartier). Le niveau macro représente la population de tous les objets (ex : tous les bâtiments). La construction de groupes d'objets passe fréquemment par la création de nouvelles *structures d'analyse* [Barillot 2002]. Les structures d'analyse les plus courantes sont les graphes spatiaux : triangulation de Delaunay et arbre de recouvrement minimum notamment. Une autre structure très classique est le diagramme de Voronoï (dual de la triangulation de Delaunay). Il forme une partition de l'espace telle que chaque point qui se trouve à l'intérieur d'une cellule de Voronoï est plus proche de son centre que de n'importe quel autre. Nous illustrons ces graphes spatiaux à la figure 51.

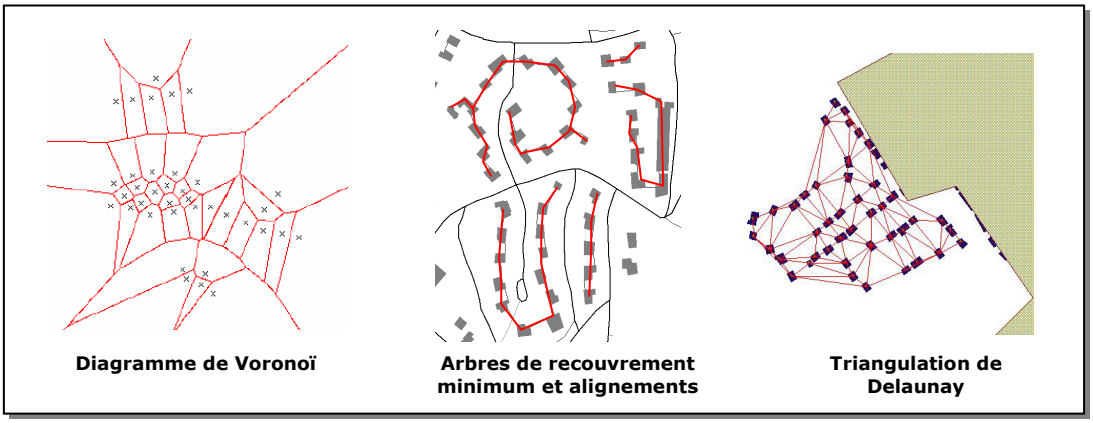


***Figure 51**. La construction d'objets méso se fait à partir de structures d'analyses. Les plus courantes sont les graphes spatiaux : diagramme de Voronoï, arbres de recouvrement minimum et triangulation notamment.*

Une des difficultés de l'utilisation de mesures pour l'extraction d'objets géométriques implicites ou la caractérisation de leurs propriétés réside dans la détermination des seuils. Dans le cas des pattes d'oie par exemple, on pourrait avoir recours à un indice de forme permettant de ne sélectionner que les faces triangulaires. Mais comment fixer le seuil de sélection pour cet indice ? A partir de quelle valeur peut-on considérer que l'objet n'est plus suffisamment triangulaire ? Souvent, ce seuil est déterminé de manière empirique. C'est ainsi que nous avons procédé pour la construction des ronds-points lors de nos expérimentations (chapitre E). Nous pensons que ces seuils pourraient être déterminés automatiquement, à l'aide d'outils d'apprentissage supervisé (cf. D.4.2.2). Quelques travaux ont déjà été menés dans ce sens [Weibel et al. 1995, Plazanet et al. 1998, Sester 2000, Mustière 2001].

C.2.4 BILAN DE L'ENRICHISSEMENT

L'enrichissement constitue la première étape de la méthode *MECO*. Il prépare la vérification de la conformité des représentations et doit être mis en œuvre chaque fois que les informations enregistrées dans les données ne suffisent pas à contrôler les bases directement. Il s'attache ainsi à matérialiser des informations implicites et à caractériser la géométrie des objets pour permettre de réaliser les contrôles intra-base et inter-bases (étapes qui seront présentées dans les sections suivantes).

Pour réaliser l'enrichissement, deux questions essentielles se posent :

- Quelles sont les connaissances à exploiter pour mener l'enrichissement ?
- Comment les acquérir ?

Les connaissances principales à exploiter sont les spécifications. Ce sont elles qui vont permettre de déduire les propriétés à mesurer et les objets à extraire. Mais l'enrichissement ne pourrait pas être mené sans l'intervention de l'expert. C'est lui qui effectue cette déduction, détermine la spécification des outils d'enrichissement, les conçoit ou les choisit. Les connaissances de l'expert interviennent donc également. Enfin, il n'est pas toujours possible de déduire toutes les mesures à effectuer et les objets à extraire sans une analyse des données. Certaines exceptions ou propriétés récurrentes peuvent apparaître dans les données sans qu'aucune information à leur sujet ne soit fournie dans les spécifications. Les données constituent aussi une source de connaissances pour déduire ce qu'il faut enrichir.

L'acquisition des connaissances sera discutée dans le chapitre suivant (méthode *MACO*). Elle consiste pour cette étape en une étude détaillée des spécifications et dans une moindre mesure, des données. L'information pour l'enrichissement est recueillie pour chaque base indépendante mais l'analyse se fait en croisant les sources de connaissances. Les données sont ainsi enrichies pour préparer l'étape du contrôle intra-bases (phase suivante) mais également l'étape du contrôle inter-base (après l'appariement). En d'autres termes, les données sont enrichies d'informations utiles au contrôle individuel des représentations mais aussi à leur comparaison.

La phase d'acquisition des connaissances est essentiellement interactive pour cette étape. Nous verrons que la structuration des spécifications selon un modèle que nous avons défini peut faciliter la tâche (D.3.2.). Par contre, l'enrichissement proprement dit est entièrement automatisable. Il suppose néanmoins l'existence d'une boîte à outils de mesures (sous forme d'algorithmes) ou son développement.

L'étape d'enrichissement est synthétisée en figure 52.

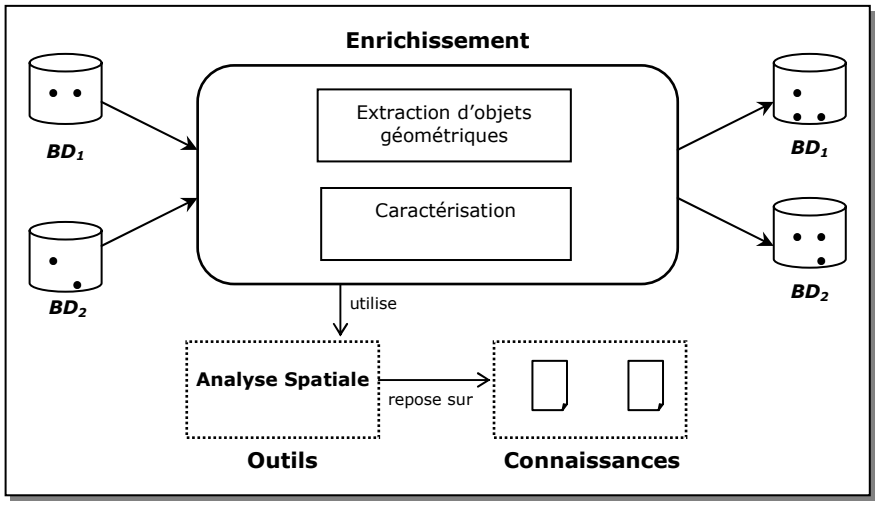


***Figure 52**. L'étape d'enrichissement est constituée de plusieurs tâches lesquelles sont réalisées grâce à des outils d'analyse spatiale qui repose sur des connaissances déduites des spécifications.*

C.3 CONTROLE INTRA-BASE

C.3.1 OBJECTIF DU CONTROLE INTRA-BASE

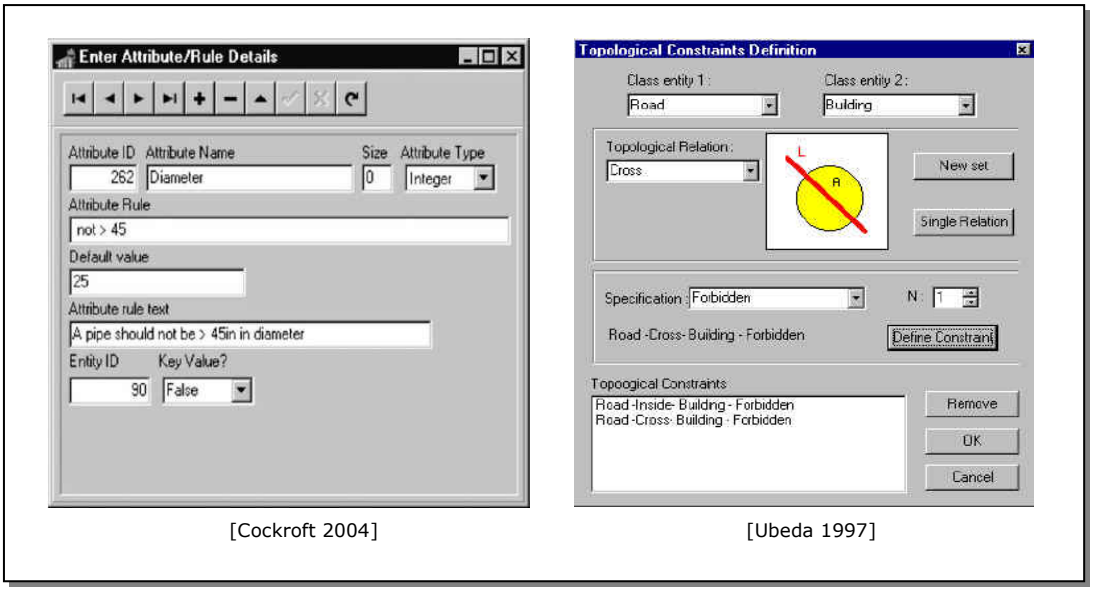
La seconde étape de *MECO*, le contrôle intra-base, a pour objectif de vérifier la conformité des représentations de chacune des bases indépendamment, avant leur mise en correspondance. Cette vérification permet de détecter un certain nombre d'erreurs sans tenir compte des représentations homologues. Elle fournit ainsi une première estimation sur la manière dont les données respectent globalement leurs spécifications.

C.3.2 CONDITIONS D'APPLICATION

Le contrôle intra-base ne porte pas sur l'ensemble des objets des bases. Il concerne uniquement les objets dont les spécifications peuvent être vérifiées sans

l'utilisation d'un autre jeu de données. Il est ainsi possible de vérifier que la longueur de la base d'une patte d'oie est suffisamment grande pour justifier sa représentation détaillée dans la base (une patte d'oie pourrait être représentée par un carrefour simple ponctuel lorsque sa base est inférieure à une longueur fixée par les spécifications). Par contre, la position de cette patte d'oie ne peut pas être contrôlée. De même, il n'est pas possible de savoir s'il y a eu une confusion avec un autre objet. En d'autres termes, on ne peut vérifier à ce stade que certaines contraintes de *cohérence logique*[18] portant sur la géométrie et les attributs (domaine de valeurs) mais pas les paramètres de qualité relatifs à l'exactitude de position, l'exactitude sémantique et l'exhaustivité.

Les contraintes que l'on vérifie à ce niveau sont donc à mettre en relation avec la notion de *contrainte d'intégrité*. Une contrainte d'intégrité est une condition sous forme de prédicat qui doit être vérifiée dans une base de données [Laurini et Millert-Raffort 1993]. Elle est le moyen de vérifier la cohérence des données. Dans le cadre des bases de données spatiales, on peut définir des *contraintes d'intégrité spatiales*. Celles-ci touchent la géométrie et la topologie [Laurini et Millert-Raffort 1993, Cockroft 1997, Borges et al. 2002]. On peut par exemple imposer que le réseau de routes soit connexe, ou que la superficie minimale d'un bâtiment soit supérieure à 50m² ou encore, qu'une route n'intersecte aucun autre objet dans la base. Ces règles correspondent à des contraintes de cohérence logique (du point de vue des standards de qualité) mais peuvent se traduire par des contraintes d'intégrité. Le langage OCL d'UML peut être envisagé pour exprimer certaines contraintes [Friis-Christensen 2003]. Quelques auteurs ont également proposé des interfaces graphiques pour les saisir [Ubeda 1997, Cockroft 2004] (figure 53).



[Cockroft 2004] [Ubeda 1997]

Figure 53. *Interfaces destinées à la saisie de contraintes d'intégrité spatiales. (Source : [Cockroft 2004] et [Ubeda 1997]).*

Néanmoins, les SGBD actuels ne permettent pas d'exprimer des contraintes d'intégrité spatiales [Borges et al. 2002], alors que l'écriture de certaines contraintes ne peut pas se résumer à des formulations déclaratives comme le propose SQL (au

[18] La cohérence logique (composante de la qualité) désigne le « *degré de cohérence interne des données selon les règles de modélisation et les règles inhérentes à la spécification de produit du jeu de données* » [David et Fasquel 1997].

moyen de la clause ASSERT). De ce fait, aujourd'hui, les contrôles de cohérence sont rarement mis en œuvre à partir de contraintes d'intégrité. Ils sont généralement entrepris à l'issue de la saisie des bases, en exploitant les outils proposés par les SIG. Bien souvent, ces contrôles se limitent à la vérification des contraintes topologiques [Ubeda 1997]. Les tests de cohérence menés sont donc assez sommaires. C'est ce qui motive la réalisation du contrôle intra-base.

Il est important de noter que nous nous attachons ici à vérifier des contraintes sur les objets de la base sans être assuré qu'il existe une correspondance exacte avec le monde réel. En effet, les règles de saisie formulées dans les spécifications concernent les objets du monde réel mais rarement ceux de la base. De ce fait, il est parfois nécessaire de reformuler les règles pour les rendre applicables aux objets des bases. Par ailleurs, si les données semblent respecter leurs spécifications, on peut seulement faire l'hypothèse qu'elles reflètent correctement le monde réel mais on ne peut pas le certifier (cf. B.2.3.2.).

Illustrons ces propos en reprenant l'exemple des pattes d'oie. Il se peut que les spécifications indiquent qu'un objet patte d'oie est saisi dans la BD, s'il existe un terre-plein central sur le terrain et que la base de ce terre-plein a une longueur supérieure à 20m. Les spécifications peuvent par ailleurs imposer que ce soit les axes des tronçons constitutifs de la patte d'oie qui soient représentés dans la BD. Dans un tel cas, il existera un décalage entre le phénomène du monde réel et sa représentation dans la BD (figure 54). Une patte d'oie dont la base mesure 26m dans la BD peut correspondre à une entité sur le terrain de base égale à 21m par exemple, puisqu'il est nécessaire de soustraire la largeur d'une bande de circulation de part et d'autre du terre-plein. De plus, la présence d'un terre-plein central ne peut pas être vérifiée. On peut seulement faire l'hypothèse qu'il existe. Après la mise en correspondance des données, on pourra peut-être confirmer cette hypothèse en exploitant la représentation des objets homologues de l'autre base.

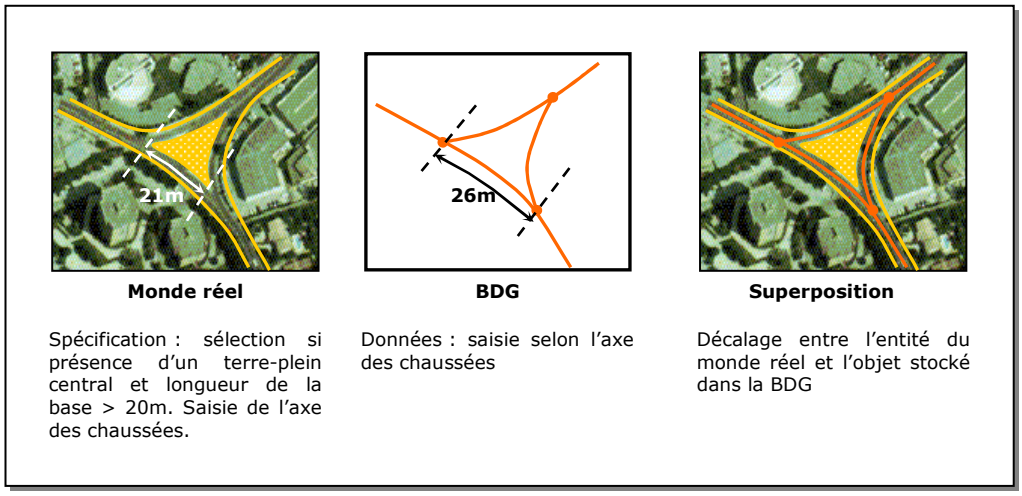


***Figure 54**. Le contrôle de la cohérence est mené sur les objets des bases en exploitant des connaissances qui portent sur les objets du monde réel.*

Dans ce cas-ci, le décalage existant entre les objets du monde réel et ceux de la base peut être pris en compte car les spécifications sont disponibles et il est possible de fixer plus ou moins précisément la longueur à soustraire de la base des pattes d'oie. On peut en effet se référer aux normes du ministère de l'équipement pour déterminer la largeur des chaussées [Setra 1998]. Mais si les spécifications sont

inexistantes, ce décalage ne peut pas être pris en compte. L'avis d'un expert du domaine s'impose dans ce cas.

C.3.3 ERREURS INTRA-BASE

Nous distinguons les *erreurs intra-base* des *erreurs inter-bases*. Les premières correspondent à des représentations qui ne respectent pas leurs spécifications et qui sont identifiées à cette étape de la méthode, sans la mise en correspondance des données. Les secondes sont celles identifiées après l'appariement, lors du contrôle inter-bases (cf. C.5.). Les deux catégories d'erreurs rendent les représentations incohérentes.

Pour illustrer ces deux catégories, reprenons l'exemple des pattes d'oie. On peut imaginer deux représentations de ce phénomène dans les bases à intégrer : une représentation ponctuelle et une représentation détaillée. Dans les deux sources, ces représentations sont conditionnées par la longueur de la base de l'objet dans le monde réel (à partir de l'axe des branches). Toutefois, les seuils fixés n'étant pas les mêmes pour chaque base, on peut s'attendre à des différences de modélisation entre les objets homologues (figure 55).

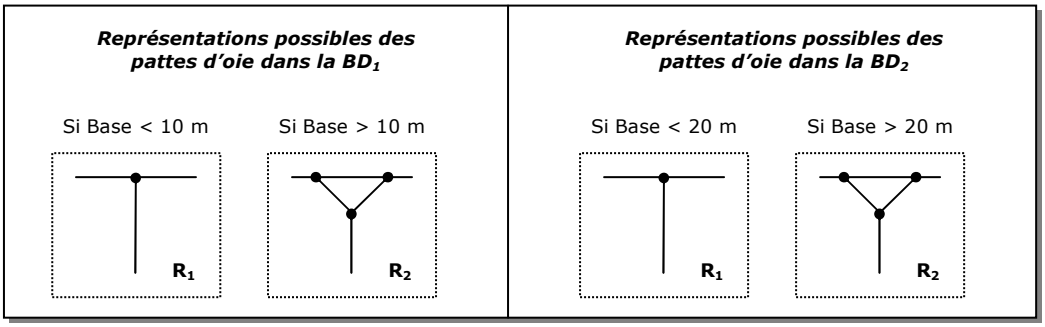


Figure 55. *Exemples de spécifications différentes relatives aux pattes d'oie de deux bases à intégrer.*

Supposons qu'une représentation détaillée soit présente dans la première BD. Dans ce cas, il est possible d'appliquer un contrôle intra-base sur l'objet (figure 56). En fonction de la valeur de la base calculée lors de l'enrichissement, nous pouvons déterminer si la représentation est erronée (erreur intra-base) ou potentiellement conforme.

Dans le cas d'une représentation ponctuelle, aucun contrôle intra-base ne peut être effectué car aucune information ne permet d'évaluer la taille réelle de la patte d'oie. Il est nécessaire d'attendre la mise en correspondance des données pour étudier la conformité de la représentation de la première base en tenant compte de celle de la seconde. C'est l'objet du contrôle inter-bases, étape qui suit l'appariement. En fonction de la modélisation de la patte d'oie dans la seconde base, les représentations seront jugées équivalentes ou incohérentes.

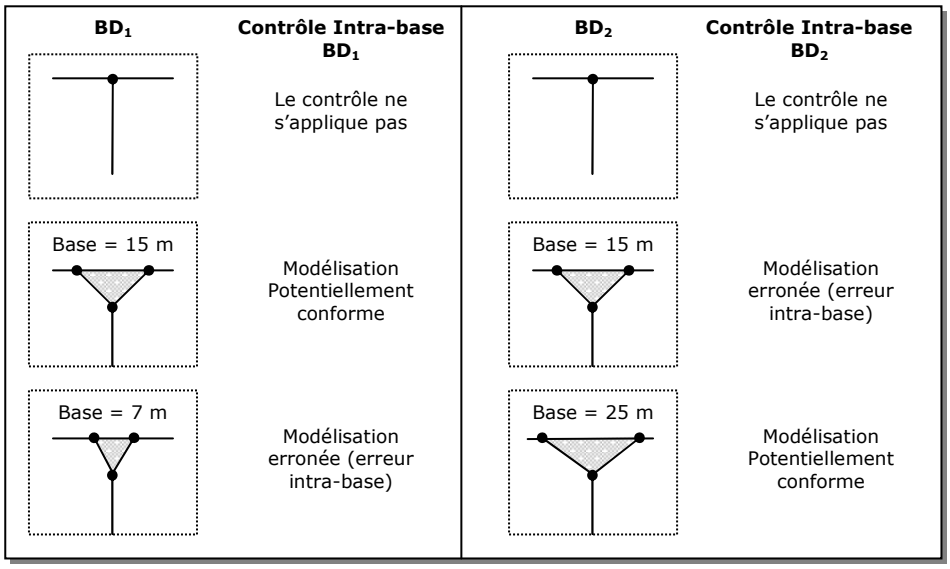


***Figure 56**. Résultats du contrôle intra-base pour différentes représentations de pattes d'oie.*

C.3.4 DEVELOPPEMENT D'UNE BASE DE REGLES

Pour mettre en œuvre le contrôle intra-base, une base de règles doit être créée (étape de *MACO*). Ces règles renferment les connaissances permettant de vérifier la conformité des représentations. L'écriture des règles se fait manuellement, par l'expert, après avoir analysé les spécifications. L'expert se charge donc de changer le langage de représentation des spécifications pour les exprimer sous une forme manipulable par une machine (un langage à base de règles). A l'avenir, on peut penser que ces règles seront dérivées automatiquement à partir d'une base de métadonnées [Cockcroft 2004]. Puisqu'elles sont directement issues des spécifications, elles pourraient être structurées selon le modèle que nous avons défini [Mustière et al. 2003] (cf. chapitre D).

Chaque BDG est donc associée à un ensemble de règles qui lui est propre. Le contrôle consiste à vérifier que les conditions des règles sont respectées par les représentations. Ce contrôle est automatique grâce à l'utilisation du moteur d'un système-expert. Il est rendu possible grâce à l'enrichissement. Les valeurs des propriétés mesurées lors de l'enrichissement sont comparées aux valeurs fixées dans les conditions des règles issues des spécifications.

C.3.5 ÉVALUATION DE LA REPRESENTATION DES OBJETS

Au terme de cette étape, la conformité d'un sous-ensemble des représentations des objets de chaque base est évaluée. Cette évaluation peut porter sur une ou plusieurs propriétés des objets (ex : nombre d'arbres dans le groupe, respect de l'alignement,…). Chacun de ceux-ci est qualifié : la représentation peut être *potentiellement conforme* ou *non conforme* (*erreur intra-base*).

C.3.6 BILAN DU CONTROLE INTRA-BASE

Le contrôle intra-base permet de détecter un certain nombre d'erreurs avant de mettre en correspondance les données. Ces erreurs permettront d'expliquer les

correspondances incohérentes après l'appariement. Seules les connaissances issues des spécifications permettent de contrôler la conformité des représentations à ce niveau. En leur absence, cette étape ne peut pas s'appliquer. Les représentations des objets ne pourront être jugées qu'au contrôle inter-bases.

Puisque cette étape fait appel à un grand nombre règles, nous proposons de manipuler ces règles automatiquement par un système-expert. Nous détaillerons les caractéristiques de ces systèmes à la fin de ce chapitre. L'étape du contrôle intra-base est synthétisée en figure 57.

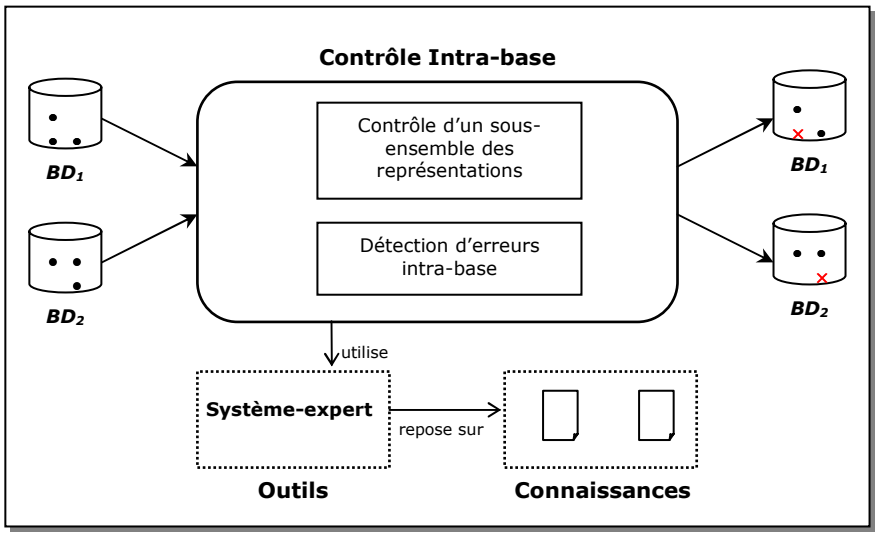


***Figure 57**. Le contrôle intra-base permet de détecter des erreurs intra-base en exploitant des connaissances issues des spécifications, décrites sous forme de règles et gérées par un système-expert.*

C.4 APPARIEMENT

C.4.1 OBJECTIF DE L'APPARIEMENT

L'appariement est une étape centrale dans *MECO*. Il permet d'associer les données des deux bases et de produire des liens explicites entre les objets homologues. L'appariement aboutit à la création d'un ensemble de couples d'objets appariés sur lesquels vont porter l'analyse de différences de représentation.

C.4.2 STRATEGIE D'APPARIEMENT ADOPTEE

Deux stratégies peuvent être envisagées pour apparier les données : soit on apparie seulement les objets qui forment des couples conformes aux spécifications des deux bases, soit on apparie tous les objets, peu importe la conformité du lien, pourvu que ces objets semblent modéliser la même entité du monde réel. Cette modélisation peut être erronée mais on peut supposer qu'il s'agit de la même entité sur le terrain si on prend la position des objets comme référence.

La première stratégie implique que l'évaluation se fasse en même temps que l'appariement. Des correspondances entre les données qui ne sont pas prévues d'après l'analyse croisée des spécifications ne sont pas matérialisées. De cette

manière, chaque correspondance calculée est directement qualifiée d'équivalence, par opposition aux incohérences. Cette stratégie présente l'avantage de ne produire en théorie aucune erreur d'appariement puisque l'appariement est réalisé en exploitant un ensemble de connaissances, en plus de la position et la forme des objets. En pratique cependant, il est rarement possible de prévoir toutes les correspondances au niveau des données. Les spécifications peuvent être imprécises et manquer. En outre, les algorithmes d'appariement qui seraient utilisés dépendraient fortement des données au détriment de leur généricité. Cette stratégie nous semble donc peu adaptée.

La seconde stratégie ne se soucie pas de la conformité des représentations. L'évaluation se fait à l'issue de l'appariement. Cet appariement est fondé essentiellement sur des critères géométriques et topologiques et est assez peu guidé par les spécifications des bases (les règles de saisie ne sont pas exploitées). De ce fait, un certain nombre d'erreurs d'appariement est susceptible d'apparaître car les algorithmes utilisent moins d'heuristiques. Cependant, la boîte à outils d'appariement est plus générique (il peut s'agir d'une « boîte noire »). Par ailleurs il n'est plus nécessaire de prévoir toutes les correspondances possibles avant l'appariement. On peut étudier comment les données se correspondent et *apprendre* au besoin les correspondances équivalentes et incohérentes par la suite (cf. contrôle inter-bases). Nous optons pour cette seconde stratégie.

Il faut toutefois préciser que la limite entre les deux stratégies n'est pas nette. On doit nécessairement utiliser des connaissances sur les bases pour apparier les objets. Cependant, pour la seconde stratégie, on peut réduire ces connaissances à des informations très générales. On peut se contenter par exemple de l'écart moyen quadratique pour fixer les paramètres des outils d'appariement.

C.4.3 CALCUL DES LIENS D'APPARIEMENT

Les techniques d'appariement que nous utilisons dont certaines ont été développées dans cette thèse (cf. chapitre E) se fondent essentiellement sur la ressemblance des formes et la proximité de localisation des objets de chaque base. Nous utilisons différentes mesures (distance, longueur, taille, etc.) pour identifier les objets homologues. Suivant les différences de représentation entre les objets, des liens de cardinalités 0-1, 1-0, 1-1, 1-m, n-1, n-m peuvent être calculés.

En terme de proximité, deux distances sont particulièrement utiles pour apparier les objets linéaires et polygonaux : la *distance de Hausdorff* et la *distance surfacique*. La première a été initialement exploitée par [Abbas 1994] dans le cadre du contrôle qualité des BDG. Elle est employée dans le module d'appariement de [Devogele 1997] pour relier les réseaux routier de la BDCarto et Georoute. Nous y reviendrons dans le chapitre E (la méthode est exposée en annexe). Nous l'avons utilisé pour valider les liens d'appariements calculés lors de nos expérimentations. La distance de Hausdorff, qui fournit l'écart maximal entre deux lignes, a deux composantes. Chaque composante correspond au maximum des plus courtes distances euclidiennes des éléments d'une des lignes par rapport à l'ensemble des éléments de l'autre ligne. La valeur maximale des deux composantes est la distance de Hausdorff recherchée (figure 58).

La distance surfacique a été définie par [Vauglin 1997]. Elle a prouvé son efficacité pour l'appariement d'objets polygonaux [Bel Hadj Ali 2001]. Nous donnons sa définition à la figure 58. Il s'agit d'une distance au sens mathématique du terme (tout comme celle de Hausdorff) dont les valeurs évoluent dans l'intervalle [0,1].

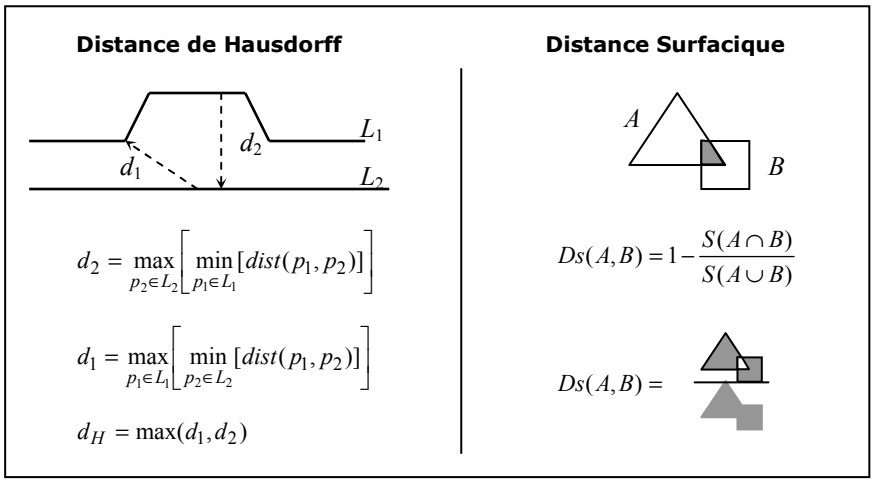


***Figure 58**. Définition de la distance de Hausdorff et de la distance Surfacique. (D'après [Badard et Lemarié 2002])*

Le calcul des liens d'appariement peut se faire de différentes manières. Les mesures évoquées ci-dessus font partie des outils d'appariement développés au laboratoire COGIT de l'IGN [Lemarié 1996, Devogele 1997, Lemarié et Bucaille 1998, Badard 2000, Bel Hadj Ali 2001, Mustière 2002]. Elles sont données à titre d'exemple. Nous ne préconisons aucune méthode spécifique à ce niveau pourvu que celles-ci calculent le mieux possible les correspondances sans se soucier de leur cohérence[19]. La tâche qui incombe à l'expert est de choisir les outils d'appariement (ou de les développer), de décider de leur enchaînement (car l'appariement requiert souvent la mise en œuvre de plusieurs algorithmes), et de les paramétrer.

Le paramétrage est un aspect délicat de l'application d'algorithmes d'appariement. Il demande une certaine expertise du domaine. Il se caractérise par la détermination des seuils de recherche des candidats à l'appariement (ce qui suppose d'utiliser des connaissances sur les bases). Cette recherche constitue la première phase de l'opération. Pour chaque objet de la base de plus faible résolution, on détermine les candidats potentiels de l'autre base. Ceux-ci sont généralement retenus en fonction de leur position. On sélectionne ensuite le meilleur candidat dans cet ensemble par filtrage et on valide finalement le lien établi.

Dans notre contexte, une attention particulière doit être portée sur la détermination de ces seuils. Il est judicieux de choisir des seuils assez larges pour permettre d'apparier des objets homologues anormalement éloignés. On peut considérer que l'écart maximal théorique possible entre les objets des deux bases correspond à la somme des erreurs moyennes quadratiques admises pour les classes considérées (métadonnée de qualité). Nous préconisons de fixer un seuil plus grand que la valeur de cet écart. De cette manière, les cardinalités de type 0-1 ou 1-0 ne concerneront que les différences de sélection des objets et non les incohérences de position, lesquelles seront identifiées notamment pour des liens 1-1. Ceci facilite l'interprétation des couples. Nous détaillerons les algorithmes utilisés et développés lors de la présentation de nos expérimentations.

[19] Bien que l'appariement fasse encore l'objet de nombreuses recherches, il est possible aujourd'hui de trouver des modules d'appariement de données géographiques en accès libre sur internet. C'est le cas du projet JUMP notamment (http://www.jump-project.org/).

C.4.4 RESTRUCTURATION DES LIENS

Suivant l'algorithme d'appariement dont on dispose, il est parfois nécessaire de restructurer les liens calculés. Il est possible en effet que l'algorithme utilisé ne soit pas tout à fait adapté au problème traité, conséquence de l'utilisation d'outils génériques. Dans ce cas, les couples d'objets devront être modifiés pour rendre possible l'évaluation de leur conformité.

Dans l'exemple des pattes d'oie, on peut ainsi imaginer deux appariements différents (figure 59). Le premier appariement fournit une correspondance entre le nœud de la première base (représentation ponctuelle) et l'ensemble du cycle de la patte d'oie (représentation détaillée). C'est une correspondance qui est bien adaptée à l'évaluation des différences (figure 59a). Le second appariement propose une correspondance entre le nœud de la première base et la base de la seconde (figure 59b). Ce couple peut être envisagé dans le cas d'applications de géocodage par exemple mais est moins bien adapté à notre contexte.

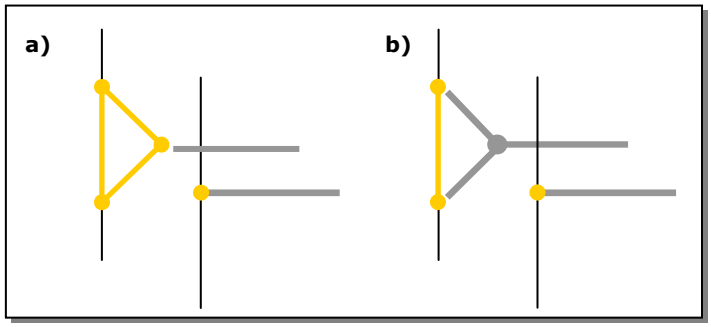


Figure 59. *Deux appariements possibles entre représentations de pattes d'oie différentes. Dans un cas (a), le nœud (patte d'oie non détaillée) est relié avec le cycle (patte d'oie détaillée). Dans l'autre cas (b), seule la base de la patte d'oie est reliée au nœud.*

C.4.5 ÉVALUATION DES LIENS

Jusqu'à présent, nous avons supposé que l'opération d'appariement était entièrement automatique, notre objectif étant de réduire, autant que possible, l'intervention humaine. Cette automatisation offre un gain de temps considérable mais en contrepartie, elle peut réduire le nombre de correspondances évaluées. Nous en discutons ci-dessous.

L'automatisation de l'appariement peut être complexe. Le processus est d'autant plus complexe quand les niveaux d'abstraction entre les bases sont éloignés. Dans certains cas, il est même difficile d'identifier les objets homologues visuellement (figure 60). En raison de cette complexité, il résulte toujours un certain nombre d'erreurs d'appariement.

Ces erreurs doivent être identifiées pour poursuivre la méthode d'évaluation ou tout au moins, nous devons être capable de différencier les couples certains des couples incertains. Les erreurs peuvent avoir une influence sur les résultats de l'évaluation. Une correspondance erronée pourrait en effet être jugée incohérente à tort.

Nous envisageons deux solutions pour évaluer l'exactitude des couples d'objets appariés. La première est une solution interactive. Elle consiste à passer en revue chaque couple d'objets pour approuver sa validité. On vérifie qu'il s'agit bien d'objets homologues lesquels sont censés représenter le même phénomène dans la réalité.

C'est une méthode qui permet de corriger les erreurs d'appariement et de poursuivre la démarche d'évaluation avec l'ensemble des couples. Naturellement, c'est une opération fastidieuse qui fait perdre en partie le bénéfice de l'automatisation du processus.

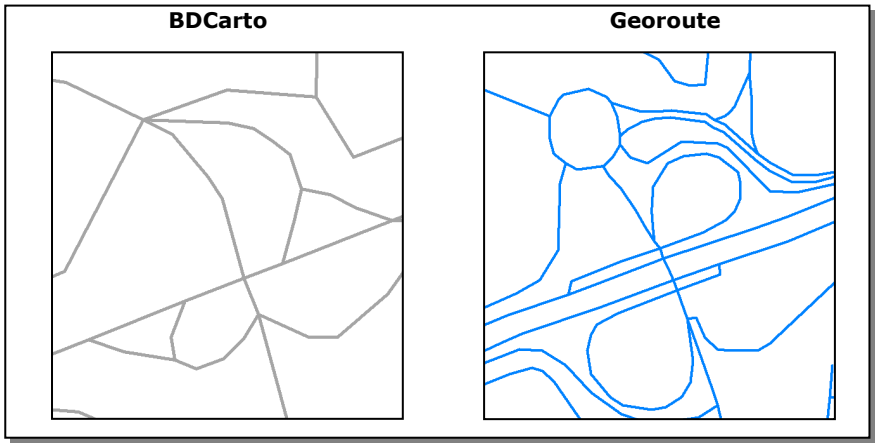


Figure 60. *L'identification des correspondances entre les objets homologues peut parfois s'avérer difficile, même manuellement.*

La seconde solution consiste à mettre en œuvre une deuxième méthode d'appariement automatique. On calcule une deuxième fois les couples d'objets avec une méthode différente de la première. La différence peut concerner la stratégie d'appariement (ascendante, descendante ou mixte) ou les critères géométriques exploités (forme, position, relations topologiques,…). On compare ensuite les résultats des deux appariements et on considère les couples présentant la même réponse comme certains. Les autres sont jugés incertains et nécessitent un traitement particulier. C'est une solution qui permet d'attribuer automatiquement un degré de confiance aux couples. Elle suppose cependant d'avoir deux méthodes différentes d'appariement à disposition, ce qui n'est pas toujours possible. Cette solution a été adoptée dans notre étude menée sur les ronds-points (chapitre E).

Une attention particulière doit donc être portée sur l'ensemble des couples incertains. Ils peuvent être traités de différentes manières. D'abord, on peut décider de les abandonner, c'est-à-dire de ne poursuivre la démarche d'évaluation qu'avec les couples jugés certains. C'est une solution qui est envisageable mais qui réduit naturellement le nombre de couples évalués. Ceci est dommageable car bien souvent, parmi les couples incertains, de nombreux couples sont bien appariés. On peut ensuite décider de les valider interactivement puisque cette fois, la quantité de couples est assez faible. L'apprentissage pourrait également aider à identifier ces couples. On peut imaginer apprendre des règles permettant de différencier les couples certains des incertains. Dans ce cas cependant, davantage de connaissances devront être utilisées ce que nous ne souhaitons pas (suite à la stratégie adoptée). Par contre, on peut très bien accorder un poids plus faible à ces couples lors du contrôle inter-bases. Si l'apprentissage est appliqué à cette étape de la méthode, on peut réduire l'influence de ces couples dans la découverte de règles. Cette solution peut également être adoptée dans le cas d'un système qui s'auto-évalue, en déterminant le seuil de confiance à partir duquel le couple doit être jugé incertain (confiance inférieure à 75% par exemple).

C.4.6 BILAN DE L'APPARIEMENT

La phase d'appariement, tout comme les autres étapes de la méthode, est guidée par des connaissances. Des connaissances sont requises pour :

- Sélectionner les ensembles d'objets sur lesquels doivent s'appliquer le calcul des correspondances ;
- Sélectionner les outils d'appariement adéquats en fonction des correspondances recherchées ;
- Paramétrer les outils d'appariement ;

Les outils d'appariement ne sont pas totalement indépendants des spécifications. Ils le sont dans une certaine mesure. Ainsi, on n'applique pas n'importe quel algorithme pour apparier les objets. On choisit l'algorithme en fonction des représentations que peuvent avoir ces objets. Pour les pattes d'oie par exemple, si on reprend les spécifications des deux bases décrites précédemment (figure 55), on peut s'attendre à trois types de correspondances : des correspondances entre points, entre surfaces et entre un point et une surface. Les algorithmes d'appariement rechercheront ces types de correspondances. Une correspondance impliquant un objet de représentation linéaire ne pourra pas être calculée car elle n'aura pas été attendue. L'incohérence sera mise en évidence lors du contrôle inter-bases (lien 1-0 anormal). Ceci ne veut pas dire qu'on apparie seulement des correspondances équivalentes. La conformité des représentations n'est pas évaluée. Cela signifie plutôt qu'on accepte seulement les correspondances qu'il est possible de définir entre les schémas.

Les correspondances déclarées entre les schémas constituent donc une source de connaissances intéressante pour identifier les outils d'appariement à utiliser, au même titre que les spécifications des bases. Mais l'intervention de l'expert est particulièrement requise à cette étape, comme ce fut le cas pour l'enrichissement. C'est l'expert qui se charge de sélectionner les ensembles d'objets à apparier, de choisir (ou développer) les outils d'appariement et de réaliser leur paramétrage. Le calcul des liens d'appariement proprement dit est entièrement automatisable en appliquant les algorithmes définis. Nous synthétisons cette étape ci-dessous (figure 61).

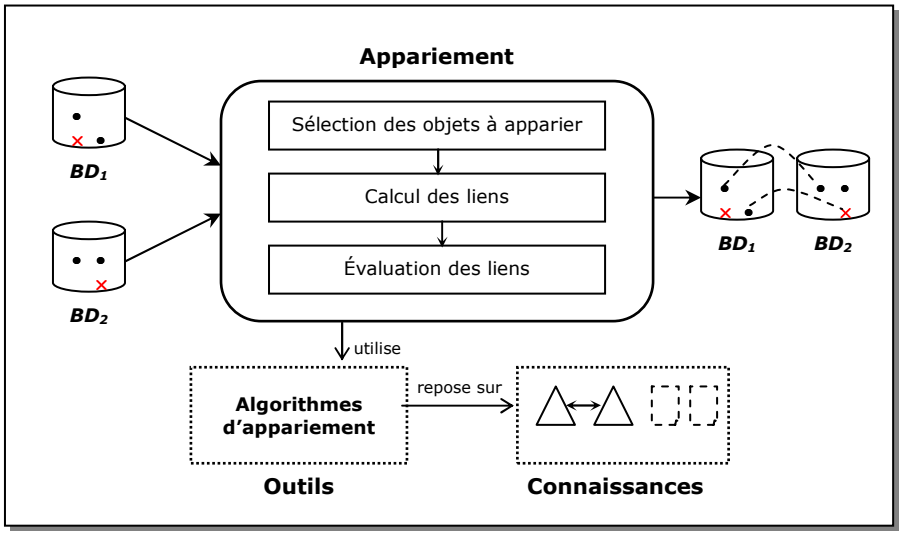


***Figure 61**. L'étape d'appariement est composée de plusieurs tâches lesquelles sont réalisées automatiquement grâce à des outils choisis ou développés par un expert du domaine, guidé par des connaissances.*

C.5 CONTROLE INTER-BASES

C.5.1 OBJECTIF DU CONTROLE INTER-BASES

L'étape qui précède l'appariement, le contrôle intra-base, a permis de mettre en évidence un certain nombre d'erreurs dans les deux sources à intégrer (cf. C.3.3.). Ce contrôle a été mené sur chaque base indépendante, en exploitant seulement les informations s'y rapportant.

A ce stade de la méthode *MECO*, c'est la cohérence inter-représentations qui est évaluée. Il faut déterminer si les différences de représentation entre les objets homologues sont normales ou pas en s'aidant de connaissances s'y rapportant. Il faut étudier la conformité du couple et préciser, quand cela est possible, la base qui est à l'origine d'une incohérence.

C.5.2 COMPARAISON DE LA REPRESENTATION DES OBJETS

Pour évaluer la cohérence inter-représentations, les données doivent être comparées et la comparaison doit porter sur différents aspects : la géométrie, les attributs, les relations entre les objets, leur sélection, leur position. Par exemple, dans le cas des arbres et leur alignement, l'évaluation de la cohérence d'un couple d'objets « alignement d'arbres » passe par la comparaison et le contrôle : de la position, il faut vérifier que l'écart entre les deux objets appariés n'est pas anormal ; de l'exactitude sémantique, il faut vérifier qu'il n'existe aucune confusion sémantique avec un autre type d'alignement par exemple ; de la complétude, il faut justifier la présence des deux objets dans le couple ; de la forme et la taille, il faut vérifier que les différences de forme et de taille des alignements sont acceptables.

En ce qui concerne la position, la conformité de l'écart peut être déterminée en fonction de l'erreur moyenne quadratique admise (EMQ) pour chaque base. Cet écart est mesuré lors de l'appariement géométrique.

Concernant le problème de confusion, ce sont les spécifications qui peuvent aider à déterminer si la sémantique des objets est exacte ou pas mais dans certains cas, il n'est pas possible de préciser dans quelle base réside l'erreur. On peut seulement constater la différence et en conclure qu'il s'agit d'une incohérence. La détermination de la source de l'incohérence imposerait de choisir une base de référence. C'est également le cas de l'évaluation de la modélisation des objets (au sens large). Certaines correspondances peuvent être jugées incohérentes ou équivalentes sans pouvoir spécifier l'origine des erreurs dans le cas des incohérences.

D'autre part, les connaissances issues des spécifications peuvent être insuffisantes pour déterminer la conformité des représentations. Dans le cas des arbres, comment fixer par exemple l'écart maximal acceptable entre les longueurs des alignements homologues sur la base de ces informations ? Les spécifications ne renferment pas toujours l'ensemble des connaissances utile à l'évaluation.

> *C'est ce qui explique l'intérêt que nous portons aux méthodes d'apprentissage automatique. L'apprentissage peut permettre de déterminer cet écart automatiquement, à partir d'un ensemble de couples d'alignements dont la conformité aura été évaluée par un expert. Nous pouvons ainsi, grâce à ces méthodes, savoir dans quel cas un écart de longueur doit être jugé anormal ou pas.*

Nous reviendrons sur ces techniques par la suite (méthode *MACO* – D.4.2.). Il faut noter que de nombreux objets peuvent ne pas avoir d'homologues dans l'autre base mais ceci n'empêche pas de les contrôler. Il faut justifier dans ce cas la non-correspondance (lien 1-0 ou 0-1).

C.5.3 ÉQUIVALENCES, INCOHERENCES, ERREURS INTER-BASES

Le contrôle inter-bases s'applique donc sur chaque couple d'objets appariés et exploite conjointement les deux représentations du couple pour justifier sa conformité et, si possible, celle des représentations respectives. Le contrôle permet toujours de spécifier si les représentations sont *incohérentes* entre elles ou *équivalentes*. Par contre, la base dans laquelle réside l'erreur n'est pas toujours précisée. C'est ce que nous allons illustrer au travers d'exemples exposés ci-dessous.

La figure 62 est composée d'une série de correspondances entre des représentations de pattes d'oie de deux bases de données. Les spécifications relatives à ces objets sont celles qui ont été présentées en section C.3.3 (figure 55). Chaque correspondance a été évaluée et nous allons les analyser pour expliquer le raisonnement qui a été suivi.

La première correspondance concerne des représentations ponctuelles. Aucun contrôle intra-base n'a pu être appliqué. Seule l'évaluation de la cohérence inter-représentations peut être considérée. Les représentations de cette correspondance ont été jugées équivalentes. En effet, l'existence d'une représentation ponctuelle dans la première base, laisse supposer que la longueur de la base de la patte d'oie sur le terrain est inférieure à 10m (d'après les spécifications). En tenant compte de cette supposition, la représentation de l'objet dans la deuxième base devrait être ponctuelle, ce qui est bien le cas. On peut donc admettre que ces représentations sont équivalentes mais on ne peut pas affirmer avec certitude que chaque représentation est conforme à la réalité et ceci est valable pour tous les cas.

La deuxième correspondance est incohérente. Il n'est pas cohérent d'avoir une représentation détaillée dans la deuxième BD avec une représentation ponctuelle dans la première. La question est alors de savoir s'il est possible de déterminer la base qui contient l'erreur. Pour y répondre, on ne peut s'appuyer que sur la représentation de la deuxième BD. Celle-ci est plus informative car un contrôle intra-base a pu être mené. Étant donné que ce contrôle a mis en évidence une erreur intra-base, on pourrait conclure que cette base est responsable de l'incohérence inter-représentations. La conformité de la représentation de la première BD doit ensuite être déterminée, dans la mesure du possible. Elle pourrait elle aussi contenir une erreur. En exploitant la représentation de la deuxième BD, on s'aperçoit que ce n'est pas le cas. Elle semble donc être conforme.

Les deux correspondances qui suivent sont assez similaires en terme de connaissances disponibles et de nombre de contrôles intra-base menés (n°3 et 4). L'une d'entre elles est incohérente et aucune des représentations n'est conforme. Une *erreur inter-base* a pu être mise en évidence. L'autre est équivalente, composée de représentations conformes. Comme précédemment, c'est la représentation détaillée qui aide à déterminer si l'objet ponctuel vérifie ses spécifications ou pas. On peut légitimement penser que si l'objet est détaillé, il représente sans doute bien la réalité même s'il ne respecte pas ses spécifications. Cette hypothèse se justifie si les objets sont saisis à partir de photographies aériennes. Il y a de fortes chances que l'opérateur ait suivi le contour de l'objet et n'ait pas inventé la géométrie. On s'appuie

donc ici sur les représentations détaillées des objets et des connaissances du domaine pour évaluer la cohérence inter-représentations.

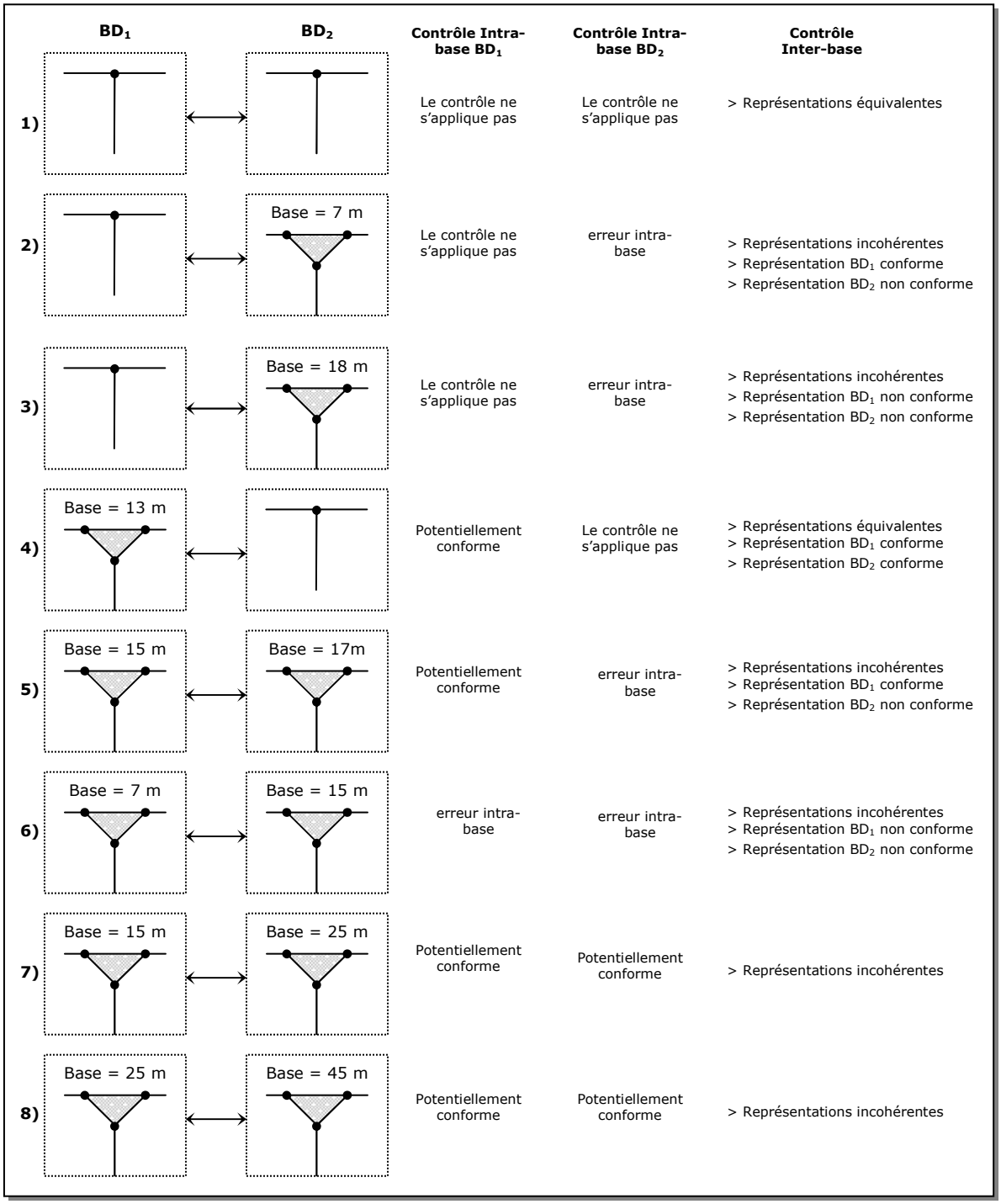


***Figure 62**. Quelques exemples de correspondances analysées entre pattes d'oie.*

La cinquième correspondance est différente des précédentes. Cette fois, le contrôle intra-base a pu être appliqué dans les deux bases. Étant donné que les valeurs mesurées sont proches et que la seconde base contient une erreur intra-base, on peut considérer que c'est elle qui est responsable de l'incohérence.

La sixième correspondance est également incohérente et les connaissances dont on dispose sont les mêmes que pour le cas précédent. Néanmoins, la différence entre les longueurs des bases des pattes d'oie est anormale. L'incohérence est liée à la présence d'erreurs intra-base dans les deux sources mais aussi à la différence anormale des longueurs des bases.

Les deux dernières correspondances illustrent des incohérences pour lesquelles il n'est pas possible de préciser dans quelle base réside l'erreur. Le contrôle intra-base a pu être réalisé sur chaque représentation et les représentations ont été jugées conformes. Ce qui a conduit à considérer les correspondances comme incohérentes c'est la différence anormale entre les longueurs mesurées. Si on sait qu'une des bases est meilleure que l'autre alors on peut supposer laquelle a tort.

Que peut-on tirer de ces exemples ? On peut constater que l'existence d'une représentation sur laquelle peut s'appliquer un contrôle intra-base joue un rôle important dans la justification de la conformité de la correspondance. Tantôt cette représentation est utilisée pour décider si la représentation homologue est conforme ou pas, tantôt cette représentation est exploitée pour confirmer ou infirmer sa conformité.

Nous allons maintenant présenter la manière d'organiser les connaissances pour mener le contrôle inter-bases. Nous proposons deux solutions différentes pour décrire les connaissances dans une base de règles destinées à évaluer la cohérence inter-représentations : la *classification directe* et la *prédiction, comparaison et classification*. Nous les exposons ci-dessous.

C.5.4 ORGANISATION DES CONNAISSANCES POUR LE CONTROLE INTER-BASES

La manière d'organiser les connaissances dans la bases de règles pour mener le contrôle inter-bases est indépendante du mode d'acquisition de ces règles. Elles peuvent être définies aussi bien par l'expert manuellement, après une étape d'analyse des spécifications, ou automatiquement, en mettant en œuvre l'apprentissage automatique. Nous y reviendrons dans le chapitre suivant, en exposant *MACO*. Les deux approches que nous proposons pour représenter les règles sont décrites ci-dessous.

C.5.4.1 CLASSIFICATION DIRECTE

Le principe de cette première approche consiste à développer une base de règles qui permettent de classer directement chaque couple d'objets appariés en terme d'incohérence ou d'équivalence en fonction de la représentation des objets constituant le lien. Autrement dit, si (O_{1i},O_{2j}) représente un couple d'objets appariés, la *classification directe* des différences est réalisée en appliquant un ensemble de règles décrites sous la forme :

SI condition$_A$ (O_{1i},O_{2j}) ALORS (O_{1i},O_{2j}) est équivalent
SI condition$_B$ (O_{1i},O_{2j}) ALORS (O_{1i},O_{2j}) est incohérent

Si on reprend l'exemple des pattes d'oie, un ensemble de règles de classification des différences peut être décrit en adoptant cette approche parmi lesquelles :

SI O_1 = Patte d'oie ponctuelle et O_2 = Patte d'oie ponctuelle
ALORS (O_1,O_2) est équivalent

SI O_1 = Patte d'oie détaillée avec longueur_base < 20m et O_2 = Patte d'oie ponctuelle
ALORS (O_1,O_2) est équivalent

Il faut noter qu'en pratique, il est préférable d'exprimer toutes les règles relatives aux équivalences et de considérer que les correspondances qui ne respectent pas ces règles sont incohérentes. Les incohérences ne sont en effet pas toutes prévisibles et le nombre de combinaisons possibles entre les représentations différentes peut rapidement devenir démesuré.

Pour déterminer dans quelle base réside l'erreur, le résultat du contrôle intra-base doit être exploité. Si ce contrôle a mis en évidence une erreur intra-base (ce qui suppose que le couple est incohérent), alors la représentation s'y rapportant sera considérée comme non conforme. La qualification des représentations de chaque correspondance doit être réalisée après avoir identifié les incohérences.

C.5.4.2 PREDICTION, COMPARAISON, CLASSIFICATION

La seconde approche que nous proposons pour organiser les connaissances distingue trois ensembles de règles différents. La classification des différences n'est pas réalisée directement. Chaque représentation du couple est d'abord utilisée pour prédire la forme de la représentation de l'objet homologue dans l'autre base. Ensuite, les représentations prédites et stockées sont comparées. Enfin, si ces représentations sont identiques dans les deux sens, les représentations sont considérées comme équivalentes. Si elles sont différentes, les représentations sont considérées comme incohérentes. En termes de règles, cette approche peut être exprimée sous la forme suivante :

Prédiction :

SI $condition_A$ (O_{1i}) ALORS (O_{2j}) doit respecter $condition_B$
SI $condition_c$ (O_{2j}) ALORS (O_{1i}) doit respecter $condition_D$

Comparaison et classification :

SI $condition_B$(O_{2j}) et SI $condition_D$(O_{1i})
ALORS (O_{1i},O_{2j}) est équivalent

Nous illustrons la mise en œuvre de cette approche en figure 63 pour une correspondance entre pattes d'oie. L'objet dans la première BD a une représentation détaillée. Sa base est de 13m. D'après les spécifications de la seconde BD, on peut prédire une représentation ponctuelle dans celle-ci ($(O_{2j})_{prédit}$). Dans l'autre sens, deux représentations possibles peuvent être prédites pour la première base à partir de la patte d'oie ponctuelle dans la seconde ($(O_{1i})_{prédit}$). Il reste donc à vérifier que les représentations stockées dans les bases appartiennent bien aux deux ensembles des représentations possibles prédites. Comme c'est le cas, on peut considérer les représentations comme équivalentes.

Cet exemple montre qu'il est possible de prédire plusieurs conditions sur la représentation homologue d'un objet pour une même représentation source. Ce cas de figure se présente fréquemment lorsque la prédiction s'applique de la base la moins détaillée vers la base la plus détaillée.

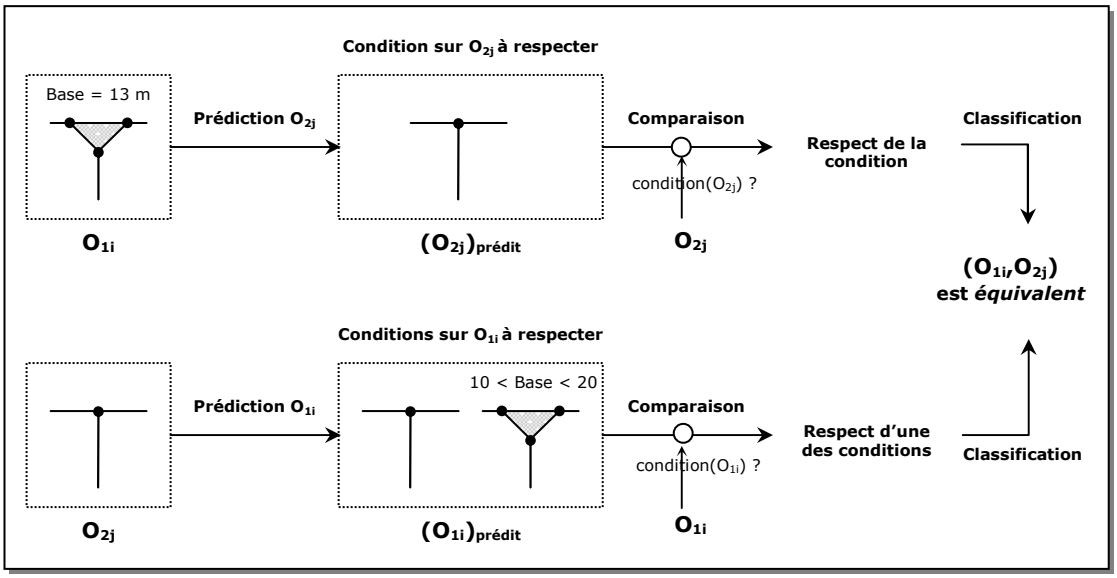


***Figure 63**. Mise en œuvre du contrôle inter-bases pour un couple de pattes d'oie en suivant la seconde approche : prédiction, comparaison, classification.*

C.5.5 BILAN DU CONTROLE INTER-BASES

Le contrôle inter-bases constitue l'étape d'évaluation proprement dite dans la méthode *MECO* que nous proposons. Sa particularité réside dans le fait que la conformité du couple et des représentations de chaque base est contrôlée en se fondant sur les représentations homologues correspondantes. Le contrôle inter-bases est entièrement automatique : il est réalisé à l'aide d'un système-expert. A l'issue de ce contrôle tous les couples d'objets appariés sont qualifiés : les représentations équivalentes et incohérentes sont identifiées.

Les connaissances qui guident l'interprétation peuvent être issues de deux sources différentes : les spécifications ou les données. La première source est exploitée lorsque les spécifications sont suffisamment précises et exhaustives. Dans ce cas, les règles utilisées sont introduites dans la base de règles par un expert du domaine. Cette tâche n'est pas automatique. L'expert doit se charger de définir les règles lui-même après une analyse des spécifications. Il doit donc reformuler les spécifications sous forme de règles pour ensuite les introduire dans le système-expert.

Mais ces règles peuvent être acquises automatiquement, à partir des données, en utilisant des techniques d'apprentissage automatique supervisé. Ces outils d'acquisition peuvent être utilisés si les spécifications sont insuffisantes, si celles-ci sont trop complexes à analyser ou encore, si on souhaite mener une évaluation en exploitant les spécifications constatées dans les données (cf. méthode *MACO*). L'intervention de l'expert peut donc être limitée et les règles acquises par apprentissage peuvent directement être introduites dans le système-expert.

Nous résumons l'étape du contrôle inter-bases à la figure 64.

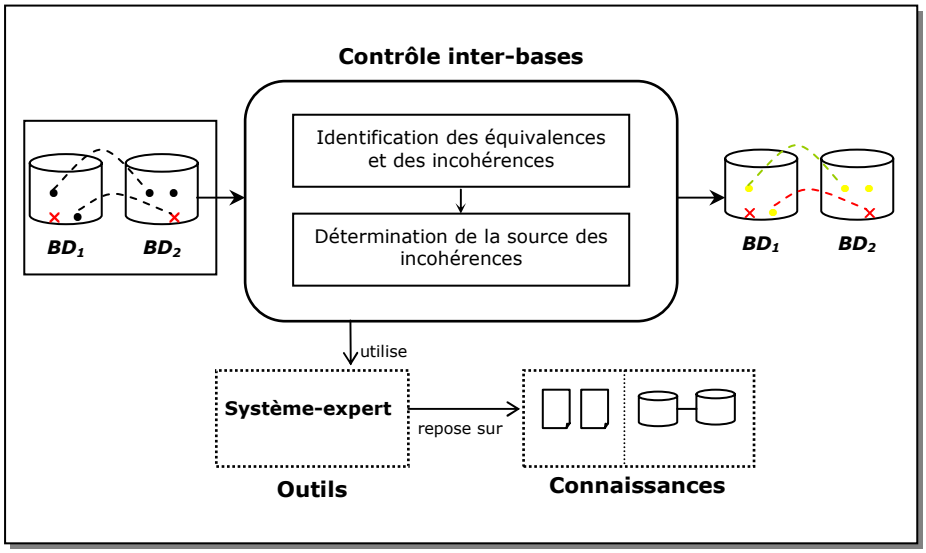


***Figure 64**. Le contrôle inter-bases permet d'évaluer la cohérence de chaque couple d'objets appariés et de détecter les erreurs inter-bases. Il exploite des connaissances qui peuvent être issues des spécifications ou des données. Ces connaissances sont décrites sous forme de règles de production et manipulées automatiquement par un système-expert.*

C.6 ÉVALUATION GLOBALE

C.6.1 OBJECTIF

Le contrôle inter-bases évalue la cohérence de chaque correspondance calculée. L'objectif de cette étape est de fournir une synthèse des résultats obtenus et de proposer certaines recommandations pour traiter les incohérences détectées.

C.6.2 SYNTHESE DES RESULTATS

CALCUL DES TAUX D'INCOHERENCES ET D'EQUIVALENCES

La première opération à effectuer est de chiffrer le nombre d'incohérences et d'équivalences. Les taux d'incohérences et d'équivalences doivent tenir compte des couples de cardinalité 1-0 et 0-1 qui ont été interprétés. On peut ainsi aboutir par exemple à la conclusion que sur 324 couples calculés (toutes cardinalités confondues), il y a 85% d'équivalences et 15% d'incohérences. Parmi ces incohérences, il est intéressant d'indiquer dans quelle base résident les erreurs (si la source est connue). Cela permet de montrer la répartition des erreurs.

Signalons que l'évaluation des couples de cardinalité 1-0 et 0-1 pour un groupe d'objets étudiés dépend parfois du contrôle d'un autre groupe d'objets. Pour les pattes d'oie par exemple, les différences de sélection des objets ne peuvent être analysées qu'après avoir étudié l'existence des routes. Si une route n'existe pas et que son absence est justifiée, alors celle de la patte d'oie l'est aussi.

Les taux d'incohérences et d'équivalences peuvent encore être décomposés en fonction de la nature des différences existant entre les objets. On peut ainsi distinguer

les différences d'existence (équivalence, déficit, excédent), les différences de modélisation (qui peuvent être décomposées selon les types de correspondance), les différences de classification (équivalence, confusion), les différences de position, d'orientation et de taille (en fonction du mode d'implantation des objets), et les différences touchant les attributs. Cette décomposition pourrait être envisagée en exploitant des méthodes de classification non supervisée, à l'image des travaux de [Bel Hadj Ali 2001] portant sur la classification des liens d'appariement à partir de mesures effectuées entre les objets.

Cette évaluation permet d'avoir une idée sur la manière dont les données respectent globalement leurs spécifications. Il se peut que l'on découvre qu'une règle n'est jamais respectée. Si c'est le cas, il faudra se poser la question de savoir si la règle déduite des spécifications a un sens, si celle-ci ne doit pas être révisée, et s'il faut considérer toutes les représentations comme erronées.

NIVEAU DE GRAVITE DES ERREURS

On peut également envisager d'attribuer un niveau de gravité aux erreurs rencontrées (ou leur associer un coût) en fonction du type d'erreur, de l'importance de la différence et de sa fréquence. L'attribution de ce niveau peut être fixé en fonction du contexte dans lequel les bases intégrées seront utilisées. Une erreur de forme d'une parcelle par exemple aura davantage d'importance qu'une erreur de position d'une route pour des applications cadastrales. Ce niveau de gravité peut également être déterminé en fonction des spécifications de qualité des bases. Celles-ci distinguent généralement des seuils qui sont des *objectifs* de qualité à atteindre et ceux qui sont des *exigence*s. L'objectif correspond à un besoin implicite. Il n'y a pas d'obligation de l'atteindre immédiatement au moment de la première version de la base, mais les moyens sont mis en œuvre pour y parvenir, au fur et à mesure de sa maintenance. L'exigence de qualité est quant à elle un besoin exprimé. On ne peut pas faire moins bien que le seuil déterminé. Il y a obligation de l'atteindre avec corrections immédiates des données si nécessaire au moment de la constitution de la base. Une erreur pour laquelle un seuil d'exigence de qualité aurait été fixé pourrait donc être considérée comme plus grave qu'une erreur pour laquelle le seuil est vu comme un objectif. L'affectation d'un niveau de gravité permettrait de distinguer les erreurs qu'il faut absolument réparer des erreurs de moindre importance.

PRESENTATION DES RESULTATS

Pour la présentation des résultats, on peut emprunter certaines solutions adoptées dans les rapports de contrôle qualité [David et Fasquel 1997] et les adapter à notre contexte.

Les différences de position entre les objets appariés peuvent être analysées en créant un histogramme de réparation des écarts. Cet histogramme peut être associé à une grille régulière de biais régionalisé (figure 65). L'information apportée par cette grille est intéressante car elle permet d'identifier les zones de l'espace étudié pour lesquelles les écarts de position sont plus importants. Sa construction est très simple. Après avoir fixé un pas à la grille, on calcule, pour un échantillon de cellule, la moyenne des écarts de position en abscisse et en ordonnée (le biais[20]). Dans notre cas, on sélectionne chaque couple d'objets appariés présent dans la cellule tirée et on

[20] Rappelons que le biais correspond à l'écart entre l'espérance d'une mesure d'une grandeur et la valeur nominale de cette grandeur.

choisit une référence. On reporte ensuite la valeur calculée sous forme de vecteur au niveau de la cellule tirée et le biais est ainsi représenté. Le choix de la référence n'a aucune importance. L'objectif est simplement de montrer la répartition des écarts et leur grandeur dans l'espace. L'analyse de cette grille doit toutefois être menée avec précaution car suivant le type d'objets traité, leur répartition et leur présence dans l'espace peut fortement varier (des échangeurs routiers ne se rencontrent pas partout par exemple).

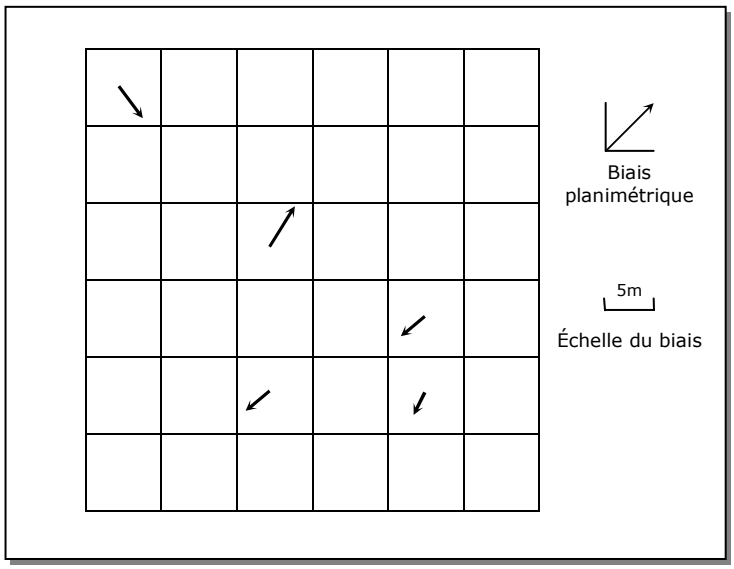


***Figure 65**. Grille de biais régionalisé.*

Pour exposer les erreurs de déficit et d'excédent (appelée aussi erreur d'omission et de commission), de même que les erreurs de confusion, nous aurions pu songer à utiliser les matrices de confusion [Congalton 1991]. Ces matrices sont régulièrement employées pour étudier l'exactitude de classification des objets et leur niveau de complétude. Dans notre cas cependant, elles ne sont pas tout à fait adaptées. Une matrice de confusion suppose l'existence d'un jeu de données de référence et d'un jeu à contrôler, lesquels répondent aux mêmes spécifications. Les différences de représentation correspondent aux erreurs à mettre en évidence, ce qui n'est pas le cas pour nous.

On peut tout de même exposer les résultats sous forme de tableau en indiquant en ligne les différentes représentations possibles de la première BD et en colonne les représentations possibles de la seconde en tenant compte des spécifications des deux bases (analyse croisée). C'est ce qui est représenté à la figure 66 pour les pattes d'oie. Les différents types de correspondances sont illustrés avec leur taux d'incohérences et d'équivalences respectifs. Dans cet exemple, pour 500 couples traités, 360 équivalences ont pu être détectées contre 140 incohérences.

Signalons que la présentation des résultats sous cette forme convient bien lorsqu'il existe des différences de modélisation et que leur nombre est fini, mais ce tableau n'est pas toujours applicable. La cohérence pourrait ne toucher que la position et la longueur des objets par exemple. Dans ce cas, une simple énumération des caractères contrôlés peut être présentée.

BD_1 \ BD_2		 Base < 20m	 Base > 20m	**Néant**	**Total**
	80/80 équivalences	8/8 incohérences	5/5 incohérences	40/50 équivalences 10/50 incohérences	120/143 équivalences 23/143 incohérences
 Base < 10m	8/8 incohérences	4/4 incohérences	0/0	20/20 incohérences	0/32 équivalences 32/32 incohérences
 10 < Base < 20	105/105 équivalences	30/30 incohérences	20/20 incohérences	28/30 équivalences 2/30 incohérences	133/185 équivalences 52/185 incohérences
 Base > 20m	15/15 incohérences	10/10 incohérences	107/110 équivalences 3/110 incohérences	0/0	107/135 équivalences 28/135 incohérences
Néant	2/2 incohérences	3/3 incohérences	0/0	0/0	0/5 équivalences 5/5 incohérences
Total	185/210 équivalences 25/210 incohérences	0/55 équivalences 55/55 incohérences	107/135 équivalences 28/135 incohérences	68/100 équivalences 32/100 incohérences	360/500 équivalences 140/500 incohérences

***Figure 66**. Illustration de résultats d'évaluation.*

C.6.3 RECOMMANDATIONS

En plus d'une synthèse des résultats, on peut également envisager de fournir certaines recommandations pour le traitement futur des incohérences. On peut ainsi spécifier qu'une erreur doit être réparée ou que cette erreur doit être signalée à l'utilisateur. Cette recommandation est étroitement liée avec le niveau de gravité de l'erreur. On peut également proposer de modifier les spécifications et de les enrichir pour qu'elle reflète mieux le contenu des bases. On peut s'apercevoir en effet qu'une spécification est trop contraignante et qu'en pratique, elle est rarement respectée dans les données, peut-être à juste titre.

C.7 MANIPULATION DES CONNAISSANCES POUR LES CONTROLES INTRA-BASE ET INTER-BASES PAR UN SYSTEME-EXPERT

Nous venons de présenter toutes les étapes de la méthode *MECO* qui définit la démarche à adopter pour réaliser l'évaluation de la cohérence inter-représentations. Pour les contrôles intra-base et inter-bases, nous avons vu que ces étapes étaient automatisées grâce à l'emploi d'un système-expert. Dans cette partie, nous présentons les caractéristiques de ces systèmes et discutons de leur intérêt pour une tâche d'interprétation comme la nôtre.

C.7.1 ORIGINE DES SYSTEMES-EXPERTS

Le développement des systèmes experts est étroitement lié à l'évolution et aux enseignements tirés des premières recherches effectuées en intelligence artificielle.

L'intelligence artificielle a pour ambition de faire reproduire par des machines des tâches et des raisonnements complexes effectués par des humains. Les premiers résultats obtenus dans ce sens, à l'époque des années 60, furent particulièrement encourageants [Russell et Norvig 2003]. On vit notamment apparaître des programmes de traduction automatique, de démonstration de théorèmes (*Logic Theorist*), un système de résolution de problème de tout ordre (GPS – *Global Problem Solver*), ou encore un programme capable de dialoguer avec un humain (ELIZA). L'approche adoptée pour réaliser de tels programmes fut une approche combinatoire : on cherche dans l'espace des possibilités la solution désirée.

Très vite, le problème de l'explosion combinatoire apparu. La puissance des ordinateurs de l'époque fut insuffisante pour traiter l'exploration de toutes les solutions de tâches réelles complexes. Les chercheurs en IA réalisèrent alors que pour faire apprendre et comprendre des choses à une machine, il était nécessaire de lui fournir des connaissances sur le domaine considéré, à l'image de ce que font les humains. C'est ainsi que naquirent les premiers systèmes experts.

C'est une équipe de l'université de Stanford qui proposa à la fin des années 60 le système DENDRAL [Buchanan et al. 1969]. La tâche qui incomba à ce programme fut d'analyser automatiquement les spectres de masse pour comprendre la structure moléculaire d'un corps chimique. Afin d'aider le système à réaliser cette analyse, des heuristiques provenant de connaissances du domaine (la chimie) furent introduites. Peu de temps après, le système expert MYCIN fut développé pour aider à diagnostiquer des infections bactériennes du sang et proposer un traitement thérapeutique [Shortliffe 1976]. L'idée retenue pour sa conception fut de séparer les connaissances nécessaires à la prise de décision, des programmes qui les manipulent. L'architecture des systèmes-experts était définie.

La période qui suivit fut particulièrement marquée par le foisonnement de projets de développement de systèmes-experts et ceci, dans des domaines extrêmement variés, aussi bien dans les laboratoires de recherche qu'en industrie. La géographie ne fut pas mise à l'écart. Des exemples peuvent être trouvés dans [Guigo et al. 1995, Openshaw et Openshaw 1997].

C.7.2 CARACTERISTIQUES D'UN SYSTEME-EXPERT

Les systèmes experts furent donc essentiellement développés pour résoudre des tâches d'analyse, de diagnostic et de prise de décisions, en simulant certains raisonnements humains. [Feigenbaum 1981] définit un système-expert comme suit :

> « *An expert system is an intelligent computer program that uses knowledge and inference procedures to solve problems that are difficult enough to require significant human expertise for their solutions* ».

Bien que depuis cette époque, les systèmes-experts de seconde génération ont été proposés [David et al. 1993b], cette définition générale est toujours valable aujourd'hui.

La caractéristique fondamentale d'un système expert est de dissocier les connaissances utiles à la résolution d'un problème et le programme qui les manipule. Ainsi, l'architecture générale d'un système expert est composée d'une *base de faits*, d'une *base de règles* et d'un *moteur d'inférence* (figure 67).

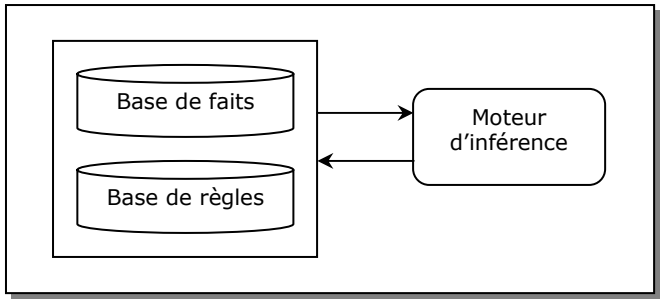


***Figure 67**. Composants d'un système-expert*

La **base de règles** contient les connaissances du domaine. Ces règles sont destinées à être appliquées sur les faits pour aboutir au diagnostic recherché. Elles représentent donc des éléments du « savoir faire » de l'expert. Dans notre contexte, cette base renferme les spécifications des bases de données géographiques à intégrer, décrites sous forme de règles. Sa création constitue la principale difficulté dans l'élaboration du système.

La **base de faits** contient les données du problème, un savoir déclaratif. Au départ, elle renferme uniquement des faits initiaux mais s'enrichit au fur et à mesure des actions effectuées par le moteur d'inférence. Dans le cadre de ce travail, les faits correspondent aux objets géographiques des deux bases qui sont représentés de manière symbolique, sous forme d'un vecteur d'attributs. Nous y reviendrons ultérieurement.

Le **moteur d'inférence** est la composante générique du système. Il est chargé de construire le raisonnement et de prouver une hypothèse posée (dans le cas d'un chaînage arrière) à partir des faits et des règles. C'est donc lui qui se charge d'enchaîner les règles et de les appliquer sur les faits choisis : il simule le raisonnement de l'expert. Le moteur d'inférence que nous utilisons est celui proposé par JESS[21] [Friedman-Hill 2003]. Nous présenterons ses caractéristiques dans le chapitre relatif aux expérimentations menées (chapitre E).

[21] JESS est téléchargeable sur le site : http://herzberg.ca.sandia.gov/jess/

Notons qu'on regroupe généralement sous le terme de *base de connaissances* la base de faits et la base de règles [Ayel et Rousset 1990].

A ces modules fondamentaux du système s'ajoutent quelques briques supplémentaires : des interfaces d'aide à l'acquisition des connaissances et à l'explication.

REGLES DE PRODUCTION

La base de règles est constituée le plus souvent de ce qu'on appelle des *règles de production*. La forme générale d'une règle de production est la suivante :

Si *conditions* **ALORS** *conclusion*

La partie gauche de la règle est appelée *prémisse*. Il s'agit d'une hypothèse qui doit être vérifiée pour que la conclusion soit déclenchée. Les opérateurs de la prémisse sont des opérateurs de comparaison. La partie droite de la règle est la *conclusion*. Les opérateurs de la conclusion sont des opérateurs d'affectation.

Prenons un exemple. Imaginons la règle de production suivante :

Si la superficie d'une parcelle est < 500m² **ALORS** la saisie est erronée

Pour que la règle soit activée, la parcelle doit vérifier la condition relative à sa superficie. Ce qu'il est important de noter, c'est que cette condition va s'appliquer sur les *faits*. Si la valeur de la superficie d'une parcelle quelconque — *fait* stocké dans la base — est inférieure à la valeur proposée dans la condition de la règle, celle-ci sera déclenchée. Le déclenchement conduit à l'exécution des affectations de valeurs aux attributs présents en conclusion. Dans notre cas, il faut affecter la valeur « saisie erronée » à un attribut du fait prêt à recevoir cette conclusion.

Précisons que les prémisses peuvent être composées, c'est-à-dire qu'une hypothèse peut être constituée de plusieurs conditions reliées par des opérateurs ET ou OU par exemple.

Ce qui distingue essentiellement les systèmes-experts, c'est la manière dont ils construisent le raisonnement, en plus du langage de représentation des connaissances qu'ils utilisent (logique des propositions, logique des prédicats).

LANGAGE DE REPRESENTATION DES CONNAISSANCES

Les systèmes-experts peuvent être qualifiés d'ordre 0, d'ordre 0+ ou d'ordre 1 en fonction du langage de représentation dans lequel il manipule les faits et les règles.

Lorsque les valeurs possibles des faits sont des variables booléennes (vrai, faux), le langage est dit de la logique d'ordre 0. Il s'agit d'un langage manipulable par la logique des propositions [Kayser 1997]. Un ensemble de formules peut être défini composées des connecteurs « et, ou, donc, équivalent à, non » et de variables booléennes. Si les variables peuvent être valuées (appartenant à un domaine fini de symboles comme X=point, ou X=noir) la logique est qualifiée d'ordre 0+.

Lorsque les valeurs possibles des faits sont des variables (valeurs réelles), le langage est dit de la logique d'ordre 1. Il s'agit d'un langage manipulable par la logique des prédicats [Kayser 1997]. Un ensemble de formules peut être défini composées des connecteurs « et, ou, donc, équivalent à, non », de quantificateurs « pour tout, il existe », et de variables.

Le moteur utilisé pour développer notre système-expert (décrit dans le chapitre E) permet de manipuler des connaissances représentées dans un langage de la logique d'ordre 1. Les règles que nous avons développées sont toutefois définies dans un langage d'ordre 0+. Les algorithmes d'apprentissage utilisés ne nous permettent pas d'acquérir des règles dans un langage d'ordre supérieur.

METHODES DE RAISONNEMENT D'UN MOTEUR D'INFERENCE

Il existe plusieurs méthodes pour produire un raisonnement par l'intermédiaire du moteur d'inférence. Pour répondre à un problème posé, le moteur d'inférence va se charger de l'activation des règles et de leur enchaînement. L'enchaînement peut être élaboré de deux manières différentes (figure 68) :

- Par *chaînage avant* : le type de raisonnement est la *déduction* (modus ponens) et est guidé par les données. Pour déduire un nouveau fait, le moteur d'inférence vérifie si les prémisses des règles sont vraies, ceci pour chaque fait initial. Si c'est le cas, la règle est activée, la valeur des attributs présents dans la conclusion de la règle est affectée, et le fait est modifié. Celui-ci est propagé et peut alors à nouveau être déclenché par d'autres règles. Le processus d'enchaînement s'arrête lorsque tous les faits ont été épuisés.
- Par *chaînage arrière* : le type de raisonnement est l'*abduction* (modus tollens) et est guidé par le but recherché. Plutôt que de rechercher les hypothèses qui sont vraies, le moteur d'inférence va essayer de démontrer les hypothèses données. On ne veut donc plus déduire un fait mais on souhaite identifier quelles règles permettent d'aboutir à une conclusion donnée et rechercher les faits qui sont nécessaires au déclenchement des règles. Ce sont donc les règles ayant pour conclusion le but fixé qui sont sélectionnées. Elles constituent les sources. A partir de celles-ci, on peut en déduire les conditions qui doivent être démontrées (prémisses), lesquelles peuvent à leur tour être considérées comme de nouveaux sous-buts susceptibles d'apparaître en conclusion d'autres règles. On poursuit ainsi le processus récursivement jusqu'à ce que tous les sous-buts soient démontrés, autrement dit, jusqu'à ce que tous les faits soient établis.

CHAINAGE AVANT	**CHAINAGE ARRIERE**
Règles : **Si** A ou B **alors** C **Si** D **alors** E **Si** E ou C **alors** F	**Règles :** **Si** A ou B **alors** C **Si** D **alors** E **Si** E ou C **alors** F
Question : Que déduit-on de B ?	**Question :** Comment peut-on conclure F ?
Réponse : F B implique C et C implique F.	**Réponse :** Par A, B ou D F suppose E ou C vrai. C suppose A ou B vrai. E suppose D vrai

***Figure 68**. Méthodes de raisonnement d'un moteur d'inférence : le chaînage avant et le chaînage arrière*

Certains moteurs d'inférence sont dotés des deux mécanismes de raisonnement. Le chaînage arrière peut ainsi être associé au chaînage avant pour former le *chaînage mixte*.

RESOLUTION DES CONFLITS ET VALIDATION

Il est possible que, pour aboutir à une même hypothèse, ou suivant les faits présents dans la base, plusieurs règles soient candidates au déclenchement. Lorsque plusieurs règles sont susceptibles d'être activées pour un même fait, c'est la structure de contrôle du moteur d'inférence qui décide de la règle à appliquer. De nombreux systèmes experts choisissent la première règle valide qu'ils rencontrent [Guigo et al. 1995]. Dans ce cas, l'ordre de saisie des règles dans la base est déterminant pour le processus de raisonnement. D'autres systèmes offrent la possibilité d'affecter une pondération aux règles [Guigo et al. 1995]. De cette manière, chaque règle possède un ordre de priorité.

Certaines règles peuvent également être en conflit provoquant des conclusions contradictoires. C'est ici qu'intervient la notion de *cohérence* d'une base de connaissances [Ayel et Rousset 1990] et sa validation. Les règles de production doivent satisfaire certaines propriétés, notamment l'absence de redondance, l'absence de bouclage, l'absence de conflit et l'absence de chaînes contradictoires. A l'issue du développement d'un système-expert, celui-ci doit donc être *validé*.

C.7.3 INTERETS D'UTILISER UN SYSTEME-EXPERT

La spécificité d'un système-expert qui, rappelons-le, est de séparer les connaissances des programmes qui les manipulent, leur confère une grande souplesse. Ainsi, la séparation offre plusieurs avantages :

- Elle permet de maintenir et d'enrichir la base de connaissances sans modifier la procédure d'inférence, les connaissances n'étant plus « noyées » dans le programme qui les gère.
- Elle permet de traiter un volume important de connaissances, ce traitement pouvant s'avérer difficile voir impossible en suivant une approche procédurale.
- Elle permet de réutiliser le moteur d'inférence pour différents domaines, les algorithmes n'étant pas spécifiques aux données qu'il traite.
- Elle permet enfin d'énoncer les règles dans un langage qui est plus facilement compréhensible qu'un langage de programmation traditionnel. Il est donc facilement programmable.

L'intérêt d'utiliser un système-expert réside en outre dans le fait qu'il est capable de fournir une explication sur le raisonnement effectué. Il peut garder une trace des règles activées pour aboutir à une solution recherchée.

Ces caractéristiques sont particulièrement intéressantes dans le cadre de notre travail. Les spécifications des bases de données géographiques doivent être reformulées en termes de règles de production et celles-ci peuvent être complexes et très nombreuses. Pour cette raison, il n'est pas envisageable de suivre une approche procédurale. Les programmes seraient beaucoup trop complexes et peu faciles à modifier. Le développement d'un système-expert constitue une bonne réponse à

l'automatisation de nos étapes de contrôles intra-base et inter-bases. Nous l'illustrerons par l'expérimentation.

C.7.4 DEMARCHE DE CONCEPTION ADOPTEE

Au début des années 80, la conception d'un système-expert se caractérisait par 5 phases différentes [Hayes-Roth et al. 1983, d'après Krivine et David 1991] :

- L'*identification* qui consiste à déterminer les caractéristiques du problème.
- La *conceptualisation* qui s'attache à trouver les concepts représentant les connaissances.
- La *formalisation* qui vise à définir la structure du système et à choisir le langage de représentation.
- L'*implémentation* qui consiste à écrire les règles et créer la base de connaissance.
- La *validation* dont la tâche est de contrôler la cohérence du système.

Cette démarche de conception caractérisait les systèmes-experts dits de première génération. La connaissance était intégrée « en vrac » dans la base. La méthode de résolution de problème était entièrement laissée à la charge du moteur d'inférence.

A la suite de travaux de William Clancey sur le système MYCIN [Clancey 1983, 1985], l'hypothèse admettant que chaque règle constituait en elle-même un morceau de connaissance valide et indépendante fut controversée. Ainsi, il apparut que l'ordre dans lequel apparaissaient les prémisses d'une règle guidait l'ordre dans lequel les règles étaient activées. Les prémisses ne possédaient pas toutes le même statut mais leur représentation dans les règles leur conférait un statut unique. En fait, Clancey mis en évidence l'existence de connaissances implicites du domaine dont une chaîne d'inférences guidant le processus. La base de règles ne formait pas un ensemble de connaissances peu structurées. Au contraire, il existait implicitement dans cette base une forme particulière de raisonnement.

Ce constat fut à l'origine des systèmes-experts dits de seconde génération et des méthodes d'acquisition de connaissances fondées sur les modèles [David et al. 1993b, Thomas 1996]. Ainsi, il ne suffit plus de transférer la connaissance de l'expert à la base pour concevoir un système-expert (tâche qui s'avère déjà très délicate) mais de modéliser le domaine et le raisonnement afin de mieux structurer le système, le raisonnement constituant lui-même une connaissance.

> *C'est cette approche que nous avons adoptée dans cette thèse. La structuration de notre système-expert est guidé par la méthode de résolution de problème que nous avons définie : la méthode MECO.*

Les étapes de la méthode qui requièrent l'utilisation du système-expert (le contrôle intra-base et le contrôle inter-bases) sont chacune associées à une base de règles qui leur est propre. L'exploitation de ces bases se fait à un moment déterminé, en suivant le déroulement de la méthode *MECO*.

C.8 SYNTHESE DE LA METHODE *MECO*

Nous synthétisons *MECO* à la figure 69. Cette méthode est composée de différentes étapes : l'enrichissement, le contrôle intra-base, l'appariement, le contrôle inter-bases, l'évaluation globale. Chaque étape peut être automatisée en utilisant des outils qui peuvent correspondre à des algorithmes (outils d'analyse spatiale et d'appariement) ou à un système-expert (outil pour les contrôles intra-base et inter-bases). Ces outils reposent sur des connaissances acquises en appliquant la méthode *MACO*. Nous présentons *MACO* dans le chapitre suivant.

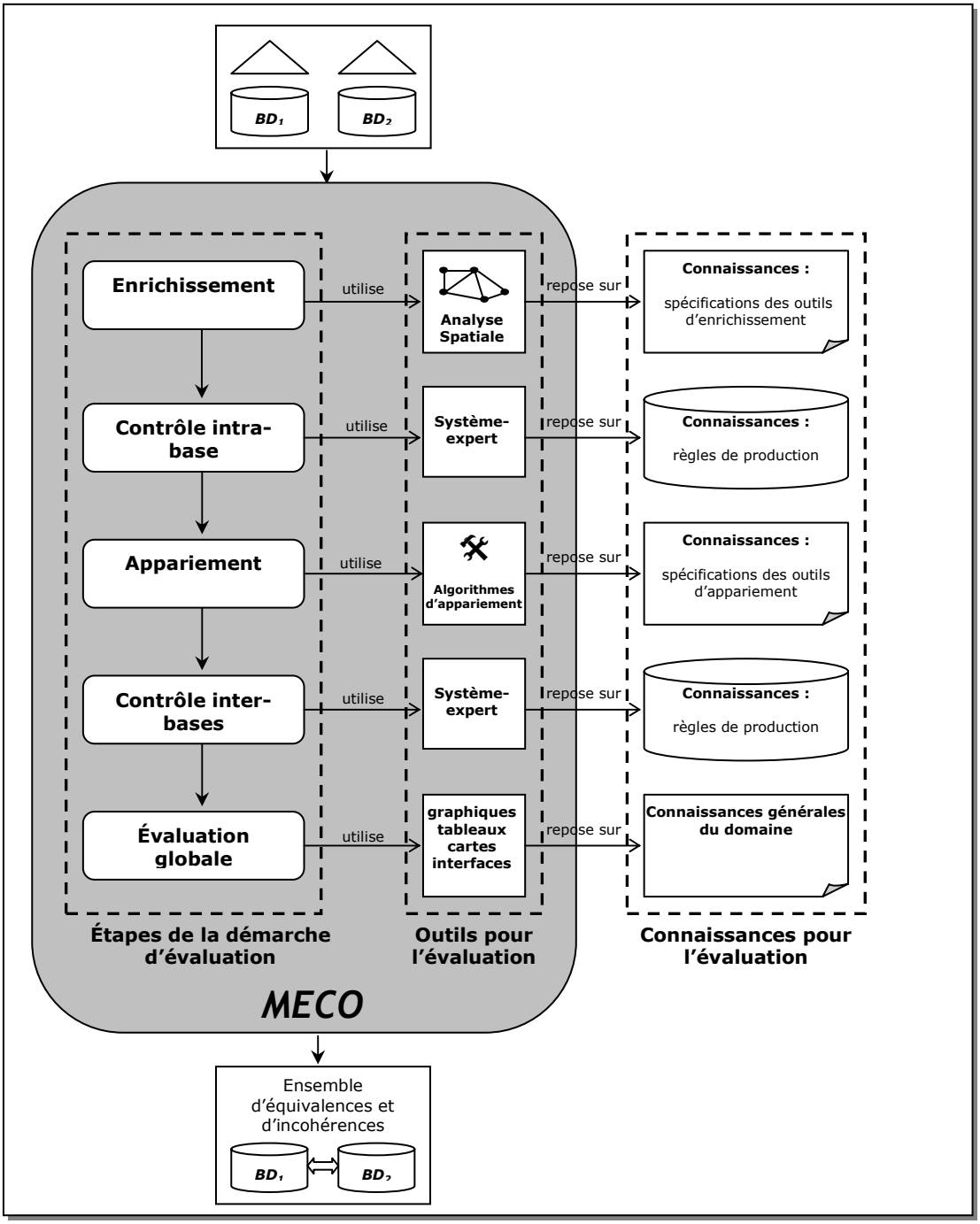


***Figure 69**. Synthèse de la méthode MECO*

CHAPITRE D

MACO : METHODE D'ACQUISITION DES CONNAISSANCES POUR L'EVALUATION DE LA COHERENCE

D.1 INTRODUCTION

Dans le chapitre précédent consacré à la méthode d'évaluation de la cohérence *MECO*, nous avons mentionné à chaque étape la nécessité d'utiliser des connaissances. Nous avons vu qu'il existait deux sources principales de connaissances : les spécifications et les données. Le problème qui se pose est de savoir comment extraire les connaissances de ces sources. Certaines de ces connaissances sont destinées à déterminer les concepts à extraire des données, définir les outils d'enrichissement et d'appariement. D'autres sont destinées à peupler la base de règles du système-expert. La méthode *MACO* que nous présentons dans ce chapitre est là pour y répondre. Elle constitue la seconde partie de notre contribution méthodologique pour l'évaluation de la cohérence inter-représentations (figure 70).

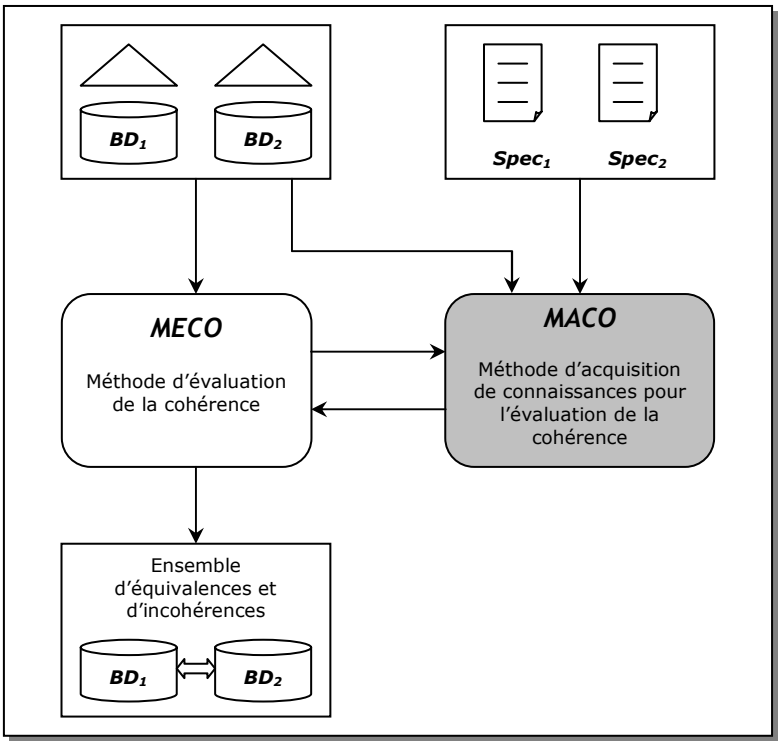


***Figure 70**. MACO : seconde méthode proposée dans la méthodologie générale d'évaluation de la cohérence inter-représentations.*

Le recueil des connaissances dans notre contexte n'est pas une tâche triviale. Les difficultés s'expliquent par la structure actuelle des spécifications et la part d'informations absentes de ces documents (cf. chapitre B). Nous sommes confrontés à un problème d'acquisition de connaissances qui nécessite, pour y faire face, de faire appel à des outils spécifiques.

La méthode que nous proposons est composée de deux étapes : une étape d'analyse des spécifications (D.3.1.) et une étape d'apprentissage (D.4.2). L'analyse des spécifications est systématique et est réalisée par un expert du domaine. Pour faciliter l'analyse et la comparaison des documents, nous proposons de formaliser la représentation des spécifications selon un modèle que nous avons défini (D.3.2). L'étape d'apprentissage automatique n'est pas obligatoire. Elle s'impose si les spécifications ne renferment pas suffisamment de connaissances pour réaliser le contrôle inter-bases. Elle peut être utile pour aider à acquérir les règles automatiquement lorsque les spécifications sont trop complexes à analyser

interactivement. Elle permet également de découvrir les spécifications *constatées*, celles que respectent en pratique les personnes chargées de la saisie des données.

Ce chapitre est organisé de la manière suivante. Dans la section D.2., nous exposons la problématique générale de l'acquisition de connaissances qui fait partie d'un domaine de recherche à part entière en intelligence artificielle. La section D.3. est consacrée à l'acquisition de connaissances issues des spécifications (présentation de l'étape d'analyse et du modèle des spécifications). La partie suivante (D.4.) expose d'abord les principes de l'apprentissage automatique supervisé. La mise en œuvre de ces techniques dans notre contexte est ensuite présentée. Nous concluons finalement le chapitre en synthétisant la méthode *MACO* (D.5.) et en présentant la démarche à suivre pour appliquer la méthodologie générale (D.6.). On peut se reporter aux figures 87 et 88 pour avoir une vue d'ensemble de *MACO* et sa relation avec *MECO*.

D.2 PROBLEMATIQUE DE L'ACQUISITION DES CONNAISSANCES

L'acquisition des connaissances est un domaine de recherche à part entière en intelligence artificielle. Le problème du recueil de connaissances s'est rapidement posé notamment avec l'apparition des systèmes-experts (SE).

Les travaux en acquisition des connaissances se sont clairement divisés en deux groupes ayant des objectifs aujourd'hui relativement différents mais toutefois complémentaires : le *transfert d'expertise* à l'aide de techniques d'élicitation des connaissances (« *Knowledge Elicitation* ») et l'acquisition de connaissances par *apprentissage automatique* (« *Machine Learning* »).

Initialement, l'acquisition des connaissances était perçue comme une activité destinée à rassembler l'information nécessaire à la résolution d'un problème particulier, et de codifier celle-ci (la transcrire dans un formalisme particulier) pour l'introduire dans une machine [Krivine et David 1991]. L'enjeu de ce transfert d'expertise était d'extraire l'information tacite, dont les experts n'ont pas conscience, en évitant d'introduire des biais. Les techniques utilisées sont nombreuses et empruntées aux sciences cognitives (psychologie, neurosciences, ergonomie, linguistique,…) [Aussenac 1989]. Le *cogniticien* a ainsi recours à des techniques de verbalisation ou d'interviews pour extraire le savoir et le savoir-faire de l'expert (les raisonnements appliqués pour résoudre un problème particulier et les ressources qu'il utilise). Il analyse également des protocoles, des manuels et des rapports, afin d'obtenir le plus d'informations possible sur le domaine étudié.

Les travaux sur le transfert d'expertise ont évolué au cours de la dernière décennie dans une perspective beaucoup plus large d'*ingénierie des connaissances*[22]. Cette évolution est liée à l'apparition des systèmes-experts dits de seconde génération [David et al. 1993b]. Le travail du cogniticien aujourd'hui n'est plus seulement de récolter l'information auprès d'experts. Il doit aussi l'organiser et la modéliser. L'acquisition des connaissances est ainsi considérée comme un problème de construction de modèles [Krivine et David 1991]. Cette nécessité de modélisation est apparue notamment à cause du décalage trop important existant entre le langage dans lequel l'expert exprimait ses connaissances et le niveau d'abstraction des

[22] On peut consulter à ce sujet le site de la communauté française d'acquisition et d'ingénierie des connaissances (GRACQ) : http://www.irit.fr/GRACQ/index.shtml

formalismes de représentation des connaissances : « *tout comme la pensée est limitée par notre langage et nos modèles mentaux, les bases de connaissance des systèmes-experts sont limités par l'expressivité de nos langages informatiques pour la représentation des connaissances et par notre habilité à utiliser ces langages* » [Musen 1993, tiré de Lépy 1997]. Avant de concevoir un système à base de connaissances, il est donc préconisé de construire des modèles. Des cadres conceptuels permettant d'aider à construire ces modèles ont été définis en distinguant la *modélisation du domaine* et celle relative aux *processus de raisonnement*. C'est notamment le cas de la méthode KADS [Wielinga et al. 1992].

Dans notre contexte, nous avons choisi de modéliser les spécifications des bases de données géographiques. L'objectif principal de cette modélisation est de mieux structurer l'information, d'harmoniser leur description et de rendre les spécifications ainsi plus facilement comparables et exploitables. En pratique, cette modélisation permet également d'analyser les spécifications de manière approfondie et d'identifier les descriptions trop imprécises.

Dans une certaine mesure, nous suivons ainsi une approche d'acquisition fondée sur la construction de modèles. Pour établir le lien avec la méthode KADS, on pourrait considérer que ce modèle représente en partie l'expertise du domaine. La méthode *MECO* que nous avons présentée dans le chapitre précédent constituerait, quant à elle, le modèle de raisonnement (la *méthode de résolution de problème*). En pratique, nous n'avons pas suivi rigoureusement une méthode de conception de base de connaissances. La mise en œuvre de KADS est réputée délicate et demande une grande expérience. Nous avons jugé qu'il n'était pas utile de l'appliquer dans ce travail. Les efforts que nous aurions dû fournir auraient largement dépassé les bénéfices de formalisation que nous aurions pu en retirer.

Parallèlement à ces travaux sur le transfert d'expertise et les activités de modélisation, un autre courant s'est développé constituant une alternative à ces techniques d'acquisition de connaissances : il s'agit de l'*apprentissage automatique* [Mitchell 1997]. Les techniques de transfert d'expertise n'ont pas toujours suffit à extraire les connaissances tacites des experts. La principale difficulté réside dans le fait que la capacité d'introspection des experts est limitée : « *Au fur et à mesure que l'expert acquiert de l'expertise, sa connaissance déclarative* [dont il a conscience] *devient procédurale et il perd conscience de ce qu'il sait* » [Lépy 1997].

Il est donc difficile pour eux d'exprimer précisément ce qu'ils savent et comment ils raisonnent. C'est le problème bien connu en intelligence artificielle du *goulot d'étranglement de l'acquisition des connaissances* [Feigenbaum 1981]. Ce problème a d'ailleurs déjà été rencontré en cartographie dans le cadre de l'automatisation de la généralisation cartographique [Weibel et al. 1995, Mustière et Zucker 2002].

Les méthodes d'apprentissage automatique supervisé ont été conçues pour répondre à cette problématique. Elles permettent de recueillir des connaissances implicites à partir d'un ensemble d'exemples fournis par l'expert. En suivant cette approche, l'expert ne doit plus expliquer le raisonnement qu'il effectue et préciser les connaissances qu'il utilise pour résoudre une tâche particulière, les méthodes d'apprentissage sont censés les découvrir. La principale difficulté est alors de construire des exemples d'apprentissage pertinents, c'est-à-dire qui contiennent les bonnes informations, pour permettre d'apprendre un modèle reflétant le raisonnement de l'expert.

Le problème du goulot d'étranglement de l'acquisition des connaissances et l'utilité de l'apprentissage automatique apparaissent clairement dans le cadre de notre

travail. Nous disposons de documents renfermant une part importante de la connaissance permettant d'interpréter les différences (les spécifications) mais ceux-ci ne sont pas suffisants. Certaines règles de saisie qu'utilisent les experts n'y sont pas mentionnées et si nous souhaitons tenir compte de ces connaissances implicites, nous devons les recueillir d'une manière ou d'une autre. On pourrait envisager d'interroger les personnes chargées de la saisie des données mais les différentes expériences menées précédemment montrent que la granularité des règles obtenues n'est généralement pas assez fine et que celles-ci ne sont pas suffisamment formalisées [Weibel et al. 1995, Kilpeläinen 2000], outre le fait que les entretiens sont particulièrement longs à mettre en œuvre. L'apprentissage automatique semble constituer une bonne alternative, d'autant plus que ces techniques ont déjà prouvé leur efficacité dans le domaine de l'information géographique, pour répondre à des problématiques assez similaires [Esposito et al. 1997, Sester 2000, Mustière 2001]. Cet outil d'acquisition de connaissances devrait ainsi permettre d'extraire les spécifications implicites à partir des données.

La section suivante porte sur l'acquisition des connaissances issues des spécifications. Nous exposerons ensuite la manière d'extraire des connaissances à partir de données en utilisant l'apprentissage automatique.

D.3 ACQUISITION DES CONNAISSANCES ISSUES DES SPECIFICATIONS

D.3.1 ANALYSE DES SPECIFICATIONS

Lors de la présentation de la méthode *MECO*, nous avons fait clairement apparaître la nécessité d'analyser les spécifications pour réaliser les étapes proposées. Les spécifications doivent être étudiées en profondeur pour comprendre ce que contiennent les bases et identifier les règles de saisie des objets qu'elles contiennent.

L'analyse des spécifications est une tâche assez fastidieuse car les spécifications sont décrites en langue naturelle, dans des documents volumineux, ce qui rend leur manipulation automatique impossible aujourd'hui. C'est donc aux cogniticiens (ou plus généralement aux experts du domaine) de mener cette étude interactivement.

La démarche d'analyse à entreprendre peut se traduire par une série de questions à se poser qui portent à la fois sur une seule base (analyse individuelle des documents) ou sur les deux bases en même temps (analyse croisée des documents) pour un phénomène à traiter (ex : les routes). Nous listons les principales interrogations ci-dessous. Pour chaque base :

- Existe-t-il plusieurs classes représentant le même phénomène ?
- Existe-t-il des règles de modélisation et de sélection associées à ces classes ? Quelles sont-elles ? Sont-elles directement formalisables ?
- Quelles règles de saisie peut-on contrôler sur les objets des bases ? Que peut-on contrôler individuellement (contrôle intra-base) ? Que peut-on contrôler en utilisant conjointement les représentations des deux bases (contrôle inter-bases) ?
- Doit-on enrichir les données ? Quel doit être l'enrichissement (caractères à extraire) ? Comment enrichir (quelles mesures) ?

Entre les bases :

- Pour un phénomène représenté dans une base, existe-t-il une représentation dans l'autre base (des éléments communs) ?
- Quelles différences existe-t-il entre les bases (en termes de modélisation, de structuration, de niveau de détail, de qualité, …) ?
- Quelles sont les représentations potentiellement équivalentes ? Est-il possible de définir *a priori* les correspondances équivalentes ?
- Les règles de correspondances sont-elles formalisables ?
- Quelles sont les propriétés à comparer ? Quelles mesures utiliser ? Faut-il caractériser différemment la géométrie d'un des deux jeux de données ?
- Comment apparier les données ? Quels outils utiliser ?

Toutes ces questions doivent donc trouver une réponse lors de l'analyse des documents par l'expert.

RECUEIL DES CONNAISSANCES POUR L'ENRICHISSEMENT ET L'APPARIEMENT

Il est possible de donner quelques indications sur la démarche à suivre pour analyser les documents et recueillir l'information nécessaire aux étapes d'enrichissement et d'appariement de *MECO* (cf. figure 88).

Pour l'enrichissement, une analyse individuelle des documents doit d'abord être menée. Il faut identifier la ou les classes dans lequel est défini le phénomène à étudier. Il faut également repérer s'il n'existe pas d'informations sur ces classes dans d'autres parties du document. En principe, les spécifications sont structurées selon les classes définies dans la base. A chaque classe correspond une fiche de spécifications qui lui est propre. En pratique, il est fréquent de retrouver aussi quelques informations sur une classe dans d'autres classes de la base. Il faut donc généralement parcourir l'ensemble du document pour collecter toutes les règles de saisie des objets concernant une seule classe.

Après avoir identifié les informations relatives au phénomène, il faut prendre connaissance des règles de saisie qui le concerne et déterminer ce qui est contrôlable dans les données. En identifiant les spécifications vérifiables (ici, celles qui ne nécessitent pas la mise en correspondance des données), on peut déduire les éléments et les propriétés à extraire des données, ceux qui manquent pour réaliser les contrôles. Il en découle une spécification des outils d'analyse spatiale à développer ou à se procurer.

En plus de l'analyse individuelle des spécifications, l'expert doit mener une analyse croisée des documents. Celle-ci est destinée à établir si les données d'une des bases doivent être enrichies pour contrôler les données de l'autre base (l'enrichissement étant une étape de préparation aux contrôles intra-base et inter-bases). Il faut donc cette fois identifier les spécifications qui peuvent être vérifiées après la mise en correspondance des données et déterminer les objets et les attributs à extraire qui n'existent dans la base qu'à travers la géométrie. Une spécification des outils d'enrichissement à développer doit être défini à la fin de cette étape.

Concernant l'appariement, l'expert peut analyser les correspondances définies entre les schémas pour identifier les objets des deux bases à relier. Ensuite, en fonction des différentes modélisations possibles des objets et de leur mode

d'implantation, il doit définir une spécification des outils d'appariement géométrique à utiliser.

RECUEIL ET REPRESENTATION DES CONNAISSANCES POUR LES CONTROLES INTRA-BASE ET INTER-BASES

L'analyse des spécifications est aussi destinée à recueillir les connaissances utiles aux étapes de contrôles intra-base et inter-bases. L'expert doit identifier toutes les contraintes de saisie se rapportant au phénomène étudié pour les deux BDG et reformuler ces contraintes sous forme de règles de production dans le langage manipulable par le système-expert. La démarche à suivre est illustrée en figure 71.

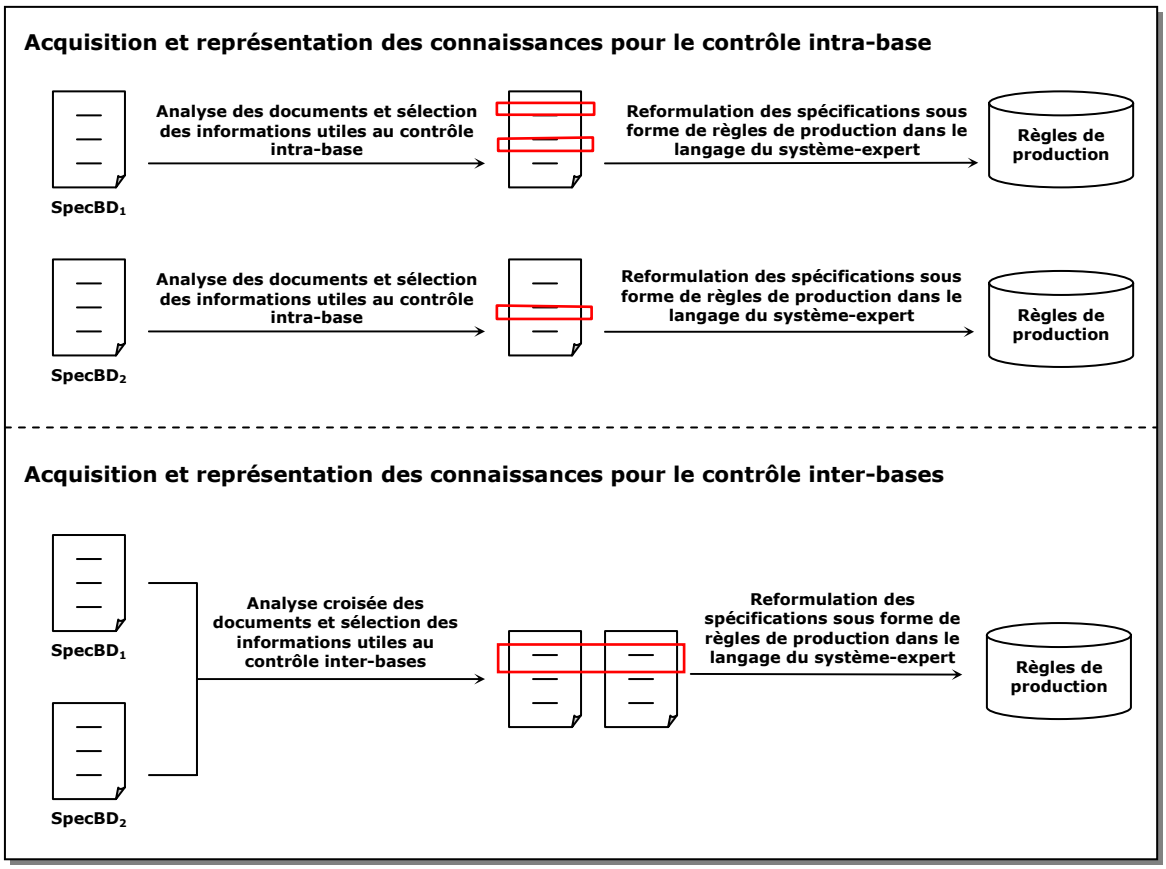


***Figure 71**. Démarche d'acquisition et de représentation des connaissances pour les contrôles intra-base et inter-bases.*

Pour le contrôle intra-base, seule une analyse individuelle des documents est nécessaire. Après avoir sélectionné dans les documents toutes les contraintes de saisie à vérifier décrites en langue naturelle, l'expert doit les représenter sous forme de règles de production pour les introduire dans le système-expert. Deux bases de règles sont produites au terme de l'analyse, une base pour chaque BD.

Pour le contrôle inter-bases, les documents sont analysés conjointement car cette fois, il s'agit de définir les règles permettant de classer les différences comme des équivalences ou des incohérences. Une seule base de règles est définie. Ces règles s'appliquent sur les couples d'objets appariés. Les règles peuvent être organisées de deux manières différentes : en adoptant l'approche par *classification directe* ou en adoptant l'approche par *prédiction, comparaison et classification* (cf. C.5.4).

BILAN

L'analyse des spécifications est une tâche qui n'est pas automatisée aujourd'hui. L'analyse est interactive et doit être réalisée par l'expert. La représentation actuelle des spécifications peut rendre le travail de comparaison difficile car les documents n'ont pas nécessairement la même structure et les règles de saisie peuvent être formulées différemment. Pour faciliter l'étude et la comparaison des documents, nous avons élaboré un modèle permettant de normaliser leur description. Nous le présentons ci-dessous.

D.3.2 FORMALISATION DES SPECIFICATIONS

La construction d'un modèle de spécifications relatives aux BDG est un problème de représentation de connaissances. Nous avons vu que les spécifications des BDG, décrites en langue naturelle, pouvaient manquer de structuration et présentaient des imprécisions. Pour que ces spécifications soient plus homogènes, plus facilement comparables et traitables par une machine, nous avons cherché à les représenter de manière plus formelle.

Précisons que les sens de *spécification* et *formell*e que nous utilisons ici sont différents de ceux utilisés dans le cadre du développement de logiciels [Fougères et Trigano 1999]. Pour ce contexte, une spécification formelle est « *l'expression, dans un langage formel et à un certain niveau d'abstraction, d'une série de propriétés qu'un système devrait satisfaire* » [Van Lamsweerde 2000]. En ce qui nous concerne, les propriétés décrites dans les spécifications doivent être respectées par le contenu du système (les objets dans la BDG) mais pas par le système lui-même. Notre objectif est de définir une meilleure structuration de documents. Nous souhaitons à terme décrire les informations exprimées en texte libre dans un langage manipulable par une machine et les associer aux schémas conceptuels des bases de données. On se rapproche davantage du problème posé par [Zweigenbaum 1999] concernant la représentation de l'information médicale : quel modèle adopter pour pourvoir traiter l'information avec une machine ? Le langage que nous avons adopté pour présenter le modèle est UML[23]. Nous n'exprimons pas les spécifications en adoptant une notation formelle tel que Z [Lightfoot 2001], ce qui est préconisé dans les méthodes de développement de logiciels.

La démarche adoptée pour élaborer ce modèle est une démarche dirigée par les ressources (les spécifications). Nous avons étudié un ensemble de documents provenant essentiellement de l'IGN et identifié progressivement des concepts communs.

Cette contribution est le fruit d'un travail commun. Le modèle a été défini en collaboration avec Nils Gesbert et Sébastien Mustière [Mustière et al. 2003], initié par Anne Ruas. Ce travail a été poursuivi par Nils Gesbert dans le cadre de sa thèse [Gesbert et al. 2004]. Nous y reviendrons par la suite.

[23] Nous supposons que ce modèle est connu (les concepts relatifs aux diagrammes de classes). On peut trouver une description détaillée dans [Muller 1997] si nécessaire.

D.3.2.1 METAMODELE

Le modèle que nous proposons s'appuie sur trois concepts principaux du métamodèle relatif aux BD géographiques[24] : les métaclasses ***Classe***, ***Attribut*** et ***Association*** (figure 72). Les classes présentes dans les BDG (route, occupation du sol, bâtiment,…) sont donc considérées ici comme des instances de la métaclasse ***Classe***. Les attributs s'y rapportant sont des instances de la métaclasse ***Attribut***, les relations étant modélisées par la métaclasse ***Association***. Les classes permettant de représenter les spécifications se greffent sur ces concepts.

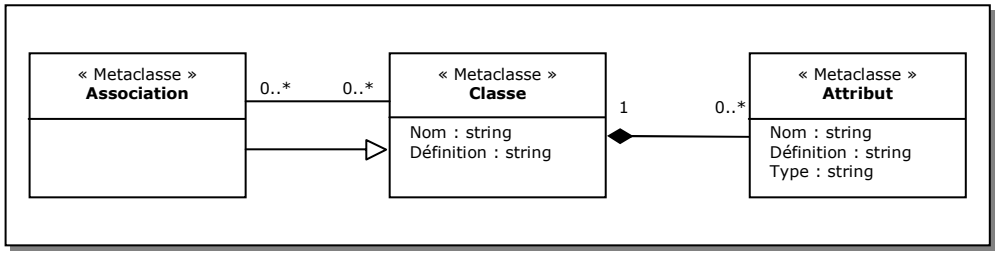


Figure 72*. Métaclasses de la BD géographique sur lesquelles s'appuie le modèle des spécifications.*

D.3.2.2 CONTRAINTES GENERALES

Les critères de saisie et de modélisation mentionnés dans les spécifications peuvent être considérés comme des *contraintes* (figure 73). Ces contraintes portent sur les objets du monde réel et non sur les objets présents dans la base. Une règle peut, par exemple, préciser qu'une route est saisie si elle est revêtue ou si sa longueur sur le terrain est supérieure à 50 m. Toutes ces conditions de représentation vont être exprimées à partir de la classe générale ***Contrainte*** qui se spécialise en deux classes filles : la classe relative aux contraintes simples (***Contrainte simple***), et la classe relative aux contraintes complexes récursives (***Contrainte Complexe***).

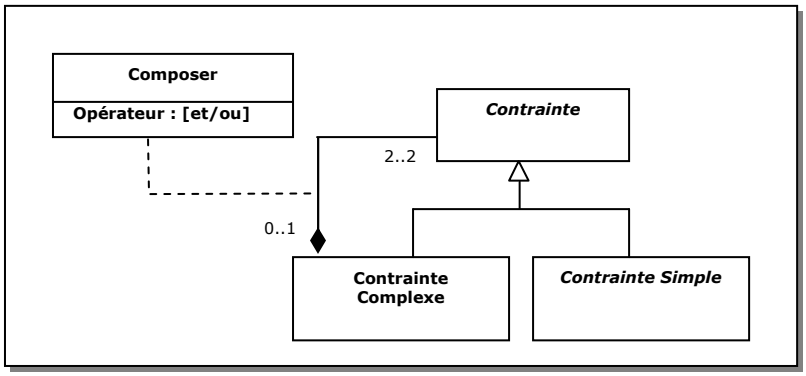


Figure 73*. Extrait du modèle de spécifications : contraintes générales.*

Les contraintes complexes permettent d'exprimer des contraintes composées de plusieurs conditions de représentation tel que « *les rivières sont saisies si elles sont permanentes et si leur largeur est supérieure à 10m* ». La condition de permanence constitue une première contrainte simple, la condition relative à la largeur en constitue

[24] Le métamodèle auquel nous faisons référence est celui décrit dans le document de l'ISO/TC211/WG 19109, *Rules for application Schema - General Feature Model.* Pour plus de clarté, les métaclasses ont été renommées.

une autre, les deux formant une contrainte complexe et étant reliées par l'opérateur logique '*et*'. Avec cette modélisation d'agrégation récursive, on obtient ainsi une structure arborescente. Les nœuds de cet arbre sont les opérateurs logiques '*et*', '*ou*', et les feuilles sont des contraintes simples. Les nœuds sont représentés par des instances de la classe ***Contrainte Complexe*** et leurs fils sont donnés par la relation '*est composé de'*. Les contraintes simples se spécialisent en trois catégories de contraintes élémentaires : les *contraintes géométriques*, les *contraintes de nature* et les *contraintes de relation*. Elles seront détaillées par la suite (D.3.2.4.).

D.3.2.3 REPRESENTATION DU « QUOI » ET DU « COMMENT »

Cette hiérarchie de contraintes sert à représenter à la fois les conditions d'existence des objets dans la base, les conditions relatives à leur modélisation, et les conditions se rapportant aux valeurs des attributs (figure 74).

CONTRAINTES D'EXISTENCE

Les *contraintes d'existence* ou *de sélection* précisent les conditions que doivent satisfaire les entités du monde réel pour qu'elles soient saisies dans la base (le *« quoi »*). Il peut s'agir de conditions géométriques (« *la superficie doit être supérieure à 50 m² »*), relationnelles (« *seuls les chemins menant à une maison sont saisis »*) ou de nature (« *on ne retient que les bâtiments en pierres du pays »*) (cf. D.3.2.4.). Les *contraintes d'existence* sont reliées aux classes du schéma de la BD grâce à l'association '*doit être instanciée ssi l'entité du monde réel correspondante vérifie*' (figure 74). Un peut trouver un exemple d'instanciation du modèle en figure 76.

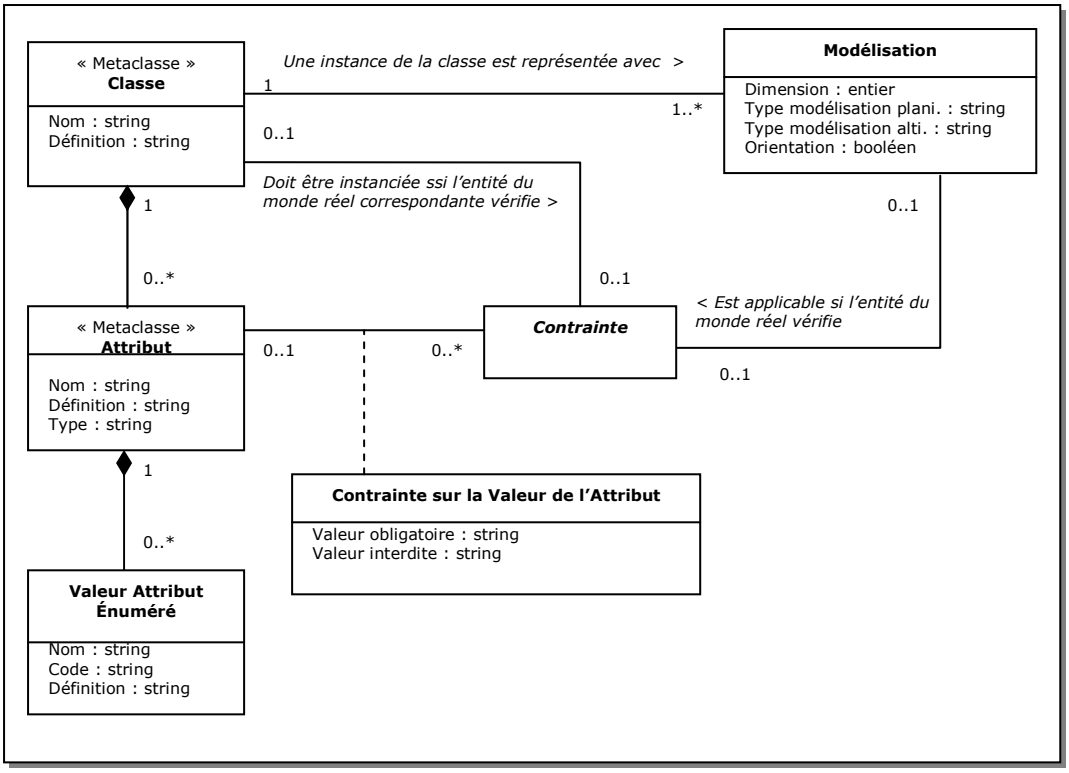


Figure 74*. Extrait du modèle de spécifications : contrainte de modélisation, contrainte d'existence et contrainte relative à la valeur des attributs.*

CONTRAINTES DE MODELISATION

Pour exprimer la modélisation que doivent respecter les objets dans la base (le « *comment* »), une classe ***Modélisation*** a été définie. On indique d'une part la dimension de l'objet (0 = point, 1 = ligne, 2 = surface) grâce à l'attribut correspondant. On mentionne d'autre part la façon dont on obtient sa géométrie à partir du monde réel, à l'aide des attributs '*type de modélisation planimétrique*' et '*type de modélisation altimétrique'*. Ces attributs précisent ce qu'on saisit. En planimétrie, il s'agit le plus souvent de l'axe de l'élément (« *on saisit l'axe de la route* »), son centre ou son pourtour extérieur ou intérieur (voire les deux). En altimétrie, la saisie se fait généralement au sommet des objets (pourtour), parfois à leur base (« *on saisit le bâtiment au sol* »). Si l'objet a plusieurs modes d'implantation (« *une rivière est représentée par une ligne si elle fait moins de 20 m de large sinon, elle est représentée par une surface* »), on crée plusieurs instances de la classe ***Modélisation***. Les contraintes se rapportant à cette modélisation sont exprimées grâce à l'association '*est applicable si l'entité du monde réel vérifie*' (figure 74).

CONTRAINTES SUR LES ATTRIBUTS

Les attributs peuvent également être contraints (figure 74). Outre le fait qu'ils doivent respecter un domaine de valeurs, on spécifie dans certaines circonstances une valeur obligatoire ou une valeur interdite. C'est le cas par exemple de la condition : *« l'attribut 'état physique' de la route, porte toujours pour les pistes cyclables, la valeur 'route revêtue'* ». Une instance de la classe-association ***Contrainte sur la valeur de l'attribut*** sera créée, reliant l'attribut '*état physique*' et la contrainte de nature '*être une piste cyclable*' (contrainte élémentaire), en rendant obligatoire la valeur '*route revêtue*'. Un seul des deux attributs de cette classe-association sera rempli (l'attribut '*valeur obligatoire'*).

L'attribut '*type*' de la classe ***Attribut*** permet de spécifier si les valeurs sont des chaînes de caractères, des entiers, des réels, etc. Lorsque les valeurs sont énumérées (choix parmi une liste prédéfinie), une instance de la classe ***Valeur Attribut Enuméré*** est créée.

Ces différentes classes et relations permettent donc de préciser ce qu'il faut représenter dans la base et comment le représenter. Elles indiquent si les contraintes se rapportent à l'existence, à la représentation, ou aux attributs des objets. Les contraintes élémentaires auxquelles elles sont associées peuvent être de plusieurs types : *contrainte géométrique*, *contrainte relationnelle*, *contrainte de nature*. Nous les présentons ci-dessous.

D.3.2.4 REPRESENTATION DES CONTRAINTES ELEMENTAIRES

Une contrainte élémentaire est une contrainte simple spécialisée (figure 75). Elle peut concerner la géométrie de l'objet du monde réel, sa nature, ou les relations qu'il entretient avec d'autres objets (métrique et topologique notamment).

Les ***Contraintes Géométriques*** permettent de définir un critère géométrique en l'associant à un seuil et à un opérateur de comparaison (inférieur, supérieur, égal, différent,…). Elles sont fréquemment utilisées dans le cadre des contraintes d'existence. Les critères géométriques les plus souvent rencontrés portent sur la *longueur*, la *largeur*, la *superficie*, la *plus grande dimension,* la *plus petite dimension*, le *diamètre* et la *hauteur* d'un objet. L'attribut '*mesurable*' spécifie si le critère géométrique peut être vérifié dans la base, et ainsi être vu comme une contrainte

d'intégrité, ce qui n'est pas toujours le cas (une route représentée par un objet linéaire n'a pas de largeur dans la base par exemple).

> *Cet attribut est particulièrement utile dans notre contexte puisqu'il permet directement d'identifier les spécifications vérifiables avant la mise en correspondance des différentes sources de données (lors du contrôle intra-base) des spécifications qui ne peuvent être contrôlées qu'après la mise en correspondance des données (contrôle inter-bases).*

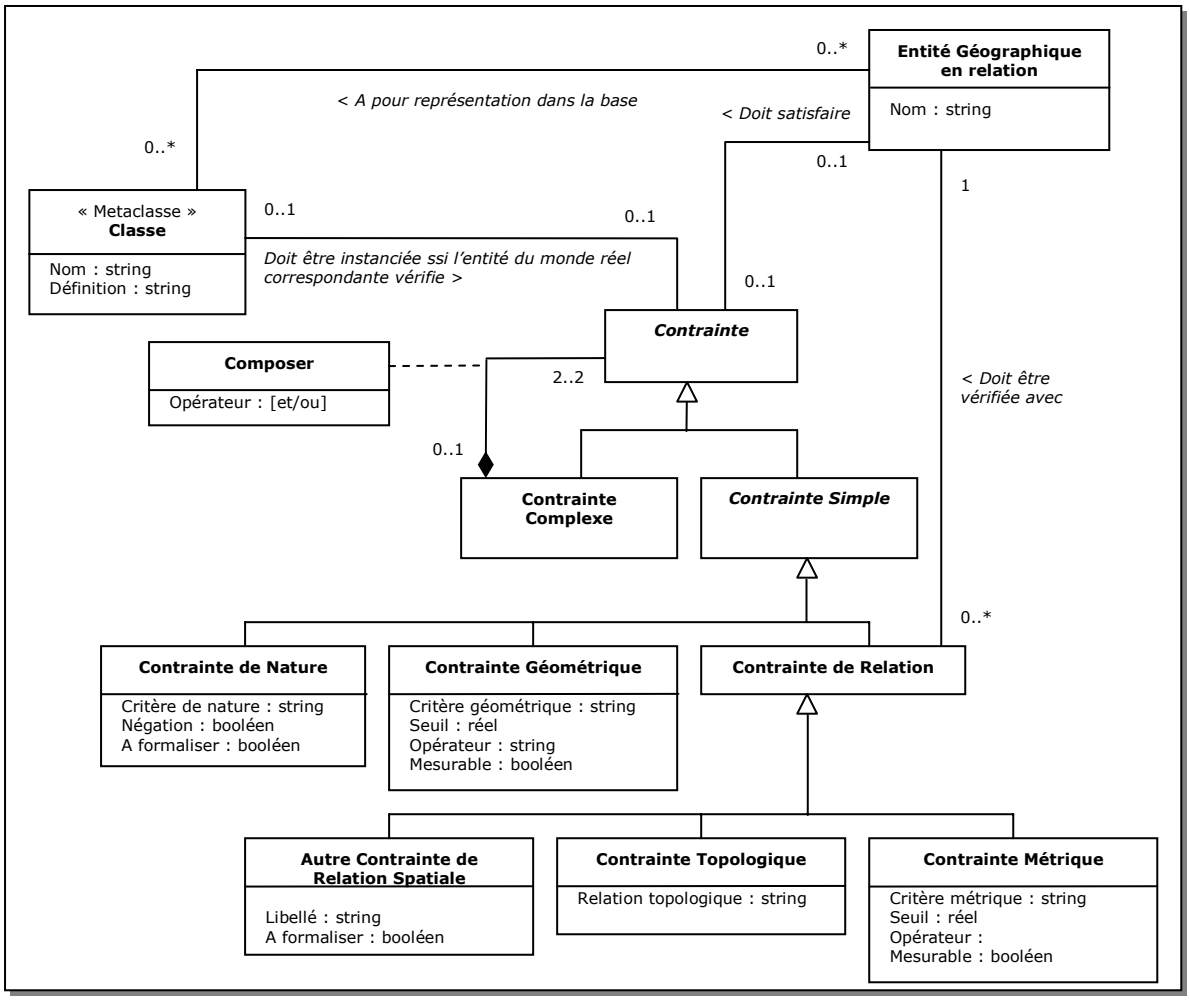


Figure 75. *Extrait du modèle des spécifications : les contraintes élémentaires géométriques, de relation et de nature.*

Les ***Contraintes de nature*** précisent la nature de l'entité. On indique par exemple que les rivières « *doivent être permanentes* » ou que les pylônes électriques « *doivent être en béton* ». La nature est parfois imprécise ou difficilement formalisable (« *une perte est saisie si le trou est important* » ; « *une pêcherie est un lieu aménagé pour une entreprise de pêche : sont concernées les cultures de palourdes, les cultures de coques, les cultures d'huîtres et les autres cultures…* »). En plus du critère de nature, deux attributs sont définis pour ces contraintes : l'attribut '*négation*' qui permet par exemple de spécifier qu'une maison « *doit être en pierre* » ou « *ne doit pas être en bois* », et l'attribut '*à formaliser*' qui précise si le critère devrait être exprimé de manière plus précise (cf. section suivante).

Les ***Contraintes de relation*** permettent de définir les exigences portant sur l'environnement d'une entité. On distingue les ***Contraintes de Relation Métrique***

(« *l'objet n'est saisi que s'il est à moins de 20 m du bord de la route* »), les ***Contraintes de Relation Topologique*** (« *l'objet doit être hors d'une zone inondable* ») et les ***Autres Contraintes de Relation Spatiale*** (« *on saisit le sentier s'il mène à une maison* » ou « *on représente la chapelle si elle est proche du village* »). Les relations topologiques peuvent être définies en utilisant par exemple les notations du formalisme CONGOO [Pantazis et Donnay 1996] ou du modèle des 9-intersections [Egenhofer et Franzosa 1991]. La classe ***Entité Géographique en Relation*** permet de préciser l'objet impliqué dans la relation sur lequel des contraintes peuvent également être fixées. Cette entité n'existe pas nécessairement dans la base et il peut s'agir d'un groupe d'objets formant une entité ayant un sens géographique (« *une piste cyclable doit être en milieu urbain* » ou « *l'arbre doit faire partie d'un alignement* »). Quand l'entité géographique existe dans la base, elle est reliée à la classe correspondante grâce à l'association *'a pour représentation dans la base'*.

D.3.2.5 ÉVALUATION DU MODELE ET PREMIERES IMPLEMENTATIONS

Après avoir défini ce modèle nous avons naturellement cherché à vérifier son pouvoir expressif. Dans cette optique, nous avons décidé de décrire l'ensemble des spécifications du thème hydrographie de la BDTopo standard de l'IGN (qui comprend plus d'une trentaine de classes), en plus de quelques autres classes dont les contraintes semblaient particulièrement difficiles à formaliser. Ce modèle a également été éprouvé dans le cadre d'un stage effectué à l'IGN portant sur l'intégration [Cleach et Fort 2003].

Nous illustrons la description de quelques contraintes relatives à la classe « Mur » de la BDTopo standard selon ce modèle en figure 76.

Le diagramme d'objets UML représenté met en évidence la structure arborescente en présence de contraintes complexes. Les critères repris dans cet exemple sont particulièrement simples et faciles à introduire dans le modèle mais il supporte également l'expression de contraintes beaucoup plus floues qui nécessitent d'ailleurs parfois la création d'objets mésoscopiques [Ruas 1999]. C'est le cas de la spécification suivante : « *si les sentiers sont trop nombreux, seul le principal est retenu* ». Il existe dans cette spécification une contrainte de nature « *être principal* » qui conditionne l'existence de l'objet dans la base. Cette contrainte doit être associée à une contrainte de relation. Cette contrainte de relation est reliée à une entité géographique « groupe de sentiers » qui porte lui-même une contrainte sur le nombre de sentiers. Ce nombre peut être exprimé de deux manières différentes. Soit on considère qu'il s'agit d'une contrainte de nature et on précise sous forme textuelle qu'il doit exister de nombreux sentiers. Soit on analyse les données dans l'objectif de fixer un seuil de densité au-delà duquel on saisit un objet représentatif d'un groupe (le principal).

> *Cette alternative souligne l'intérêt de travailler sur les données et l'importance de formaliser les spécifications. En modélisant les spécifications, on met en évidence les contraintes imprécises qui nécessitent une étude des instances géométriques pour les expliciter. La portée du modèle ne se limite donc pas à la structuration des connaissances. C'est véritablement une aide à leur acquisition et leur enrichissement.*

Cette constatation est le principal enseignement que l'on a pu tirer de l'application du modèle. Nous parvenons généralement à exprimer des contraintes assez vagues à l'aide des classes ***Contraintes de nature*** et ***Autres Contraintes de Relation Spatiale*** (lorsque la description fait référence à des objets en relation), mais ces contraintes pourraient être mieux formalisées et exprimées dans les autres classes de contraintes définies en analysant les données. Le fait d'« *être isolé* » par exemple peut s'exprimer

par une *autre contrainte de relation spatiale* mais pourrait être mieux précisé par une *contrainte de relation métrique* (distance) en apprenant un seuil traduisant l'éloignement de l'objet le plus proche. De manière analogue, une contrainte spécifiant qu'il faut « *être de petite taille* » se traduit par une *contrainte de nature* mais une étude des données permettrait de transformer cette condition en *contrainte géométrique* (en fixant un seuil de superficie). C'est encore le cas de notions comme « *être non parallèle* » (une condition à exprimer sous forme de *contrainte de relation métrique* à partir d'une mesure traduisant le parallélisme), « *être le long de* », « *faire partie d'une succession* » ou « *être juxtaposé* » (relation à exprimer en terme de distance et nécessitant la création d'un groupe comme entité géographique en relation). Pour formaliser ces contraintes, les outils d'analyse spatiale sont très utiles (cf. C.2.3.).

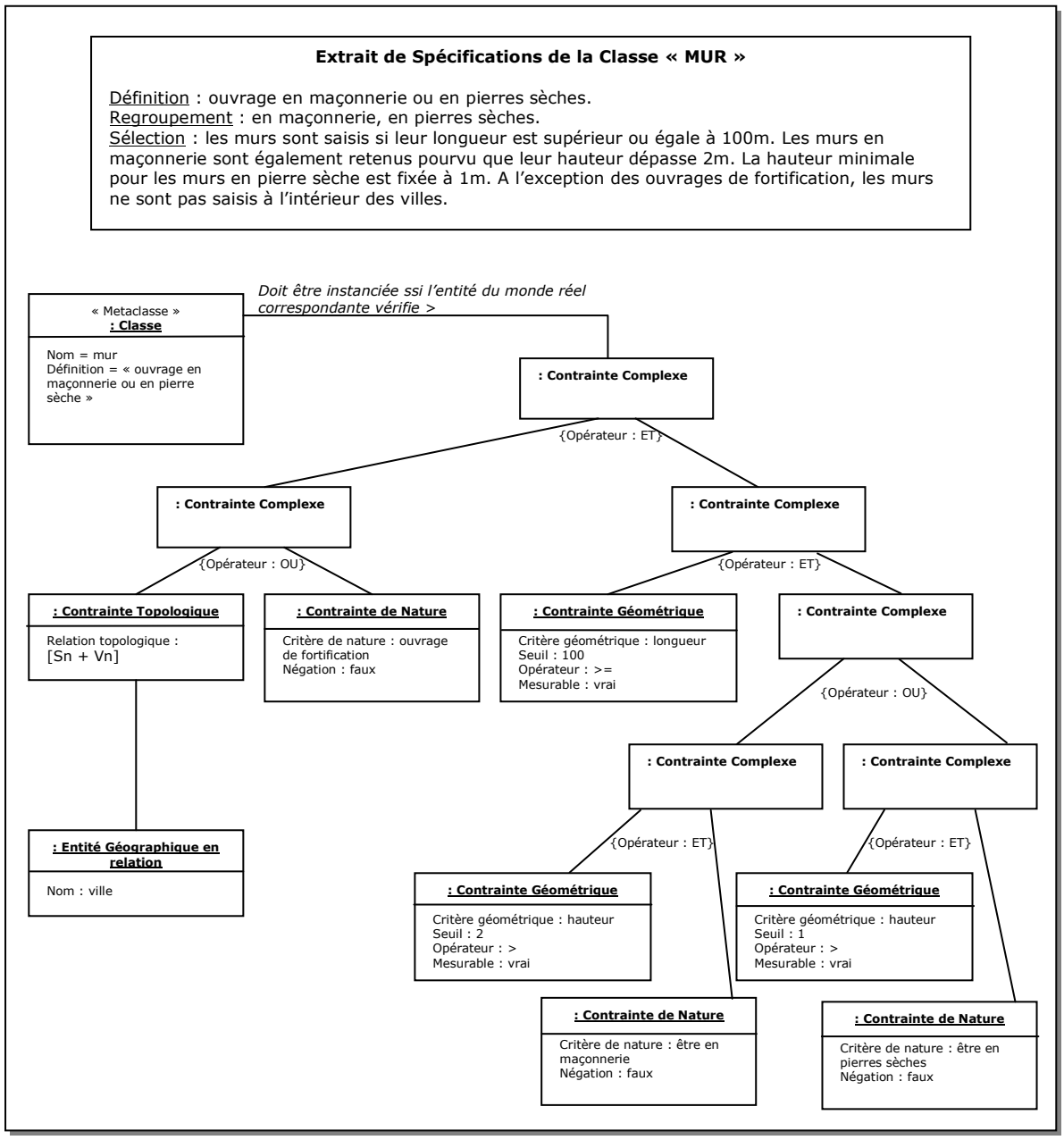


Figure 76. *Diagramme d'objet UML décrivant un extrait des spécifications de la classe « Mur » de la BDTopo standard selon le modèle des spécifications défini.*

D.3.2.6 BILAN ET PERSPECTIVES

Notre implication dans ce modèle s'est volontairement arrêtée à la version présentée. Le travail de formalisation des spécifications a été poursuivi par Nils Gesbert dans le cadre de sa thèse. Suite à ses travaux, quelques extensions ont été apportées au modèle [Gesbert et al. 2004]. Celui-ci propose une description des spécifications sur la base de concepts « partageables », sans établir de correspondance directe entre la structure du schéma conceptuel des BD et le modèle des spécifications (figure 77). Une classe « Entité Géographique » est ainsi définie pour représenter un élément du terrain *conceptualisé,* autrement dit, un élément d'une ontologie. Chaque classe du schéma conceptuel de la base (« Objet dans la base ») est maintenant reliée à cette ontologie et les différentes contraintes s'expriment à partir de cette relation (la classe « Contrainte » de la figure 77 fait référence à la classe « Contrainte » de la figure 75).

Cette modélisation offre plusieurs avantages. En termes d'exploitation pour l'intégration, elle facilite la comparaison des spécifications car la même ontologie sera définie pour les différentes sources. Les concepts communs seront donc directement perçus. Elle met également en évidence la représentation multiple des phénomènes du monde réel puisque chaque représentation donne lieu à une relation avec l'ontologie. En terme de structuration, cette modélisation évite le mélange des notions spécifiques à la base avec celles caractéristiques du monde réel.

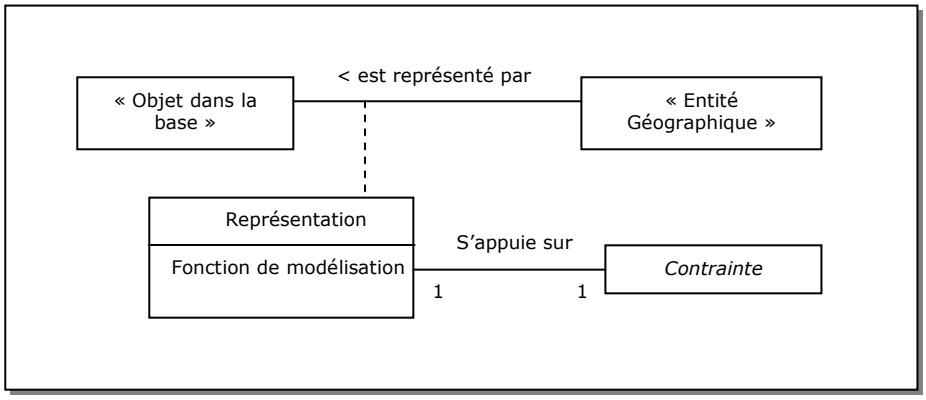


Figure 77. *Extrait du profil UML (extension du métamodèle UML) permettant de représenter les métaéléments émergeant des spécifications (Source : [Gesbert et al. 2004])*

Actuellement, le modèle des spécifications que nous proposons est présenté selon la notation UML. Ce langage nous paraît bien adapté, les langages objets étant d'ailleurs considérés comme de bons formalismes de représentation, d'un bon niveau d'abstraction pour modéliser un domaine [Lépy 1997]. Le choix du langage pour stocker et manipuler les spécifications n'a quant à lui pas encore été fixé[25]. Nous avons mené quelques tests d'implémentation mais à titre exploratoire. Nous avons ainsi étudié les possibilités de stockage des spécifications dans une base de données relationnelle mais cette solution s'est rapidement révélée inadaptée en raison du nombre important de jointures nécessaires pour reconstituer une contrainte ou pour rechercher toutes celles relatives à une classe particulière. Cette structure est d'autant plus inadéquate que la faible quantité d'enregistrements présents dans les tables est

[25] Dans cette thèse, ce choix a été fait puisqu'une partie des spécifications est décrite sous forme de règles de production dans un système-expert. Nous faisons référence ici au langage qui devrait être adopté par l'IGN pour représenter toutes les spécifications des bases sous une forme numérique.

faible (un enregistrement par classe d'objet géographique). Nous avons également examiné les possibilités d'utiliser XML, langage semi-structuré fondé sur la notion de marqueurs et offrant la possibilité de définir un modèle de document (DTD : « *Document Type Definition* ») jouant le rôle de schéma [Michard 1998]. Ce langage fut très facile à mettre en œuvre, le passage du modèle objet UML à XML étant pratiquement immédiat [Euzenat 1999b, Sardet 1999]. Après avoir défini une DTD, nous avons ainsi instancié le schéma pour différentes classes relatives au thème hydrographie de la BDTopo. Nous donnons un extrait de fichier XML en figure 78. D'autres exemples sont donnés en annexe 1.

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<!--  Nom du fichier : Chateau_Eau.xml   -->
<!DOCTYPE Specifications (View Source for full doctype...)>
- <CLASSE>
    <NOM>"Château d'eau"</NOM>
    <DEFINITION>"Grand réservoir construit en élévation, destiné à l'alimentation en eau d'une
      collectivité"</DEFINITION>
  - <ATTRIBUT>
      <NOM>"Toponyme"</NOM>
    - <TYPE_VALEUR>
        <STRING />
      </TYPE_VALEUR>
    </ATTRIBUT>
  - <MODELISE>
    - <MODELISATION>
        <DIMENSION>"2"</DIMENSION>
        <TYPE_MODELISATION_XY>"pourtour"</TYPE_MODELISATION_XY>
        <TYPE_MODELISATION_Z>"sommet"</TYPE_MODELISATION_Z>
      </MODELISATION>
    </MODELISE>
  - <CONTRAINTE_EXISTENCE>
    - <CONTRAINTE_COMPLEXE>
        <TYPE_DE_LIEN>"OU"</TYPE_DE_LIEN>
      - <CONTRAINTE_COMPOSANTE>
        - <CONTRAINTE_SIMPLE>
          - <CONTRAINTE_DE_NATURE>
              <CRITERE_NATURE>"destiné à l'alimentation d'une collectivité"</CRITERE_NATURE>
              <NIER>"Non"</NIER>
            </CONTRAINTE_DE_NATURE>
          </CONTRAINTE_SIMPLE>
        </CONTRAINTE_COMPOSANTE>
      - <CONTRAINTE_COMPOSANTE>
```

Figure 78. *Extrait d'un fichier XML relatif à la classe « Château d'eau » de la BDTopo et décrit selon le modèle des spécifications.*

Si ce modèle est mieux adapté à la représentation des spécifications et leur consultation, il est sans doute encore trop informel. Le fait de pouvoir intégrer des connaissances formalisées avec des mentions textuelles constitue un avantage mais aussi une limite : on risque de ne pas assez structurer les spécifications. Nous pensons qu'il reste encore un travail à mener sur la formalisation des spécifications (raffinement de la classification de certaines contraintes) et que le choix du langage de manipulation reste à déterminer.

S'il est nécessaire de réaliser une étude plus approfondie pour envisager une véritable utilisation automatique des spécifications, ce modèle constitue néanmoins une bonne base pour mieux comprendre ces spécifications. En pratique, ce modèle s'est avéré être un bon outil d'analyse des spécifications.

D.4 ACQUISITION DE CONNAISSANCES ISSUES DES DONNEES PAR APPRENTISSAGE AUTOMATIQUE SUPERVISE

D.4.1 APPRENTISSAGE

Le modèle que nous venons d'exposer est un outil d'aide à l'acquisition des connaissances pour l'étape d'analyse des spécifications. La seconde étape proposée dans *MACO*, qui est optionnelle, est l'apprentissage automatique (cf. figure 87). Les techniques d'apprentissage peuvent aider à faire émerger des connaissances implicites à partir des données. En proposant l'apprentissage pour recueillir des connaissances, nous adoptant la seconde approche proposée en acquisition des connaissances.

D.4.1.1 METHODES D'APPRENTISSAGE AUTOMATIQUE

L'apprentissage désigne toute méthode permettant de construire un modèle de la réalité à partir de données, soit en améliorant un modèle partiel ou moins général, soit en créant complètement le modèle [Cornuéjols et Miclet 2002].

En intelligence artificielle, les techniques d'apprentissage sont particulièrement nombreuses. L'apprentissage *par induction* que nous exploitons ici est une des méthodes les plus étudiées dans le domaine. L'objectif de cette approche est de construire de manière empirique un modèle *général* d'une réalité, à partir de connaissances *particulières*. Elle se distingue de l'approche *par analogie* qui exploite les connaissances d'une tâche bien connue pour déterminer des solutions destinées à résoudre une tâche moins connue, en procédant par une analyse de similarités : il s'agit du raisonnement à partir de cas notamment. Elle diffère également de l'*approche déductive* qui s'attache à inférer de nouvelles connaissances en étudiant le lien existant entre un cas particulier et le concept qu'il relie : il s'agit de l'apprentissage fondé sur l'explication (EBL : « *Explanation-Based Learning* »).

Dans l'apprentissage inductif, on distingue l'*apprentissage non supervisé* de l'*apprentissage supervisé*. Dans les deux cas, l'apprentissage se fait à partir d'un ensemble d'*exemples* mais la forme de ces exemples est différente. Pour l'apprentissage supervisé, les exemples sont constitués de descripteurs (des attributs) et sont *étiquetés* ou *classés*. La classe est fournie par un « oracle » (en général l'expert) et la tâche de l'algorithme d'apprentissage est de découvrir la relation générale existant entre les descripteurs et les étiquettes. La fonction apprise, appelée *hypothèse*, doit permettre de prédire la classe d'un nouvel exemple d'apprentissage. Cette fonction peut prendre la forme d'un arbre de décision, d'un ensemble de règles de production ou d'un réseau de neurones (« boîte noire prédictive »), etc. Pour l'apprentissage non supervisé, les exemples (*observations*) sont dépourvus d'étiquette. Les algorithmes d'apprentissage cherchent à les regrouper sur base de régularités : ils rapprochent les exemples les plus similaires tout en éloignant ceux qui présentent des caractéristiques différentes. On retrouve dans cette famille des méthodes comme les réseaux bayésien, le *clustering* (méthode des *k-moyennes* par exemple), les cartes de kohonen ou les règles d'associations.

Dans l'apprentissage supervisé, on distingue encore deux approches différentes en fonction du langage de représentation dans lequel est exprimée l'hypothèse apprise : l'approche numérique et l'approche symbolique. La première est à mettre en relation avec les réseaux connexionnistes. La fonction de classification est difficilement

interprétable et constitue une « boîte noire ». La seconde fournit une hypothèse plus facilement compréhensible. Il s'agit d'arbres de décisions ou de règles de production. Cette approche symbolique est généralement retenue pour la constitution de systèmes-experts.

> *C'est ce qui explique notre choix d'utiliser l'apprentissage supervisé symbolique. Nous souhaitons apprendre des règles intelligibles, facile à réviser et à introduire dans notre système-expert.*

D.4.1.2 PRINCIPES DE L'APPRENTISSAGE INDUCTIF

DEFINITION

L'apprentissage supervisé symbolique que nous exploitons ici est un processus inductif, nous l'avons déjà précisé. Ce processus consiste donc à déterminer à partir d'un ensemble de connaissances particulières, un modèle générale d'une réalité. De manière plus formelle, l'apprentissage supervisé peut être décrit de la manière suivante (figure 79) :

> *Étant donné un échantillon d'exemples d'apprentissage S constitués d'éléments x_i tirés aléatoirement suivant une distribution D sur l'ensemble des exemples possibles, et d'étiquettes c_i fournies par l'expert qui utilise une fonction inconnue f pour les déterminer, le système d'apprentissage cherche à inférer f ou à en trouver une approximation h (une hypothèse) à partir de cet échantillon d'exemples $\{x_i, c_i\}$.*

Les éléments de l'échantillon x_i correspondent à des descriptions d'objets du monde réel, le plus souvent sous forme d'un vecteur d'attributs. La classe c_i associée à ces éléments est fournie par l'expert du domaine. A partir de ces exemples, le système d'apprentissage cherche à estimer une fonction de classification capable de prédire automatiquement la classe de nouveaux éléments dont l'étiquette est inconnue et qui n'apparaissent pas forcément dans l'échantillon des exemples. Cette prédiction doit générer un minimum d'erreurs de classification sur les nouveaux éléments, c'est-à-dire qu'elle doit refléter au mieux la décision qu'aurait prise un expert.

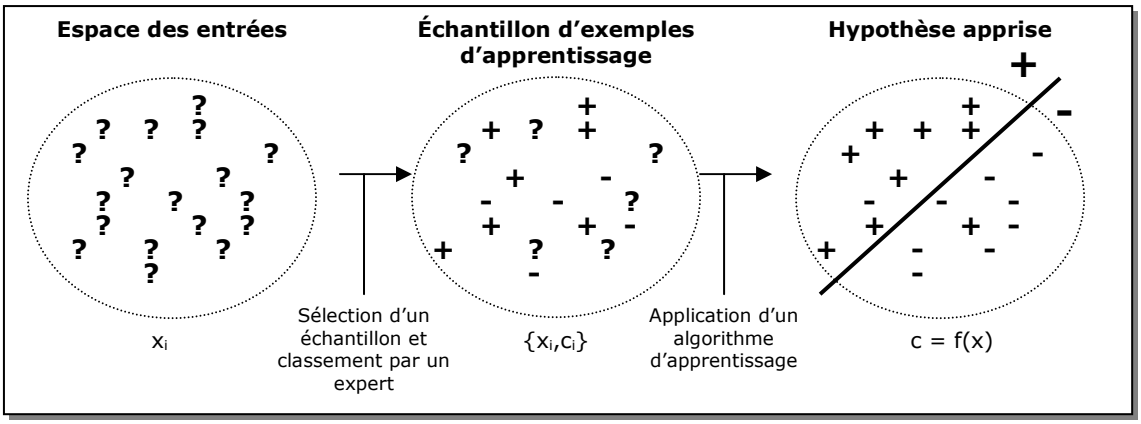


Figure 79. *Mise en œuvre d'un processus d'apprentissage supervisé : classification interactive des observables, obtention automatique d'une fonction de classification.*

Illustrons la tâche d'apprentissage à l'aide d'un exemple. Imaginons que nous souhaitions prédire si un vin est bouchonné ou non, sans le goûter. Pour être capable

d'effectuer cette prédiction par apprentissage, nous devons disposer d'un échantillon de bouteilles pour lesquelles nous savons si le goût du vin est bouchonné (*exemples positifs*) ou non bouchonné (*exemples négatifs*). Le vin doit être décrit par un ensemble d'attributs (type de bouchon, année, origine, appellation, etc.) et nous supposons que ceux-ci sont suffisamment informatifs pour permettre d'apprendre une procédure de classification pertinente. La tâche d'apprentissage consisterait donc à inférer une fonction de classification permettant de déterminer, en fonction de la valeur des attributs, si une nouvelle bouteille de vin qui n'a pas encore été goûtée, est bouchonnée ou non.

La difficulté de l'apprentissage réside dans l'induction [Mitchell 1997, Cornuéjols et Miclet 2002]. En effet, comment passer de cas particuliers à un modèle général assez exact ? La fonction doit être ainsi capable de classer des bouteilles de vin jusqu'alors inconnues, qui n'apparaissent pas nécessairement dans la liste des exemples. On pourrait en effet envisager de mémoriser tous les exemples de l'échantillon d'apprentissage dans une table et de parcourir celle-ci lorsqu'une nouvelle bouteille à classer est présentée. Si la bouteille existe dans la liste, on retournerait la classe correspondante, sinon, on pourrait proposer une classe au hasard. Un système d'apprentissage réalisant une telle procédure de classification ne ferait aucune erreur sur les exemples de l'échantillon. En revanche, son pouvoir inductif serait médiocre. L'apprenant n'aurait aucune capacité de généralisation. Un système d'apprentissage doit permettre de construire une procédure de classification qui soit non seulement correcte sur l'échantillon d'apprentissage mais aussi qui génère par ailleurs un minimum d'erreurs de classification sur de nouveaux exemples.

Les principes qui rendent possible l'induction sont présentés ci-dessous et tirés de [Mitchell 1997, Cornuéjols et Miclet 2002].

BIAIS D'APPRENTISSAGE INDUCTIF

L'échantillon d'apprentissage à lui tout seul ne fournit pas suffisamment d'informations pour réaliser une induction. Sans connaissances supplémentaires, la généralisation est impossible.

Illustrons ce problème à l'aide d'un exemple. Supposons un échantillon d'apprentissage décrit par 3 attributs binaires et dont la valeur de l'étiquette est '+' ou '-' (on parle d'*apprentissage de concepts* dans ce cas). Imaginons que nous cherchions à décrire l'ensemble des partitions possibles de cet échantillon. On peut calculer qu'il existe 2^3 soit 8 formes différentes d'exemples possibles pour cet échantillon (figure 80). On peut également calculer qu'il existe 2^8 manières différentes d'étiqueter les exemples par '+' ou '-' si on suppose que tous les cas sont réalisables *a priori*. Supposons que l'on dispose de 5 exemples étiquetés. Il reste donc 3 formes d'exemple de classe inconnue, donc 8 hypothèses envisageables. Quelle hypothèse choisir ? Avec ces connaissances, il n'est pas possible de faire un choix. Il n'existe aucune raison d'étiqueter les 3 exemples dans une classe plutôt qu'une autre. Pour chaque exemple, il existe quatre hypothèses qui sont associées à la valeur '+' et quatre hypothèses associées à la valeur '-' (figure 80). Il existe donc autant d'hypothèses qui associent les exemples à la classe positive qu'à la classe négative. Une telle procédure ne permet pas de classer des exemples inconnus. En outre, l'énumération de toutes les formes d'exemple et d'étiquettes associées n'est pas une solution réaliste car la taille de l'espace des hypothèses peut rapidement devenir démesurée en rajoutant de nouveaux attributs (le nombre de possibilités tend rapidement vers l'infini). Il n'est donc pas possible d'apprendre en suivant cette stratégie. Dans ce cas, comment induire ?

Échantillon d'exemples d'apprentissage :

ID	Attributs X_1	X_2	X_3	Classe
1	0	0	0	+
2	0	0	1	-
3	0	1	0	+
4	0	1	1	?
5	1	0	0	+
6	1	0	1	?
7	1	1	0	?
8	1	1	1	-

Hypothèses possibles relatives au trois exemples de classe inconnue :

h_1 { (0-1-1) = + ; (1-0-1) = + ; (1-1-0) = + }
h_2 { (0-1-1) = - ; (1-0-1) = + ; (1-1-0) = + }
h_3 { (0-1-1) = + ; (1-0-1) = - ; (1-1-0) = + }
h_4 { (0-1-1) = + ; (1-0-1) = + ; (1-1-0) = - }
h_5 { (0-1-1) = - ; (1-0-1) = - ; (1-1-0) = + }
h_6 { (0-1-1) = + ; (1-0-1) = - ; (1-1-0) = - }
h_7 { (0-1-1) = - ; (1-0-1) = + ; (1-1-0) = - }
h_8 { (0-1-1) = - ; (1-0-1) = - ; (1-1-0) = - }

Figure 80. *La recherche de toutes les partitions possibles d'un espace d'hypothèses ne permet pas l'induction.*

La solution adoptée pour résoudre le problème de l'induction est de réduire la taille de l'espace des hypothèses en introduisant des *biais d'apprentissage*. Un biais est une connaissance qui restreint le champ des hypothèses que l'apprenant doit considérer à un moment donné [Cornuéjols et Miclet 2002]. Le problème revient donc à déterminer, dans un espace d'hypothèses fixé $H=\{h_1,\ldots,h_n\}$, l'hypothèse la plus satisfaisante h*, qui approche au mieux la fonction f. L'espace des hypothèses est limité par le *biais de représentation* et le *biais de restriction*. Le choix de l'hypothèse satisfaisante est dicté par le *biais de préférence* [Mitchell 1997].

Le biais de représentation est introduit en choisissant un langage de description. Ce langage concerne à la fois les exemples et les hypothèses. On décrira ainsi les éléments sous forme d'un ensemble d'attributs/valeurs par exemple. L'hypothèse pourra quant à elle être décrite sous forme d'un arbre de décision ou d'une liste de règles de production. Ce langage peut restreindre l'espace des hypothèses en imposant certaines contraintes par le biais de restriction. On peut par exemple imposer que les hypothèses soient décrites par une conjonction de conditions sur les attributs. La croissance de la taille d'un arbre de décision peut également être limitée. On impose de cette manière à l'apprenant de ne pas explorer l'ensemble de toutes les hypothèses possibles mais de chercher dans un espace plus restreint. La question est alors de savoir comment guider l'exploration dans cet espace fixé et de choisir l'hypothèse la plus satisfaisante ?

C'est à ce niveau qu'intervient le biais de préférence. Il est nécessaire de définir une mesure ou un critère qui permet de comparer les hypothèses et de sélectionner celle qu'on considère comme étant la plus pertinente. On peut ainsi préférer les hypothèses qui couvrent le plus d'exemples ou celles qui sont les plus simples. Pour la construction d'arbres de décision, par exemple, on préfère souvent les arbres de plus petite taille : on suppose qu'ils ont une plus grande capacité de généralisation et ils sont moins complexes. Les biais de préférence permettent donc de choisir parmi l'ensemble des hypothèses potentiellement valides, celle qui satisfait le mieux une condition fixée.

Pour illustrer la mise en pratique de ces biais et montrer leur importance, décrivons le principe de construction d'un arbre de décision. Pour développer un arbre, les algorithmes procèdent de manière descendante. Ils vont chercher à diviser les exemples d'apprentissage de manière récursive, en effectuant des tests sur les attributs, jusqu'à obtenir plusieurs sous-ensembles d'exemples possédant presque

tous la même classe (nœuds purs). Lors de la croissance de cet arbre, plusieurs questions doivent être résolues : il s'agit de décider si un nœud est terminal ou pas ; si c'est la cas, l'algorithme doit affecter une classe à la feuille ; dans le cas contraire, un test sur les attributs doit être sélectionné pour savoir comment développer les sous-arbres. Suivant l'algorithme d'apprentissage, les méthodes de construction vont différer. Elles se distinguent par les biais sélectionnés et la manière dont ceux-ci sont implémentés.

Considérons à nouveau l'échantillon d'apprentissage représenté en figure 81 et appliquons la procédure de construction d'un arbre. Au départ, l'arbre est initialisé : il est vide. A la racine de cet arbre, l'échantillon pourrait être caractérisé par le couple (2,3), étant donné que 2 exemples sont négatifs et 3 exemples sont positifs. Est-il possible de trouver une partition plus homogène de l'échantillon, autrement dit, ce nœud racine est-il terminal ? Si nous avions à la racine un couple caractérisé par les valeurs (0,5), une feuille serait créée. Dans notre cas, si on envisage de poursuivre sa croissance, trois choix de développement sont possibles (figure 81). Soit on développe l'arbre en sélectionnant l'attribut X_1 qui décomposerait l'échantillon en deux branches, suivant les valeurs 0 et 1, lesquelles contiendraient respectivement un exemple négatif et deux exemples positifs (branche de valeur 0) et un exemple négatif et positif (branche de valeur 1). Soit on sélectionne un des deux autres attributs X_2 et X_3.

Échantillon d'exemples d'apprentissage :

ID	Attributs X_1	X_2	X_3	Classe
1	0	0	0	+
2	0	0	1	-
3	0	1	0	+
4	0	1	1	?
5	1	0	0	+
6	1	0	1	?
7	1	1	0	?
8	1	1	1	-

Partitions possibles en racine selon les trois attributs :

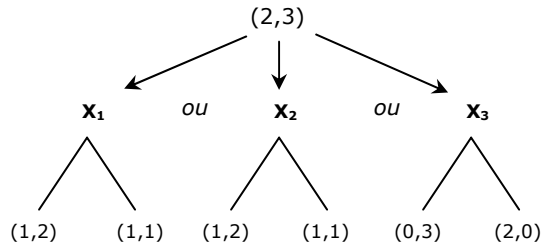


Figure 81. *Suivant le choix de l'attribut, la croissance d'un arbre de décision est différente.*

La sélection de l'attribut X_3 est la solution la plus intéressante. Elle permet d'obtenir une partition plus homogène de l'échantillon puisque que pour les deux branches (celles correspondant à la valeur 0 et 1), tous les éléments sont dans la même classe (respectivement '+' et '-').

Ceci montre que suivant l'attribut sélectionné, l'arbre peut être différent et l'échantillon peut être plus ou moins bien discriminé. Pour automatiser la procédure, il s'agit donc de trouver une mesure pour choisir cet attribut. Les algorithmes d'apprentissage utilisent des fonctions qui permettent de mesurer le degré de mélange des exemples entre les différentes classes. Ainsi, l'algorithme C4.5. développé par [Quinlan 1986, Quinlan 1993] exploite la fonction d'entropie. CART utilise quant à lui l'indice de Gini [Breiman et al. 1984]. Ces fonctions permettent d'évaluer le degré d'homogénéité des classes pour toute position dans l'arbre et aident ainsi à sélectionner le bon attribut. Il suffit de calculer pour chaque attribut le gain d'homogénéité qui serait généré en créant cette nouvelle partition et sélectionner l'attribut qui le maximise.

Le critère d'arrêt de la construction de l'arbre est de ne créer que des feuilles comprenant des exemples appartenant tous à la même classe. On obtient ainsi un arbre assez complexe dont les feuilles correspondent à des échantillons d'exemples purs. Mais un tel arbre n'est pas très intéressant car il est trop spécifique aux données. On poursuit donc le processus par une étape d'élagage des branches qui vise à réduire la complexité de l'arbre et à accroître son pouvoir de généralisation. On attribue alors aux feuilles ainsi retenues les étiquettes correspondant à la classe la plus fréquente.

Cette méthode illustre la mise en œuvre de biais d'apprentissage. Elle montre ainsi que l'espace des hypothèses n'est pas entièrement exploré puisque la construction se fait de manière descendante, en raffinant progressivement la partition de l'échantillon. Ce biais de préférence est également implémenté par la méthode « croître / élaguer », puisque celle-ci a pour effet de simplifier l'hypothèse apprise et donc de privilégier les arbres de petite taille.

Les biais d'apprentissage sont ainsi particulièrement utiles puisque sans eux, l'induction ne serait pas possible. Ils évitent que l'hypothèse apprise ne couvre trop les exemples et que le système ne réalise un *sur-apprentissage*.

SUR-APPRENTISSAGE

La notion de sur-apprentissage (ou « *overfitting* ») fait référence à des hypothèses trop complexes (comme des arbres de décisions trop touffus), dont le modèle coïncide trop avec la base d'exemples utilisés pour apprendre (figure 82). Ce problème peut apparaître si les biais d'apprentissage implémentés dans les algorithmes sont trop restrictifs. On peut imaginer qu'un algorithme d'apprentissage soit programmé pour construire une hypothèse qui privilégie uniquement un très faible taux d'erreurs de classification sur les exemples de l'échantillon (on fait référence ici à l'*erreur apparente*, nous y reviendrons). Dans ce cas, les règles seraient trop spécifiques aux données et leur pouvoir de prédiction serait fortement restreint. A l'extrême, un algorithme d'apprentissage qui fournirait un taux d'erreurs de classification nul sur l'échantillon serait très performant pour cet échantillon mais n'aurait aucune capacité de généralisation. Il apprendrait *par cœur*. Pour les arbres de décision, la phase d'élagage des branches est donc particulièrement importante. Elle permet l'induction en évitant l'*overfitting.*

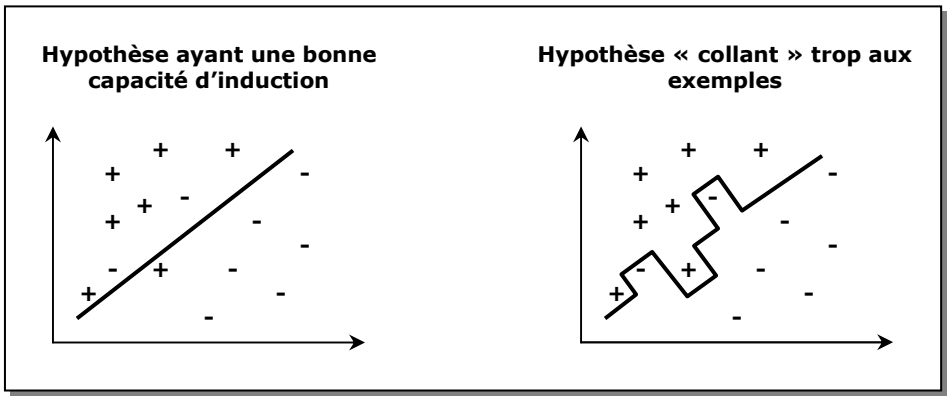


Figure 82. *Sur-apprentissage.*

Le sur-apprentissage se manifeste également en présence d'exemples *bruités*. Un échantillon d'apprentissage contient inévitablement du *bruit*, c'est-à-dire des exemples mal étiquetés, mal décrits ou présentant des valeurs manquantes. Quand la quantité

de bruit est faible, l'algorithme d'apprentissage est capable de ne pas en tenir compte. Sa capacité d'induction lui évite de proposer des règles s'y référant. Par contre, si le nombre d'exemples bruités devient important, l'algorithme d'apprentissage risque de modéliser ce bruit. Cela signifie que les règles apprises vont refléter les exemples erronés. L'hypothèse n'aura donc plus un bon pouvoir prédictif.

> *Ce problème de bruit est particulièrement important dans notre contexte. Si le nombre d'incohérences sur des couples d'objets appariés de même type est élevé, on risque d'apprendre une hypothèse qui tient compte de ces incohérences et donc d'obtenir des règles qui ne reflètent pas les spécifications des bases de données (cf. D.4.2.3).*

C'est la raison pour laquelle il est essentiel d'analyser les règles apprises. Nous avons mené une étude concernant l'influence du bruit sur le *taux d'erreur réelle* lors des expérimentations réalisées sur les différences entre ronds-points. Nous y reviendrons dans le chapitre E.

Cette notion d'*overfitting* nous amène à introduire dans la partie suivante deux critères de qualité relatifs à une hypothèse apprise : le taux d'*erreur apparent* et le taux d'*erreur réelle*.

D.4.1.3 ÉVALUATION DE L'APPRENTISSAGE

ERREUR APPARENTE, ERREUR REELLE

Les méthodes d'apprentissage inductif servent à construire des hypothèses qui génèrent non seulement un faible pourcentage d'erreurs de classification sur les exemples de l'échantillon (taux d'erreur apparente), mais aussi un minimum d'erreurs sur l'ensemble des nouveaux exemples possibles (taux d'erreur réelle).

La sélection d'une procédure de classification qui minimise l'erreur apparente pose peu de difficultés. Le problème est qu'un faible taux d'erreur apparente ne garantit pas un faible taux d'erreur réelle. L'erreur apparente est en général une version trop optimiste de l'erreur réelle. Des résultats théoriques ont montré que lorsque la taille de l'échantillon des exemples tendait vers l'infini, l'erreur apparente convergeait vers l'erreur réelle mais en général, le nombre d'exemples dont on dispose est trop petit pour tenir compte de ces résultats. Comment alors minimiser l'erreur apparente en assurant un faible taux d'erreur réelle, c'est-à-dire en garantissant une bonne capacité de généralisation ?

C'est ici qu'on retrouve à nouveau l'intérêt des biais d'apprentissage et l'importance du choix de l'espace des hypothèses. Il est possible de minimiser l'erreur apparente en complexifiant de plus en plus l'espace de recherche (en augmentant la taille des arbres de décisions) et finalement trouver un bon compromis entre cette complexité et un faible taux d'erreur réelle. La figure 83 illustre ce compromis. On peut remarquer que les taux d'erreurs réelle et apparente diminuent progressivement lorsque le nombre de feuilles de l'arbre augmente et que, pour un certain niveau de complexité de l'arbre, l'erreur réelle reste stable avant d'augmenter (zone de sur-apprentissage). C'est dans cette région de stabilité que se trouve l'hypothèse la plus intéressante, celle à fournir à l'utilisateur.

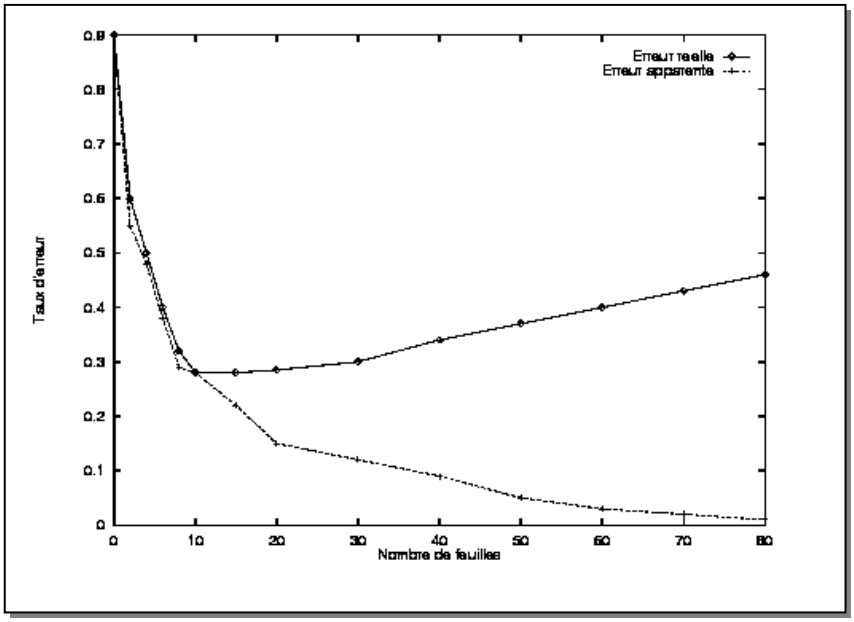


***Figure 83**. Évolution des taux d'erreurs réelle et apparente en fonction de la complexité d'un arbre de décision, pour des données de reconnaissance de caractères (D'après [Breiman et al. 1984]).*

Cette figure illustre à nouveau l'intérêt de la méthode « croître / élaguer » à laquelle ont recours les algorithmes d'apprentissage qui construisent des arbres. L'étape d'élagage permet de diminuer l'erreur réelle tout en garantissant une faible erreur apparente. Ceci suppose néanmoins que l'on soit capable d'estimer l'erreur réelle. D'une manière générale, à l'issue de l'apprentissage, il faut pouvoir évaluer l'hypothèse apprise et savoir si celle-ci a un bonne capacité prédictive.

ESTIMER L'ERREUR REELLE

L'évaluation *a priori* de l'erreur réelle est théoriquement possible mais quasiment impossible en pratique. L'estimation de la performance en apprentissage s'opère généralement de manière empirique, *a posteriori*. Les méthodes utilisées fournissent des résultats plus précis [Cornuéjols et Miclet 2002].

Notons qu'il s'agit bien ici d'une estimation de l'erreur réelle puisqu'il n'est pas possible de connaître la classe de tous les exemples potentiels pour la calculer. Cette erreur est inconnue et on cherche, à partir d'un échantillon, à l'évaluer.

La première méthode d'évaluation consiste à utiliser un échantillon de données test. L'idée est de séparer les exemples d'apprentissage aléatoirement en deux ensembles, dont l'un est utilisé pour apprendre et l'autre pour mesurer la qualité de l'hypothèse. Le taux d'erreur réelle est estimé par l'erreur apparente mesurée sur l'échantillon test, éventuellement associée à un intervalle de confiance. Étant donné que ce jeu test est indépendant de l'échantillon qui a servi à générer l'hypothèse, on considère que l'erreur apparente calculée sur le jeu test constitue une bonne approximation de l'erreur réelle.

Cette méthode n'est malheureusement pas toujours applicable. Elle requiert en effet un nombre suffisamment grand d'exemples d'apprentissage dont la disponibilité peut manquer. La qualité d'un apprentissage augmente avec la taille de l'échantillon.

Si le nombre d'exemples n'est pas suffisant[26], l'apprentissage peut donner de mauvais résultats. Ce manque de disponibilité des exemples est une difficulté récurrente pour la mise en œuvre de l'apprentissage. Ce problème se pose pour de nombreuses applications, notamment dans le domaine de l'information géographique.

La seconde méthode, permettant d'évaluer empiriquement l'erreur réelle en s'affranchissant de cette contrainte, est la validation croisée [Schaffer 1993]. Le principe est le suivant :

1. On divise l'échantillon d'apprentissage E au hasard, en k parties de taille équivalente $E_1,\ldots,E_k$;
2. Pour chaque partie E_i, (i variant de 1 à k) :
   - On applique une procédure d'apprentissage sur les exemples correspondant à l'échantillon E-E_i ;
   - On calcule l'erreur apparente sur la partie restante E_i.
3. L'erreur réelle est estimée par la moyenne des erreurs apparentes mesurées successivement.

Habituellement, le nombre de parties est fixé à 10. Parfois, la valeur de k correspond au nombre total d'exemples disponibles. Dans ce cas, le test est répété autant de fois qu'il y a d'exemples, mais évalué sur un exemple seulement à chaque fois (méthode « *leave-one-out* »). Cette procédure est relativement coûteuse en temps de calcul mais fournit une bonne estimation du taux d'erreur réelle [Mitchell 1997].

Pour conclure, précisons que toutes les méthodes de validation donnant des approximations de l'erreur réelle ne dispensent pas d'évaluer les règles apprises interactivement, à l'issue de la procédure [Mitchell 1997]. De faibles taux d'erreur n'empêchent nullement d'apprendre des hypothèses totalement erronées, qui ne reflètent pas le raisonnement de l'expert. Le problème d'apprentissage peut avoir été mal posé, exploitant des exemples qui ne sont pas suffisamment informatifs ou l'évaluation a pu être biaisée. La pertinence d'une hypothèse apprise doit donc toujours être analysée et ne jamais être acceptée aveuglément.

D.4.1.4 APPRENTISSAGE ET INFORMATION GEOGRAPHIQUE

Qu'en est-il de l'utilisation de ces techniques dans le domaine de l'information géographique ?

Depuis quelques années, plusieurs travaux ont été développés pour étendre les techniques de la *fouille de données*[27] aux bases de données spatiales [Miller et Han 2001]. Ce qui distingue principalement l'analyse de données spatiales des méthodes traditionnelles est la prise en compte des relations spatiales entre les objets [Zeitouni et Yeh 1999, Aufaure et al. 2000]. Les notions de dépendance et d'hétérogénéité

[26] Les travaux théoriques en apprentissage n'ont pas pu fixer précisément le nombre d'exemples nécessaires pour obtenir une « bonne » hypothèse. Ce nombre varie d'une application à l'autre. En pratique, on considère qu'il faut disposer de plus d'une centaine d'exemples mais ce nombre est très approximatif et dépend de la complexité des exemples (du nombre d'attributs).

[27] La fouille de données (DM : « *Data Mining* ») fait partie du processus plus général d'extraction de connaissances à partir de données (KDD : « *Knowledge Discovery in Databases* ») [Fayyad et al. 1996]. Les outils qu'elle utilise sont issus de différents domaines : les bases de données, l'analyse de données et l'apprentissage inductif.

spatiale sont en effet fondamentales en géographie. A l'image de la fouille des bases de données traditionnelles, l'exploration des données géographiques exploite des méthodes empruntées aux statistiques spatiales [Cressie 1993], aux bases de données spatiales et plus récemment, à l'intelligence artificielle. Les récentes contributions ont apporté des développements, notamment en matière de recherche de règles d'associations spatiales [Koperski et Han 1995, Appice et al. 2003] et de *clustering* [Han et al. 2001]. Certains auteurs ont également adapté des langages d'interrogation de bases de données pour exprimer des tâches d'exploration dans une requête spatiale [Malerba et al. 2002]. Un prototype dédié au *data mining spatial* a par ailleurs été mis au point par une équipe du laboratoire de base de données de l'université Simon Fraser au Canada : il s'agit de GeoMiner [Han et al. 1997].

Il faut noter que la plupart de ces travaux se sont focalisés sur des méthodes d'apprentissage ou de classification non supervisée. Il existe relativement peu d'adaptations d'algorithmes d'apprentissage supervisé symbolique. Mentionnons toutefois les contributions de [Ester et al. 1997, Koperski et al. 1998]. La méthode de classification proposée par les premiers auteurs est fondée sur l'algorithme ID3 [Quinlan 1986] et la construction d'un graphe de voisinage. Ils proposent de développer un arbre de décision en prenant en compte non seulement les attributs des objets à classer, mais aussi la nature des objets présents dans le voisinage. De cette manière, il est possible de découvrir des règles qui indiquent par exemple que le pouvoir économique d'une ville est élevé parce que sa population est élevée et qu'il existe un aéroport à proximité (voisin de la ville). Les objets sont traités en tant que voisins s'ils satisfont une relation de voisinage qui peut être topologique et métrique (un seuil étant fixé pour la distance). La proposition de [Koperski et al. 1998] est assez proche de la précédente mais prend en compte davantage d'informations. Elle exploite ainsi les attributs des objets voisins et agrège la valeur des attributs non spatiaux de ceux-ci lorsqu'ils sont identiques. Les propriétés relatives à chaque voisin et au groupe de voisins identiques sont donc utilisées. Les relations spatiales sont décrites sous forme de prédicats.

Ces propositions sont intéressantes et on ne peut qu'espérer qu'elles se développent davantage. Ceci n'empêche pas pour autant d'utiliser les algorithmes d'apprentissage supervisé non spécifiques aux données géographiques pour explorer les bases de données spatiales [Gahegan 2002]. Tout dépend du type d'analyse que l'on souhaite réaliser et de la tâche d'apprentissage à accomplir. Plusieurs expérimentations ont ainsi été développées en utilisant des algorithmes classiques d'apprentissage (comme C4.5. [Quinlan 1993] ou RIPPER [Cohen 1995]) dans un contexte cartographique [Duckham et al. 2000, Mustière et al. 2000b, Sester 2000, Elias 2003]. Ces algorithmes d'apprentissage fournissent de bons résultats si le problème d'apprentissage est bien posé.

> *C'est l'approche que nous adoptons dans cette thèse. Nous utilisons des algorithmes d'apprentissage supervisé symboliques non spécifiques aux données géographiques. D'un point de vue méthodologique, nous n'excluons pas la possibilité d'utiliser d'autres techniques, celles auxquelles fait appel la fouille de données, spatiale ou non, comme les statistiques, mais nous nous restreignons en pratique à exploiter ici l'apprentissage inductif.*

La principale difficulté dans la mise en œuvre de l'apprentissage réside dans la définition du problème. Deux questions se posent :

- Quelles sont les propriétés spatiales et non spatiales pertinentes qui vont permettre de facilement discriminer les exemples de l'échantillon d'apprentissage ?
- Comment exprimer au mieux ces propriétés spatiales dans les langages de représentation acceptés par les algorithmes d'apprentissage ?

La réponse à ces questions n'est pas triviale. L'identification des propriétés pertinentes est un problème propre à toute mise en œuvre d'une tâche d'apprentissage supervisée. La description des propriétés spatiales est quant à elle une difficulté spécifique à l'information géographique. Comment décrire au mieux la forme d'un objet ? Quelles mesures reflètent ses caractères géométriques tels que la sinuosité, l'élongation ou l'orientation, caractères qui sont par ailleurs implicites (cf. chapitre précédent) ? L'avis d'experts et l'analyse des spécifications doivent nécessairement intervenir à ce niveau. Cela peut aider à identifier les bonnes mesures qui décrivent symboliquement la représentation de chaque objet géométrique sous forme d'attributs. Une fois les mesures identifiées et évaluées, l'échantillon d'exemples d'apprentissage peut être construit et la tâche d'apprentissage expérimentée.

> *Ceci justifie la phase d'enrichissement que nous proposons avant de mener l'interprétation des différences de représentation (chapitre C). L'enrichissement vise aussi à extraire les propriétés implicites des objets géométriques pour construire des exemples d'apprentissage pertinents.*

D.4.2 MISE EN OEUVRE DE L'APPRENTISSAGE

D.4.2.1 ALGORITHMES D'APPRENTISSAGE EXPLOITES

Avant d'expliquer la manière de mettre en œuvre les techniques d'apprentissage dans le cadre de notre méthodologie d'évaluation, nous devons préciser que nous nous plaçons en tant qu'utilisateur de celles-ci et en particulier, de l'apprentissage inductif supervisé. Nous n'avons donc pas développé de nouvel algorithme d'apprentissage ni adapté un algorithme existant. Nous nous sommes concentrés sur la manipulation d'algorithmes qui étaient à notre disposition.

De ce point de vue, nous avons dû faire un choix sur la méthode d'apprentissage à utiliser et les algorithmes à exploiter, de manière à sélectionner des outils adaptés au problème traité. Puisque les connaissances que nous souhaitons acquérir sont destinées à être intégrées dans un système-expert, nous avons jugé que les outils d'apprentissage symbolique étaient les plus adaptés. Les hypothèses induites par de tels outils sont plus facilement compréhensibles car elles sont généralement exprimées sous forme de règles de décision ou d'arbres de décision. Celles-ci peuvent donc aisément être contrôlées et révisées au besoin. Les règles de décision peuvent en outre être directement insérées dans la base de règles du système-expert (en les traduisant au préalable dans le langage de celui-ci), ce qui constitue un gain de temps important.

De nombreux algorithmes d'apprentissage supervisé symbolique ont été mis au point par les chercheurs en intelligence artificielle. Parmi ceux-ci, deux algorithmes sont particulièrement populaires et reconnus : C4.5. [Quinlan 1993] et RIPPER [Cohen 1995]. En raison de leur efficacité prouvée dans des domaines très variés, ce sont ces algorithmes que nous avons retenus pour entreprendre nos expérimentations. Ceux-ci présentent comme biais de représentation des exemples, un langage attribut/valeur.

Les exemples sont donc décrits par un ensemble de valeurs d'attributs sélectionnés par l'expert et jugés pertinents pour le problème d'apprentissage posé.

D.4.2.2 APPRENDRE POUR AIDER A ENRICHIR LES DONNEES

La première étape pour laquelle l'apprentissage peut apporter une aide est l'enrichissement (étape de *MECO*, cf. chapitre C). L'apprentissage peut être utile pour déterminer automatiquement les paramètres des indicateurs, comme les indicateurs de forme par exemple.

Dans le chapitre précédent, nous avions pris l'exemple de carrefours particuliers (les pattes d'oie) pour exposer la méthode *MECO*. Nous avions précisé que l'existence de ces objets pouvait être implicite dans les bases et que leur extraction pouvait être requise si l'étude du respect des spécifications l'exigeait. Comment pourrait-on extraire ces objets et quel peut être le rôle de l'apprentissage dans cet enrichissement ?

Lors d'un stage effectué à l'IGN, une étude a été menée pour détecter divers types de carrefours dont les pattes d'oie [Grosso 2004]. L'extraction des pattes d'oie a été rendue possible après une analyse et une caractérisation de chaque face constituée par les tronçons de route (ce qui suppose la constitution d'une structure topologique). Les propriétés retenues pour sélectionner les faces correspondant aux pattes d'oie furent les suivantes :

- Une face est candidate si elle est constituée de 3 nœuds de degré 3 et si sa superficie < $seuil_1$.
- Une face est candidate si la distance surfacique entre celle-ci et le triangle qu'il est possible de construire en reliant les trois nœuds < $seuil_2$.

Dans cette étude, les seuils ont été fixés empiriquement, de manière interactive. C'est la solution la plus fréquemment adoptée pour paramétrer des indicateurs ou des algorithmes géométriques destinés à ce type de tâches [Trévisan 2005]. Toutefois, il est possible de déterminer ces seuils automatiquement, en exploitant une méthode d'apprentissage supervisé. A partir d'un ensemble d'exemples étiquetés, c'est-à-dire des faces pour lesquelles les propriétés précédentes auraient été calculées et dont la classe aurait été identifiée manuellement (une *patte d'oie* ou un *autre carrefour*), il est possible d'appliquer un algorithme d'apprentissage pour déterminer la relation entre la classe des exemples et leur descripteurs (figure 84). Cette solution permettrait d'éviter la multiplication des tests destinés à trouver les bons seuils interactivement.

On peut également citer à ce sujet les travaux de [Sester 2000] concernant l'identification de parcelles et de routes sur base de critères de forme, en utilisant des données cadastrales ou encore les travaux de [Plazanet et al. 1998, Mustière et al. 2000b] pour qualifier de manière symbolique la forme de routes (ex : lisse/sinueux) ou de bâtiments (ex : rectangulaire/en L/ en escalier) ;

L'apprentissage peut ainsi aider à paramétrer les outils d'analyse spatiale utilisés lors de l'enrichissement. Sa mise en œuvre a toutefois un coût. La construction d'exemples prend du temps. Son utilisation n'a donc de sens que si la détermination interactive des seuils est difficile et que le nombre de propriétés est élevé.

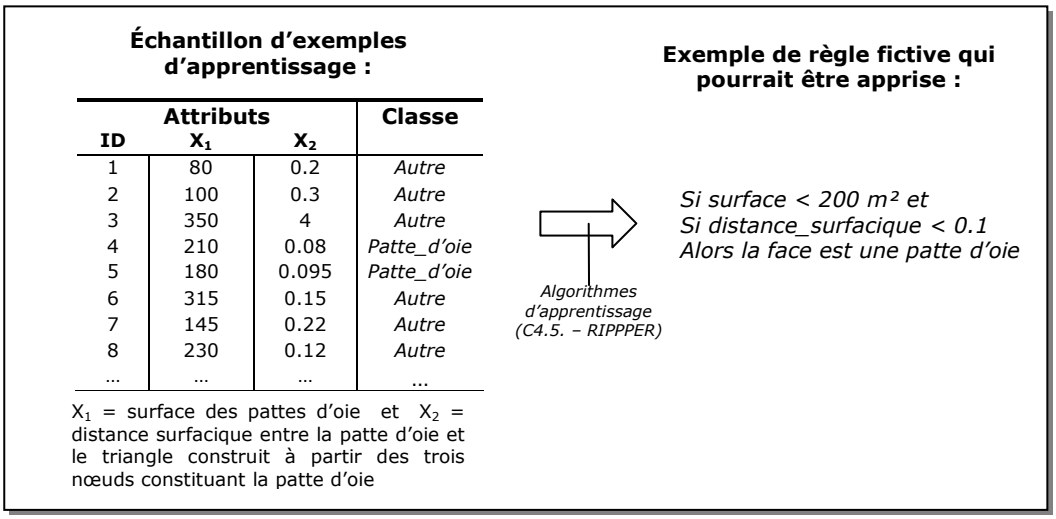


***Figure 84**. L'apprentissage peut aider à déterminer automatiquement des règles permettant de relier un concept que l'on souhaite extraire des données (les pattes d'oie) aux descripteurs qui le caractérisent (les mesures effectuées sur les faces).*

D.4.2.3 APPRENDRE POUR ACQUERIR DES REGLES RELATIVES AU CONTROLE INTER-BASES

L'utilisation de l'apprentissage pour acquérir des règles destinées au contrôle inter-bases s'impose plus fréquemment. Ces techniques peuvent être mises en œuvre :

- En raison du manque de spécifications, de leur imprécision et d'une manière générale, de leur insuffisance, voire en raison de l'absence totale de spécifications ;
- Dans l'optique de découvrir l'écart existant entre les spécifications décrites dans les documents et celles contenues dans les données qui sont effectivement respectées par les opérateurs de saisie ;
- Dans l'optique d'automatiser l'acquisition de règles pour le contrôle inter-bases si les spécifications sont complexes et les règles trop nombreuses ;

Dans le chapitre consacré à la méthode *MECO*, nous avons exposé deux solutions différentes pour organiser les connaissances destinées à contrôler la cohérence inter-représentations. La première est la classification directe et la seconde se compose des étapes de prédiction, comparaison et classification (cf. C.5.4.). Nous allons reprendre ces deux solutions et expliquer comment des règles s'y rapportant peuvent être acquises automatiquement par apprentissage.

APPRENDRE DES REGLES DE CLASSIFICATION DIRECTE

La première manière d'organiser les connaissances permettant d'interpréter les différences de représentation des objets est la classification directe (C.5.4.1). Rappelons que le principe de cette approche consiste à décrire les connaissances sous forme de règles qui spécifient directement si les couples d'objets appariés ont des représentations équivalentes ou incohérentes, en tenant compte de la forme des représentations de chaque objet. Les règles sont ainsi formulées de la manière suivante :

$$SI\ condition_A\ (O_{1i}, O_{2j})\ ALORS\ (O_{1i}, O_{2j})\ est\ équivalent$$

Ces règles peuvent être acquises par apprentissage supervisé. Pour cette forme de règle, un exemple correspond à un couple d'objets appariés *caractérisé*, autrement dit, décrit par un ensemble d'attributs, dont la classe (*incohérence* ou *équivalence*) est attribuée interactivement par un expert. C'est l'expert qui construit ainsi les exemples d'apprentissage sur lesquels s'applique un algorithme d'apprentissage, de manière à obtenir des règles ou un arbre de décision.

Illustrons ce processus d'apprentissage en reprenant l'exemple des pattes d'oie. On peut imaginer vouloir apprendre des règles permettant de classer directement les représentations de couples de pattes d'oie en incohérence ou équivalence. Pour réaliser cet apprentissage, il faut recueillir des exemples et préciser les conditions qui portent sur les couples. Il faut donc apparier les pattes d'oie des deux bases, sélectionner un certain nombre de couples et décrire les objets les constituant. La description est une tâche importante car la qualité des règles apprises en dépend. Les descripteurs doivent être choisis judicieusement car ce sont eux qui vont permettre de discriminer les exemples relatifs à chaque classe à apprendre. C'est une tâche qui incombe à l'expert et qui doit être guidée par les connaissances du domaine ou si possible, par les spécifications. Pour les pattes d'oie, les descripteurs pourraient correspondre au type de modélisation des objets, la longueur de leur base et leur superficie par exemple. Chaque couple de pattes d'oie doit ensuite être classé. En appliquant un algorithme d'apprentissage, un ensemble de règles de classification peut ainsi être découvert (figure 85).

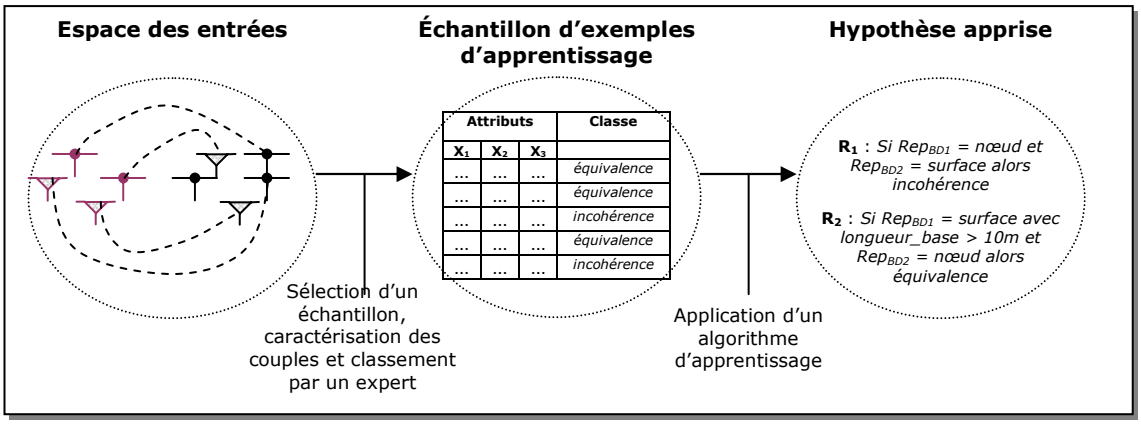


***Figure 85**. Mise en œuvre de l'apprentissage pour classer directement la représentation des objets de couples de pattes d'oie en équivalence ou incohérence.*

APPRENDRE DES REGLES DE PREDICTION

La seconde manière d'organiser les connaissances pour mener le contrôle inter-bases est de définir des règles de prédiction, de comparaison et de classification (C.5.4.2.). Le principe de cette approche, rappelons-le, est de déterminer les conditions que doit respecter la forme des objets de la première base à partir des objets de la seconde base, et les conditions que doit respecter la forme des objets de la seconde base à partir des objets de la première base. Une fois les conditions sur les représentations prédites déterminées, il suffit de les comparer aux représentations effectivement contenues dans les données. Si celles-ci respectent les conditions apprises, les représentations sont jugées équivalentes. Dans le cas contraire, les représentations sont incohérentes. En terme de règles, la prédiction s'exprime sous la forme :

SI condition$_A$ (O_{1i}) ALORS O_{2j} doit respecter condition$_B$
SI condition$_C$ (O_{2j}) ALORS O_{1i} doit respecter condition$_D$

Pour cette approche, l'apprentissage supervisé peut également être mis en œuvre. L'apprentissage concerne la phase de prédiction et doit donc être mené dans les deux directions (prédiction des conditions que doivent respecter les représentations de la BD_1 et prédiction des conditions que doivent respecter les représentations de la BD_2). Un exemple d'apprentissage est de ce fait constitué de descripteurs relatifs à la représentation d'un des objets du couple, la classe de cet exemple correspondant aux conditions que doit satisfaire la représentation de l'autre objet du couple. En appliquant un algorithme d'apprentissage, des règles de prédiction peuvent ainsi être découvertes.

Nous pouvons illustrer cette approche pour les pattes d'oie. Le recueil des exemples suit le processus d'appariement. Parmi les couples calculés, un sous-ensemble doit être sélectionné pour constituer les exemples d'apprentissage. Un exemple est d'abord définit par une série de descripteurs décrivant la représentation d'un des objets du couple. Il peut s'agir du type de modélisation de la patte d'oie (ponctuelle ou détaillée) et de la longueur de la base de la patte d'oie. On pourrait également tenir compte des attributs des objets. Un exemple est ensuite étiqueté. La classe correspond à la représentation de l'autre objet du couple associée aux conditions qu'elle doit satisfaire. Autrement dit, la classe correspond à la représentation de l'autre objet associée à une des catégories de représentation qu'elle peut avoir (par exemple, une représentation ponctuelle ou détaillée). On applique ensuite un algorithme d'apprentissage afin d'apprendre des règles permettant de prédire les conditions que doivent satisfaire les représentations des pattes d'oie de la seconde base connaissant les représentations de la première, et inversement (figure 86).

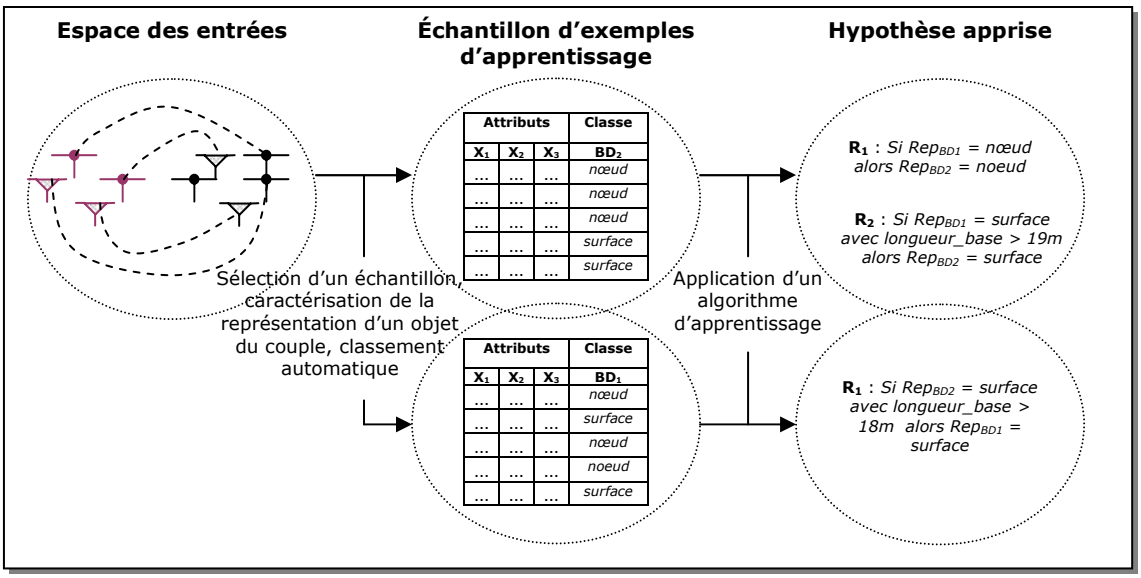


***Figure 86**. Mise en œuvre de l'apprentissage pour prédire les conditions que doivent respecter les représentations de chaque base connaissant les représentations de l'autre base.*

COMPARAISON DES DEUX APPROCHES

Nous discutons ci-dessous des deux approches car elles présentent des différences importantes du point de vue de l'apprentissage, de sa mise en œuvre et

des sources de connaissances acquises (un tableau comparatif est donné en fin de section).

D'abord, revenons sur la classification directe. Cette solution est sans doute la plus naturelle pour organiser les connaissances permettant d'évaluer la cohérence inter-représentations. La construction des exemples pour la classification directe implique l'intervention de l'expert. Celui-ci doit attribuer la classe de chaque exemple interactivement (ce qui est généralement le cas de toute mise en œuvre d'un apprentissage supervisé). Le recueil des exemples est une tâche qui demande du temps et par conséquent, le coût de l'apprentissage n'est pas négligeable.

Puisque c'est l'expert qui attribue la classe de chaque exemple, les connaissances apprises découlent des connaissances de l'expert. L'hypothèse apprise peut donc offrir des règles légèrement différentes des spécifications indiquées dans les documents. On apprend en effet la vision de cet expert et les règles implicites qu'il utilise pour décider de la conformité des représentations. En termes d'évaluation, cela signifie qu'on utilise des connaissances plus proche de la réalité de la saisie. L'expert tient compte des spécifications pour attribuer la classe mais admet une certaine tolérance sur les seuils qu'il rencontre car il sait par expérience que lors de la saisie des objets, cette tolérance est introduite. Celle-ci varie légèrement d'un opérateur à l'autre et si les classes des exemples d'apprentissage étaient attribuées par un autre expert, cette variation apparaîtrait dans les règles apprises. Du point de vue de l'évaluation, cette variation peut engendrer des petites différences d'interprétation, mais nous considérons que la qualité de l'évaluation est la même. Nous accordons la même compétence à chaque expert.

La classification directe est donc une bonne solution pour décrire les règles d'évaluation. Elle peut s'appliquer pour n'importe quel type de différence. Les règles apprises correspondent aux connaissances d'un expert. La construction des exemples est toutefois une tâche fastidieuse. C'est le principal défaut de l'approche.

La seconde solution proposée présente un avantage par rapport à la classification directe. L'intervention de l'expert peut être évitée. En effet, puisqu'une des deux représentations des objets du couple est utilisée pour déterminer la classe de l'exemple, celle-ci est fixée automatiquement à l'issue de l'appariement, en l'associant aux conditions qu'elle doit satisfaire. Par exemple, pour les pattes d'oie, si on attribue à chaque représentation ponctuelle et détaillée à prédire un attribut « type de modélisation », en leur affectant leur valeur (*nœud* ou *surface*), les exemples d'apprentissage sont automatiquement construits au terme de l'appariement. Un algorithme d'apprentissage peut donc directement être appliqué pour découvrir les règles de prédiction. L'intervention de l'expert n'est plus nécessairement requise pour déterminer la classe de chaque exemple. Elle est juste limitée à la définition de la forme des exemples.

La source des connaissances acquises correspond donc cette fois aux données. Les règles reflètent les connaissances implicites utilisées lors de la saisie par le ou les multiples opérateurs mais non celles de l'expert. La tolérance sur les seuils des spécifications que s'accordent les opérateurs de saisie peut à nouveau être mise en évidence.

C'est une solution qui est particulièrement intéressante pour la construction des exemples d'apprentissage mais, en contrepartie, on est susceptible d'apprendre avec davantage d'exemples bruités, c'est-à-dire des exemples mal étiquetés. En effet, la classe des exemples n'est pas contrôlée à l'issue de l'appariement et de ce fait, les

couples incohérents sélectionnés sont utilisés pour apprendre. De plus, du bruit peut être introduit à cause d'erreurs d'appariement. Cela signifie que si le nombre d'exemples bruités (les incohérences et les erreurs d'appariement) est trop important, on risque d'apprendre des règles qui ne correspondent pas aux spécifications des bases. Du point de vue de l'induction, on risque de faire du *sur-apprentissage* (cf. D.4.1.2.). On peut imaginer passer en revue chaque exemple d'apprentissage pour vérifier l'exactitude de leur classe mais on perd dans ce cas l'avantage de la méthode : l'intervention de l'expert est à nouveau requise. Pour limiter la sélection d'exemples bruités nous préconisons plutôt un filtrage fondé sur le résultat du contrôle intra-base. Si le contrôle intra-base a jugé la représentation comme non conforme, l'exemple n'est pas utilisé pour apprendre. C'est ce qui justifie la position du contrôle intra-base dans *MECO*. Il doit être mené avant le contrôle inter-bases. Nous pouvons également limiter le nombre d'exemples bruités en écartant les couples d'objets appariés incertains. En l'absence de spécifications, ce filtrage n'est toutefois pas possible. Dans ce cas, il est préférable d'utiliser des algorithmes d'apprentissage adaptés aux données bruitées [Liu 1996, Decatur 1997, Azé et Kodratoff 2002].

Dans le cas d'expérimentations menées sur des ronds-points, nous avons étudié la relation existant entre le nombre d'exemples bruités dans un jeu d'exemples d'apprentissage, c'est-à-dire le nombre d'incohérences, et l'erreur réelle portant sur les règles apprises à partir du jeu d'exemples (cf. chapitre E). Nous avons pu mettre en évidence la forte corrélation positive existant entre ces deux valeurs (nous avons calculé un coefficient de corrélation de 0.98 lors de ces tests). Ceci nous a permis de conclure que, pour un taux d'erreur réelle de 20% sur des règles apprises, on pouvait supposer approximativement l'existence de 20% d'incohérences dans le jeu d'exemples. La capacité inductive de l'algorithme fut donc particulièrement bonne.

Ces résultats sont intéressants mais nous pensons qu'aujourd'hui il est encore trop tôt pour généraliser cette conclusion. En effet, on a pu constater cette corrélation car l'algorithme d'apprentissage n'a jamais fait de sur-apprentissage lors de ces tests. L'erreur réelle calculée augmentait toujours dans le même sens que le taux d'erreurs introduit (jusqu'à 40%). Les règles apprises étaient donc toujours en accord avec les spécifications. Mais en cas de sur-apprentissage, l'erreur réelle aurait brusquement chutée (pour un taux de bruit plus important) ce qui aurait pu remettre en cause la corrélation. Ces tests mériteraient donc d'être approfondis avant de généraliser cette conclusion.

Un autre aspect limitatif de l'apprentissage de règles de prédiction est lié au type d'algorithmes d'apprentissage utilisés. Bien souvent, les algorithmes d'apprentissage symbolique ne permettent pas d'attribuer une classe de valeur numérique aux exemples. Par conséquent, on ne peut pas toujours apprendre la classe que l'on souhaite. Par exemple, nous avons autorisé uniquement deux valeurs pour la classe se rapportant aux pattes d'oie à prédire en figure 86 : la valeur *nœud* (i.e. représentation ponctuelle) et la valeur *surfac*e (i.e. représentation détaillée). Il aurait été plus riche d'apprendre pour une représentation détaillée la valeur approximative de la longueur de la base ou plus exactement, les bornes inférieures et/ou supérieures (*la représentation détaillée doit avoir une base de longueur supérieure à 20m* par exemple). Cependant, les algorithmes symboliques que nous utilisons ne nous le permettent pas. On peut toutefois envisager de discrétiser les valeurs numériques [Dougherty et al. 1995, Liu et al. 2002]. On pourrait ainsi distinguer les représentations détaillées des pattes d'oie de base inférieure à 20m (représentation non conforme si on se réfère aux spécifications définies précédemment en figure 55.) des représentations détaillées de base supérieure à 20m (représentation conforme). Cette distinction permettrait de préciser les conditions que doivent respecter les

représentations de la seconde base. Néanmoins, cette discrétisation dépend fortement des spécifications. Un choix *a priori* des seuils de discrétisation est difficile. Les spécifications doivent être connues (le seuil de 20m n'est pas pris au hasard), ce qui n'est pas toujours le cas. D'autres types d'algorithmes d'apprentissage symbolique ne nécessitant pas une discrétisation de la classe devraient donc être appliqués dans ce cas. L'utilisation d'outils d'apprentissage numérique peut également être envisagée mais c'est au détriment de la lisibilité des règles apprises.

L'apprentissage de règles de prédiction semble donc particulièrement intéressant du point de vue du recueil des exemples car cette tâche ne nécessite pas l'intervention d'un expert. En revanche, le risque d'apprendre des règles qui ne correspondent pas aux spécifications des bases est bien plus important et les conditions apprises portant sur les représentations peuvent manquer de précision. Un contrôle *a posteriori* des règles est donc systématiquement nécessaire.

Pour conclure, nous présentons un tableau récapitulatif des caractéristiques et des avantages et faiblesses des deux solutions de description des règles pour le contrôle inter-bases (tableau 1). Puisque pour acquérir ces règles, il n'est pas toujours nécessaire d'utiliser l'apprentissage, nous distinguons les cas pour lesquels les règles sont définies manuellement, à partir des spécifications, et automatiquement, par apprentissage.

Nous considérons qu'une évaluation qui tient compte des seuils fixés dans les spécifications est moins tolérante puisqu'ils ne sont pas rigoureusement respectés par les opérateurs de saisie en pratique. Néanmoins, on pourrait envisager de définir une tolérance sur ces seuils (sans l'apprendre) pour rendre ainsi l'évaluation plus souple. Mais la définition de cette tolérance sans analyse des données n'est pas évidente. Seul un expert du domaine pourrait donner un ordre de grandeur.

Tableau 1. *Comparaison des deux solutions proposées pour organiser les connaissances relatives au contrôle inter-bases dans une base de règles*

	Classification directe	**Prédiction**
Connaissances acquises par analyse des spécifications	• S'applique si les spécifications sont simples et suffisantes • Les seuils des règles correspondent aux seuils fixés dans les spécifications • Évaluation peu tolérante aux écarts existant entre les spécifications et la réalité de la saisie.	• S'applique si les spécifications sont simples et suffisantes • Les seuils des règles correspondent aux seuils fixés dans les spécifications • Évaluation peu tolérante aux écarts existant entre les spécifications et la réalité de la saisie.
Connaissances acquises par apprentissage	• S'applique si les spécifications sont trop complexes, trop nombreuses ou insuffisantes. • Source des connaissances : l'expert et les données • Les connaissances apprises reflètent les connaissances de l'expert. • La construction des exemples requiert l'intervention de l'expert. • Mise en évidence de l'écart toléré par l'expert sur les seuils fixés dans les spécifications • Évaluation plus proche de la réalité des bases	• S'applique si les spécifications sont trop complexes, trop nombreuses ou insuffisantes. • Source des connaissances : les données • Les connaissances apprises reflètent les connaissances des opérateurs de saisie. • La construction des exemples est automatique. • Mise en évidence de l'écart toléré par les opérateurs de saisie sur les seuils fixés dans les spécifications • Évaluation plus proche de la réalité des bases • Risque d'apprendre des règles insuffisamment représentatives des spécifications • Nécessite systématiquement un contrôle a posteriori

D.4.2.4 ÉVALUATION DES CONNAISSANCES APPRISES

Lorsqu'on met en œuvre une méthode d'apprentissage, il est essentiel d'évaluer la pertinence des connaissances apprises. Cette évaluation doit tenir compte :

- du taux d'erreur réelle calculé pour l'hypothèse apprise ;
- des résultats de l'analyse interactive des règles apprises ou de leur application sur de nouveaux exemples ;

Ainsi, la première information à récolter à l'issue de l'apprentissage est le taux d'erreur réelle. Rappelons que le taux d'erreur réelle correspond au pourcentage d'erreur que l'hypothèse apprise effectue sur l'ensemble des exemples possibles (D.4.1.3.). Celle-ci est généralement estimée par validation croisée.

Le taux d'erreur réelle donne une première idée de la qualité de l'hypothèse proposée. Il permet de savoir si le problème d'apprentissage a été bien posé, c'est-à-dire, si les exemples sont suffisamment bien décrits et différenciés pour permettre de les discriminer et donc d'apprendre. Un taux d'erreur réelle correspondant à une

hypothèse appliquant une procédure majoritaire indique que l'algorithme n'est pas capable d'apprendre puisque l'hypothèse acquise ne classe pas mieux les nouveaux exemples qu'une procédure attribuant automatiquement la classe la plus fréquente. En-dessous de ce taux, il est difficile de fixer une limite au-delà de laquelle l'apprentissage doit être considéré comme inefficace. L'évaluation interactive des règles doit intervenir dans ce cas. Cette limite est d'autant plus difficile à fixer dans notre contexte car nous pouvons apprendre avec un certain nombre d'exemples bruités (cas de la prédiction). Le taux d'erreur est donc susceptible d'être élevé pour des règles en accord avec la réalité de la saisie.

L'analyse interactive des règles ou l'application des règles apprises sur de nouveaux exemples fait donc aussi partie de l'évaluation de l'hypothèse obtenue par apprentissage, après avoir vérifié que le taux d'erreur réelle était acceptable. Les règles doivent refléter les spécifications des bases ou les connaissances implicites utilisées lors de la saisie des données. Elles doivent donc être analysées par un expert ou être comparées aux spécifications. S'il s'avère que les règles sont trop contradictoires avec les spécifications, elles doivent être révisées.

D.4.2.5 REVISION DES CONNAISSANCES APPRISES

Il convient de bien faire la distinction entre révision et validation [Dupin de Saint-Cyr et Loiseau 2000].

Les connaissances que nous apprenons sont destinées à être manipulées par un système-expert. La conception de ces systèmes doit être suivi d'une phase de *validation* durant laquelle la cohérence de la base de connaissances est contrôlée [Ayel et Rousset 1990]. Ce contrôle ne concerne pas la véracité des règles (par rapport à la réalité) mais leur qualité interne. La validation ne vérifie donc pas l'exactitude des règles mais plutôt la présence de règles incohérentes, inutiles ou mal énoncées dans la base de connaissances.

La révision en revanche a pour but de modifier la base de connaissances afin que cette dernière puisse intégrer une nouvelle connaissance ou rendre ces connaissances plus exactes, tout en conservant la cohérence du système. C'est à ce problème que nous nous intéressons ici.

En matière de révision, de nombreux travaux ont été menés, y compris dans le contexte de l'information géographique [Jeansoulin et Papini 2000][28]. La révision est généralement abordée sous un angle très formel (comme les théories AGM [Alchourron et al. 1985] ou KM [Katsuno et Mendelzon 1991]) et semble difficilement applicable en pratique. Nous traitons de ce fait le problème de la révision de manière informelle, à l'image du travail réalisé par [Bard 2000].

La révision doit d'abord être envisagée au début de la mise en place de la base de connaissances, c'est-à-dire lors de l'introduction des règles apprises dans le système-expert. Dans ce cas, la révision suit l'évaluation des connaissances apprises. On pourrait d'ailleurs considérer que l'évaluation est une étape de la révision. La démarche de révision consiste dans un premier temps à détecter les règles inexactes, ensuite à les modifier, et finalement à valider la base révisée. La détection des règles inexactes se fait manuellement, d'une part en analysant directement les règles, et d'autre part, en analysant un échantillon de couples interprétés. L'analyse d'un

[28] Une partie du projet européen REVIGIS (« *Revision of the Uncertain Geographic Information* » - IST/1999/1489) est d'ailleurs consacrée à ce sujet : http://www.lsis.org/REVIGIS/

échantillon permet parfois de remettre en cause des règles ou de les enrichir alors qu'elles semblaient pertinentes (ces modifications apparaissent souvent pour prendre en compte les exceptions dans les données).

La révision doit ensuite être envisagée après une série d'apprentissages successifs menés sur des données de différentes régions. Les résultats de l'apprentissage et les descripteurs choisis sont en effet susceptibles de différer en fonction de la localisation des données sur le territoire : certaines régions présentent des particularités que d'autres n'ont pas (routes à lacets pour les régions montagneuses par exemple) ; les régions ne sont pas non plus saisies par les mêmes opérateurs, ce qui signifie que d'une région à l'autre, les connaissances implicites utilisées ne sont pas tout à fait les mêmes. Les règles relatives à un apprentissage exploitant des données d'une région peuvent donc être modifiées ou enrichies après d'autres procédures d'apprentissage de manière à les rendre applicables sur tout le territoire et prendre en compte les particularités de toutes les régions.

D.5 SYNTHESE DE LA METHODE *MACO*

La méthode *MACO* que nous venons de présenter constitue la deuxième facette de notre méthodologie d'évaluation de la cohérence. Son application permet de recueillir les connaissances utilisées par *MECO*.

Deux étapes composent la méthode : l'analyse des spécifications et l'apprentissage automatique. La première étape s'impose systématiquement. Elle est destinée d'une part à spécifier les caractéristiques des outils d'enrichissement et d'appariement. Elle permet d'autre part d'alimenter les bases de règles utiles aux contrôles intra-base et inter-bases. Cette analyse peut être facilitée par l'emploi d'un modèle de représentation des spécifications. En décrivant les spécifications sous une forme normalisée, il plus facile d'identifier les règles imprécises à affiner et de comparer les documents des différentes bases.

L'apprentissage est la seconde étape de la méthode. Elle n'est pas systématiquement mise en œuvre. Elle doit l'être si les spécifications ne suffisent pas à concevoir la base de règles pour le contrôle inter-bases ou si l'expert souhaite réaliser une évalution qui tienne compte de la réalité de la saisie. Les exemples d'apprentissage peuvent être représentés en suivant l'approche *par classificaiton directe* ou l'approche par *prédiction, comparaison et classification*.

La méthode *MACO* est synthétisée en figure 87

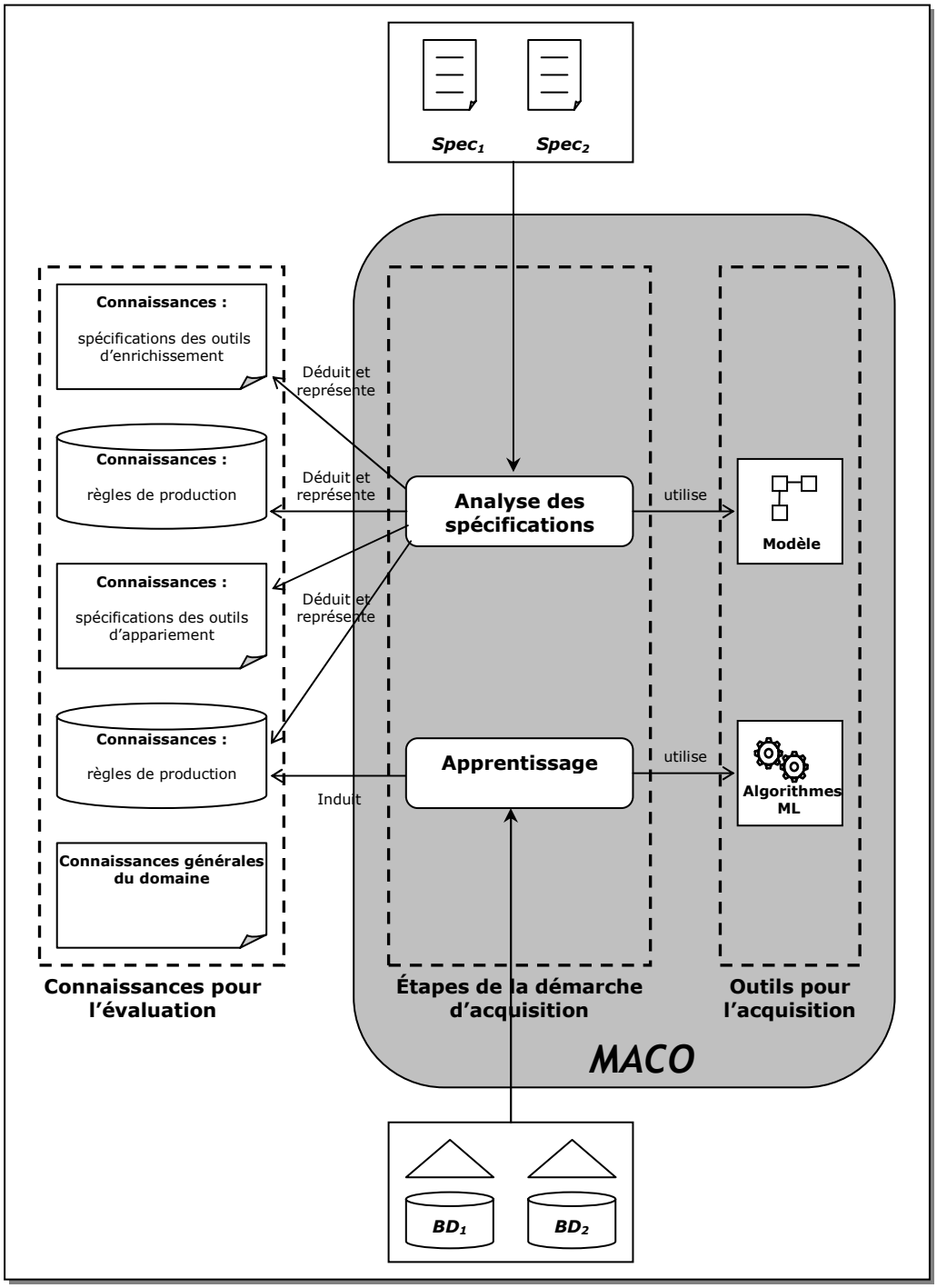


Figure 87. *Synthèse de la méthode MACO.*

D.6 SYNTHESE DE LA METHODOLOGIE D'EVALUATION

Pour appliquer la méthodologie que nous proposons, constituée des méthodes *MECO* et *MACO*, nous recommandons de procéder de la manière suivante :

- Un expert formalise la description des spécifications en adoptant le modèle défini en D.3.2. (cette étape s'impose si la structure des documents est très hétérogène) - tâche dans *MACO* ;

Pour mettre en œuvre l'enrichissement :

- Un expert identifie les objets à extraire et les propriétés à mesurer après une analyse des spécifications (analyse individuelle et croisée des documents) - tâche dans *MACO* ;
- Un expert enrichit les schémas des bases – tâche dans *MECO* ;
- Un expert développe ou sélectionne des outils d'analyse spatiale pour instancier les schémas enrichis – tâche dans *MECO* ;
- Les algorithmes définis sont appliqués pour enrichir les données des bases - tâche dans *MECO*.

Pour mettre en œuvre le contrôle intra-base :

- Un expert identifie les connaissances utiles au contrôle en analysant les spécifications de chaque base (analyse individuelle des documents) - tâche dans *MACO*.
- Un expert représente les connaissances sous forme de règles et les introduit dans le système-expert pour chaque base - tâche dans *MACO* ;
- Le système-expert est activé pour chaque base, pour contrôler les données en utilisant les règles et identifier les erreurs intra-base - tâche dans *MECO*.

Pour mettre en œuvre l'appariement :

- Un expert détermine les objets à apparier en analysant les correspondances entre schémas et les spécifications - tâche dans *MACO* ;
- Un expert développe ou sélectionne des outils d'appariement - tâche dans *MECO* ;
- Les algorithmes d'appariement sont appliqués pour calculer les correspondances entre les données des deux bases - tâche dans *MECO* ;
- Une méthode d'évaluation des liens d'appariement est appliquée pour identifier les couples incertains - tâche dans *MECO*.

Pour mettre en œuvre le contrôle inter-bases :

- Un expert choisit une source de connaissances à exploiter pour réaliser le contrôle inter-bases ;

  Si les spécifications sont choisies :

  - Un expert identifie les connaissances utiles au contrôle en analysant les spécifications de chaque base (analyse individuelle et croisée des documents) - tâche dans *MACO*.
  - Un expert représente les connaissances sous forme de règles en adoptant l'approche par classification directe ou l'approche par prédiction,

comparaison et classification, et les introduit dans le système-expert - tâche dans *MACO* ;

- Le système-expert est activé pour contrôler tous les couples d'objets appariés et les classer en incohérences ou équivalences - tâche dans *MECO*;

Si les données sont choisies :

- Un expert adopte une approche pour organiser les connaissances dans la base de règles : classification directe ou prédiction, comparaison et classification - tâche dans *MACO* ;

  Si la classification directe est adoptée :

  - Un expert recueille des exemples d'apprentissage et attribue la classe à chaque exemple - tâche dans *MACO*;
  - Un algorithme d'apprentissage est appliqué pour apprendre des règles de classification directe - tâche dans *MACO* ;
  - Un expert valide et révise au besoin les règles apprises - tâche dans *MACO* ;
  - Un expert introduit les règles dans le système-expert - tâche dans *MACO*;
  - Le système-expert est activé pour contrôler tous les couples d'objets appariés et les classer en incohérences ou équivalences - tâche dans *MECO*;

  Si la prédiction, comparaison et classification est adoptée :

  - Un expert sélectionne des couples d'objets appariés - tâche dans *MACO*;
  - Un algorithme d'apprentissage est appliqué sur les couples (dans les deux directions) pour apprendre des règles de prédiction - tâche dans *MACO* ;
  - Un expert valide et révise au besoin les règles apprises - tâche dans *MACO* ;
  - Un expert détermine les régles de comparaison et classification - tâche dans *MACO* ;
  - Un expert introduit les règles dans le système-expert - tâche dans *MACO*;
  - Le système-expert est activé pour contrôler tous les couples d'objets appariés et les classer en incohérences ou équivalences - tâche dans *MECO*;

La méthodologie que nous proposons est synthétisée en figure 88.

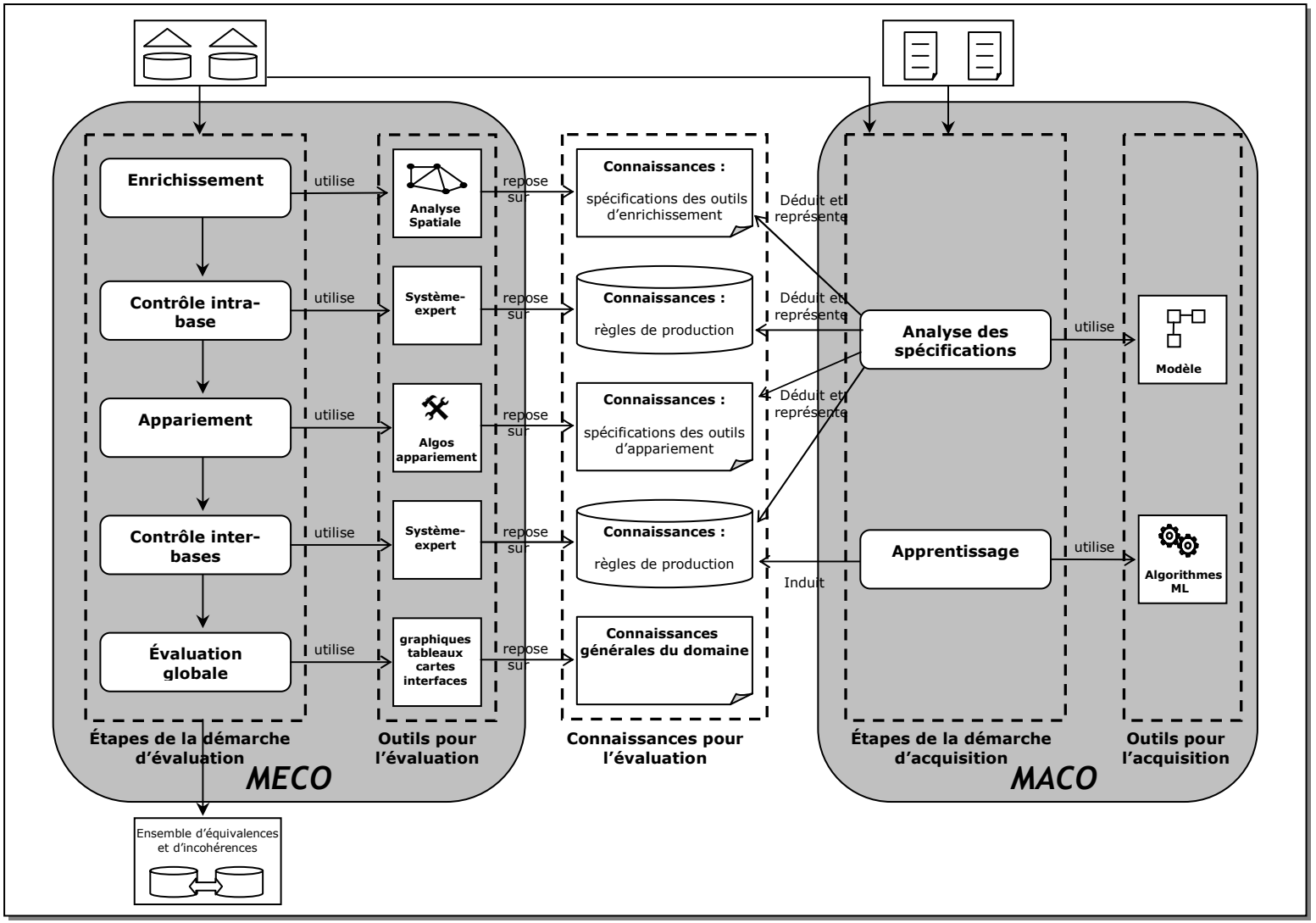


***Figure 88**. Synthèse de la méthodologie d'évaluation.*

D.7 EXPLOITATION GLOBALE DES RESULTATS

Les résultats de l'évaluation des différences sont destinés à assurer une intégration cohérente des données. En fonction de la stratégie d'intégration retenue (fusion ou multi-représentations), l'exploitation des résultats peut différer.

On peut d'abord décider de notifier les incohérences. Les données ne sont donc pas corrigées mais le système avertit l'utilisateur que certaines données sont erronées. Cette possibilité peut être envisagée pour des systèmes à représentations multiple. On peut ensuite décider de corriger les données. La correction peut être menée en fonction de la gravité des erreurs. Son coût n'est pas négligeable mais à son terme, la qualité des données sera améliorée. Par conséquent, le produit fusionné contiendra moins d'erreurs et le système multi-représentations garantira des réponses cohérentes à l'utilisateur.

Puisque la méthode *MECO* est composée d'une phase d'enrichissement, sa mise en œuvre a également pour effet d'enrichir le contenu des bases. Cela se traduit pas l'appariation de nouveaux objets, d'attributs ou de relations. Cela peut aussi concerner la structure des données (création d'une structure topologique par exemple). De ce fait, l'évaluation de la cohérence des données présente un intérêt pour la qualité des bases mais aussi pour la richesse de leur contenu.

Enfin, puisque des connaissances sur les bases sont extraites à partir des données par apprentissage automatique, il devrait être possible d'exploiter ces connaissances pour enrichir les spécifications ou les affiner, lorsque celles-ci sont trop imprécises.

CHAPITRE E

APPLICATION DE LA METHODOLOGIE D'EVALUATION DE LA COHERENCE

E.1 INTRODUCTION

Nous avons présenté dans les chapitres précédents (C et D) la méthodologie que nous proposons pour évaluer la cohérence inter-représentations. Afin d'étudier la faisabilité de notre approche, nous avons mis au point plusieurs expérimentations qui illustrent la mise en œuvre complète de *MECO* et *MACO*. Nous les présentons dans ce chapitre. La première application développée est décrite en section E.3. Il s'agit d'une étude sur la cohérence entre représentations de ronds-points de deux BD de l'IGN. La seconde application est présentée dans la section E.4. Elle est consacrée à l'évaluation de la cohérence entre bâtiments. Enfin, la dernière application fait l'objet de la section E.5. Son but est de montrer les possibilités d'utilisation de l'apprentissage automatique pour acquérir des règles de correspondance entre attributs de routes.

Toutes ces applications ont été développées dans un prototype conçu à cette fin : HÉTÉROGENE. L'architecture de ce prototype est présentée ci-dessous.

E.2 ARCHITECTURE DU PROTOTYPE HETEROGENE

Nous décrivons dans cette première partie l'architecture du système que nous avons mis au point pour mener nos expérimentations. Elle est constituée de trois éléments : la plate-forme OXYGENE, le système-expert développé à partir du moteur JESS, et un ensemble d'algorithmes d'apprentissage proposé par le logiciel WEKA.

La plate-forme de travail OXYGENE est d'abord présentée. Nous détaillons les caractéristiques du noyau ainsi que les extensions que nous avons réalisées. Nous exposons ensuite le moteur du système-expert : JESS. Nous l'avons relié à la plate-forme OXYGENE. Finalement, le logiciel WEKA est présenté. Il nous a servi à la réalisation des tests d'apprentissage.

E.2.1 PLATE-FORME OXYGENE

E.2.1.1 PRESENTATION GENERALE

L'ensemble des développements menés dans le cadre de cette thèse a été réalisé dans la nouvelle plate-forme de travail du laboratoire COGIT de l'IGN : OXYGENE [Badard et Braun 2003, Braun 2004].

Cette plate-forme a pris naissance il y a environ 4 ans. Elle a été conçue au cours de notre thèse. Nous avons participé à sa mise au point grâce aux expérimentations réalisées. Elle fut conçue pour rassembler les différentes applications de recherche développées au sein du laboratoire et implémentées dans plusieurs systèmes (plate-formes PlaGe, StratèGe et $GéO_2$), dont la dispersion favorisait la multiplication du code et limitait son utilisation. OXYGENE constitue aujourd'hui l'environnement de développement de plusieurs équipes de recherche du laboratoire COGIT et accompagne la plate-forme de généralisation cartographique automatique AGIT.

La figure 89 représente l'architecture générale d'OXYGENE. La plate-forme est fondée sur un *schéma objet* prenant en compte l'aspect géométrique, topologique et sémantique des données géographiques. Ce schéma s'appuie sur les standards développés par l'ISO et l'OpenGIS (normes 19107, 19109) et a été entièrement

implémenté en JAVA, langage orienté-objet. Il constitue le *noyau* de la plate-forme. Celui-ci est relié au SGBD relationnel ORACLE (version 9i avec l'extension spatiale) qui permet de gérer et stocker les données des différentes bases exploitées.

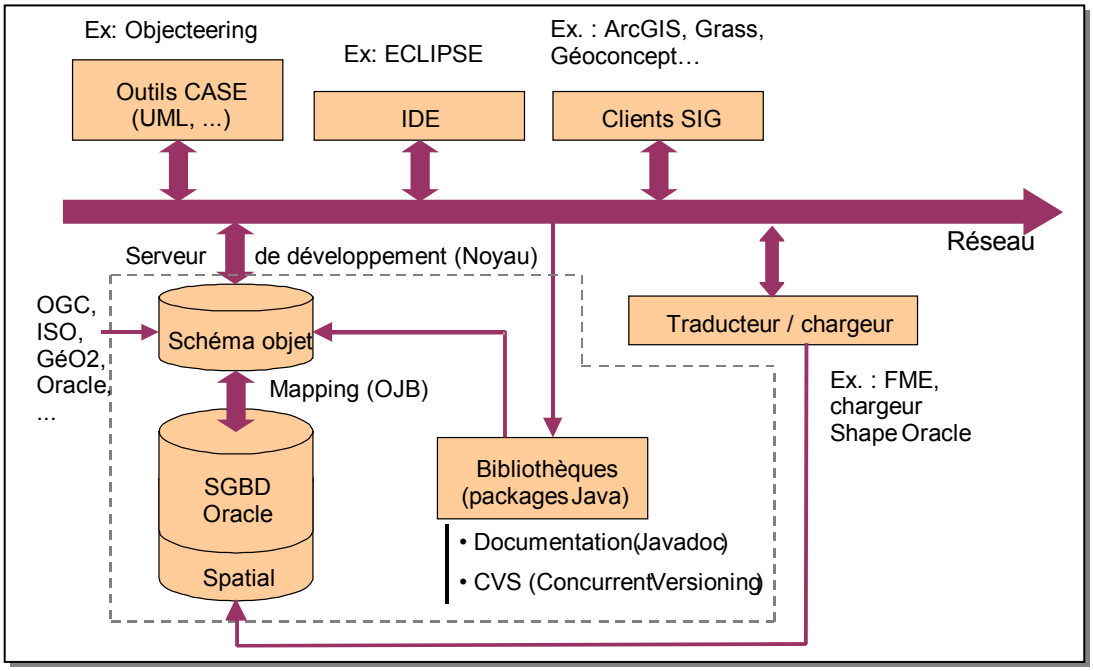


***Figure 89**. Architecture générale d'OXYGENE. (D'après [Badard et Braun 2003])*

Le lien entre le SGBD choisi et le schéma objet, ce qu'on appelle le « *mapping* », se fait à l'aide d'une bibliothèque de fonctions écrites en JAVA, nommée *OJB* (« *ObJect relational Bridge* »[29]). Les correspondances entre les tables stockées dans ORACLE et les classes correspondantes définies en JAVA sont décrites dans des fichiers XML et gérées par OJB (figure 90).

L'utilisateur de la plate-forme ne manipule pas directement les tables mais passe par l'intermédiaire des classes JAVA correspondantes. Aucune requête n'est faite directement au niveau du SGBD, elles sont masquées grâce à OJB qui ne traite que des objets Java. Cette solution offre l'avantage d'assurer une relative indépendance du noyau par rapport au SGBD utilisé. La sélection d'un autre SGBD n'implique que de faibles modifications du code JAVA.

A cette structure vient se greffer une bibliothèque d'opérateurs géométriques permettant de manipuler les données des bases et d'effectuer des analyses sur celles-ci. Un module d'appariement est également disponible de même que plusieurs algorithmes permettant la construction et le traitement de modèles numériques de terrain (MNT). Ces outils ont été développés par les chercheurs du laboratoire, au fur et à mesure de leurs besoins. Certains opérateurs ont également été récupérés de travaux extérieurs. C'est le cas de la bibliothèque *JTS Topology Suite*[30] par exemple qui offre une série de fonctions géométriques simples codées en JAVA (calcul de longueur, de superficie, d'intersection, de zone tampon,…). La liste des opérateurs continue de s'enrichir et les algorithmes que nous avons développés sont intégrés à celle-ci.

[29] OJB est disponible en accès libre sur le site : http://db.apache.org/ojb/index.html

[30] JTS peut être téléchargé sur le site http://www.vividsolutions.com/jts/jtshome.htm

***Figure 90**. Les correspondances entre les tables ORACLE dans lesquelles sont stockées les données et les classes JAVA correspondantes sont déclarées dans une série de fichiers XML, gérés par OJB.*
(D'après [Badard et Braun 2003])

Tout le code de la plate-forme est documenté à l'aide de la *Javadoc*, mécanisme qui permet de générer automatiquement de la documentation sur le code JAVA développé à partir de commentaires insérés dans celui-ci. Ce code est par ailleurs partagé à l'aide d'un CVS (« *Concurrent Versionning System* »). Ce système se charge de centraliser le code des multiples développeurs sur un seul serveur et de prendre en compte les mises à jour effectuées sur eux tout en gardant l'historique.

D'autres composants sont également rattachés à la plate-forme, outre l'interface de développement intégré ECLIPSE. Un atelier de génie logiciel est ainsi associé pour permettre de modéliser les schémas des bases de données et la structure des programmes d'applications (OBJECTEERING). Un traducteur de données permettant d'importer dans ORACLE des données stockées dans des formats de SIG commerciaux est également utilisé (FME). Enfin, une interface de visualisation des données est couplée à la plate-forme pour assurer la représentation graphique des données. Celle-ci est issue d'un projet de développement mené par le centre de géo-informatique de l'université de Leeds. Il s'agit de *GeoTools*[31].

[31] GeoTools est disponible sur le site : http://docs.codehaus.org/display/GEOTOOLS/Home.

E.2.1.2 NOYAU

Comme nous l'avons indiqué, le schéma objet du noyau de la plate-forme s'appuie sur les travaux de normalisation de l'ISO et les spécifications de l'OGC. Il s'organise ainsi en différents *packages*[32]. Ceux qui nous intéressent sont les suivants :

- Le package *Spatial* (norme 19107) : il contient différents sous-packages dans lesquels figurent les classes relatives aux primitives géométriques et topologiques de base.
- Le package *Geoschema* (d'après la norme 19109) : il contient les différents schémas définis par l'utilisateur. Il peut s'agir des schémas de bases de données géographiques (cf. E.2.1.3). Il peut également contenir des schémas définis pour des applications particulières (comme un schéma relatif à l'élaboration d'une triangulation de Delaunay par exemple).

D'autres packages sont définis dans les normes ISO associées mais celles-ci n'ont pas été implémentées dans la plate-forme. Il s'agit du modèle des métadonnées (norme 19115), du modèle qui traite des systèmes de coordonnées et de projections (norme 19111) et du modèle se rapportant à la définition des types de données (norme 19103).

Dans le schéma objet de la plate-forme, la classe mère (abstraite) relative à des classes d'objets géographiques s'appelle *FT_Feature* [OpenGIS 1999]. Toute classe appartenant à un schéma d'une BD géographique en hérite (ex : *Route*, *Chemin*, *Bâtiment*,…). Elle est reliée à une classe inclue dans le package *Spatial* : la classe *GM_Object* qui permet d'associer une géométrie aux objets géographiques. Il existe également la classe *Element_CarteTopo* qui concerne la topologie (figure 91).

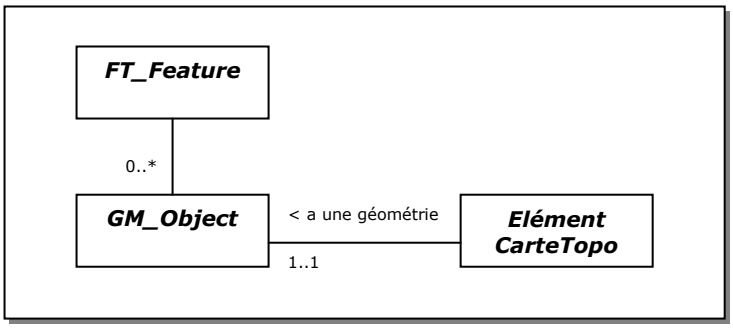


***Figure 91**. FT_Feature constitue la classe mère de tout schéma géographique. Elle est associée à une géométrie (GM_Object). La classe Element_CarteTopo constitue la classe mère de la carte topologique.*

La classe *GM_Object* se spécialise en d'autres classes correspondant aux primitives géométriques de base (point, ligne, polygone) ou à des agrégats de primitives. La classe *Element_CarteTopo* se spécialise également pour former une structure de carte topologique, inspirée de [David et al. 1993a].

La structure de carte topologique définie dans la plate-forme est représentée en figure 92 [Mustière et Bonin 2003]. Elle est composée de différentes classes : *Arc*, *Noeud*, *Face, Groupe*. Les relations entre les classes sont modélisées. La topologie Arc/Face (propriété de contiguïté) et Arc/Nœud (propriété de connexité) est ainsi prise en compte. Cette structure n'est pas systématiquement instanciée. Elle l'est au

[32] Le terme *package* est issu de la terminologie UML (et plus généralement du vocabulaire orienté-objet). Il correspond à un regroupement d'éléments de modélisation qui constitue généralement une partie du diagramme de classes global.

besoin, si le schéma de la base de données géographiques stockée ou l'application développée l'exigent.

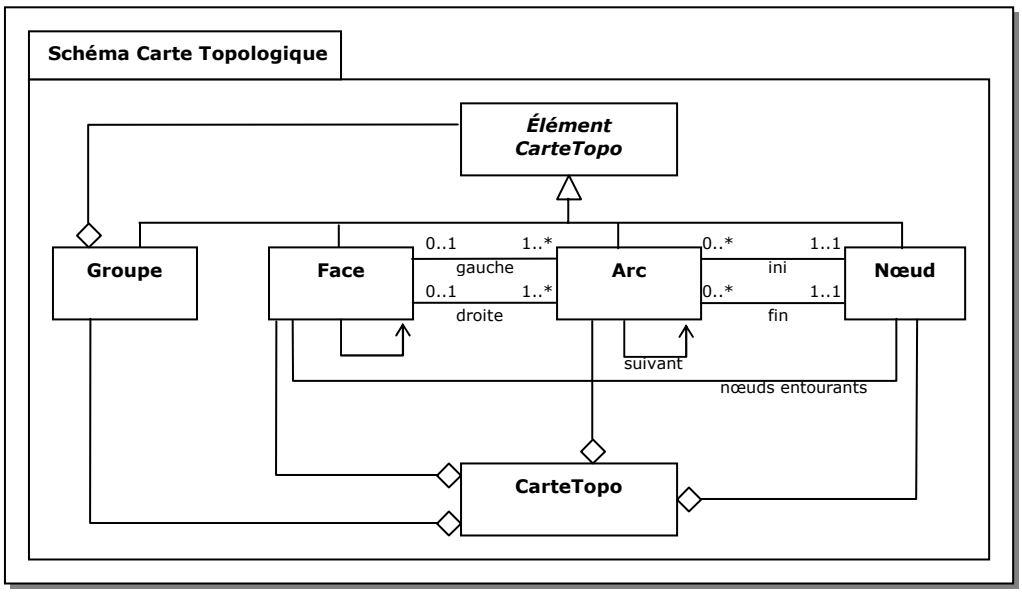


***Figure 92**. Schéma de la carte topologique définie dans OXYGENE.*
(D'après [Mustière et Bonin 2003])

A ces éléments viennent s'ajouter trois autres classes : *FT_FeatureCollection*, *Population* et *DataSet*. La première représente un agrégat de *FT_Feature* qui peut porter des méthodes d'indexation spatiale par exemple. La classe *Population* est une *FT_FeatureCollection* particulière qui contient tous les éléments d'une classe *FT_Feature* et qui possède une relation avec la classe *DataSet*. Un objet « *population_route* » par exemple représente l'ensemble des instances de la classe *Route*, laquelle correspond à une classe *FT_Feature* spécialisée. La classe *DataSet* modélise quant à elle un jeu de données correspondant à une agrégation de populations. Il peut s'agir de la base entière, d'une portion de celle-ci ou d'un thème particulier. Grâce à une relation d'agrégation récursive, des liens entre jeux de données peuvent être définis. Un *Dataset* peut ainsi être décomposé en plusieurs *Datasets* (décomposition d'un jeu de données relatif à une région particulière en plusieurs thèmes).

L'organisation des classes *Dataset, Population* et *FT_FeatureCollection* est représentée en figure 93.

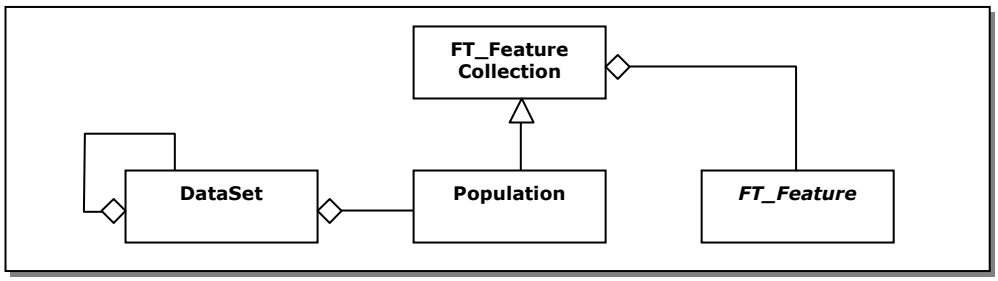


***Figure 93**. Classes de base du schéma objet du noyau.*

E.2.1.3 SCHEMAS DES BASES DE DONNEES GEOGRAPHIQUES

C'est sur ces classes générales que viennent se greffer les schémas des bases de données géographiques que l'on souhaite utiliser. On définit pour chaque base une classe *ElementBDG* qui hérite de la classe *FT_Feature.* La classe *ElementBDG* représente la classe mère pour toutes les classes de la BDG à modéliser. On définit également une classe *JeuDeDonnéesBDG* qui hérite de *DataSet* de manière à reconstituer si on le souhaite les thèmes de la BDG (routier, administratif, hydrographique,…) ou à créer un objet qui représente l'ensemble des éléments de la BDG (figure 94).

Nous avons participé à la définition des schémas de trois bases de données de l'IGN dans la plate-forme. Il s'agit de la BDCarto, Géoroute et la BDPays (ancienne BDTopo).

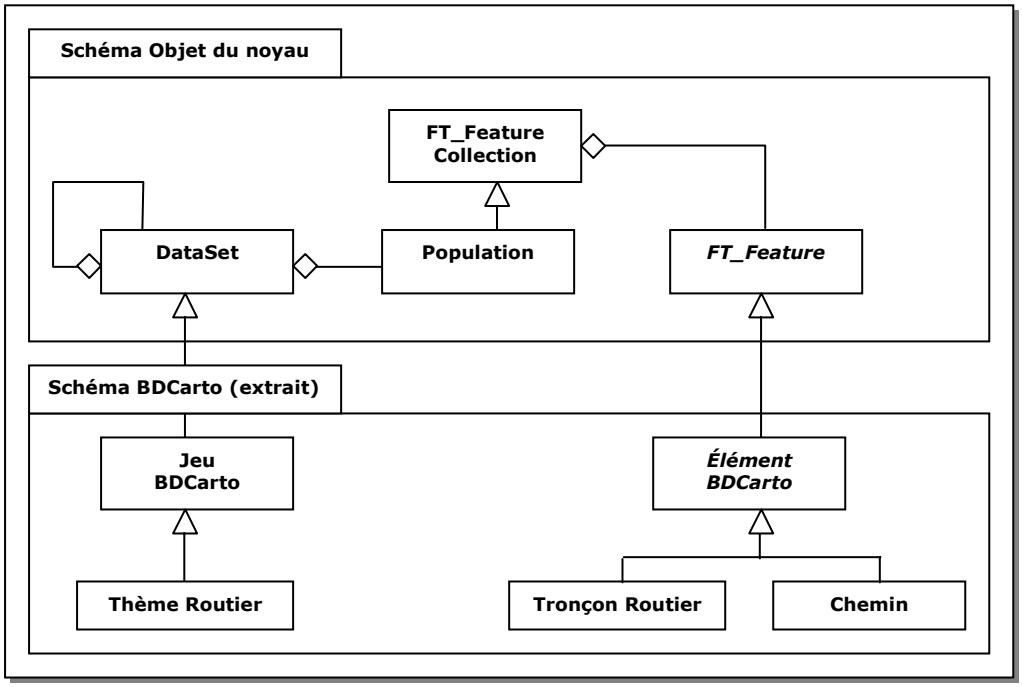


***Figure 94**. Relation existant entre le schéma du noyau et un des schémas des BD géographiques de l'IGN, la BDCarto.*

Précisons que s'il existe un seul schéma pour chaque base de données géographiques, la structuration des données dans ORACLE peut prendre diverses formes en fonction du format dans lequel les données ont été initialement enregistrées. Pour les BDG de l'IGN implémentées dans la plate-forme, deux structures existent aujourd'hui. La première est la structure *shape*. Elle s'applique aux données enregistrées selon le modèle imposé par le format *shapefile* d'ESRI, traduit en JAVA. C'est une structure assez pauvre dans laquelle aucune relation entre classes n'est représentée. La deuxième structure correspond au schéma de la base tel qu'il est définit dans les spécifications des BDG. Nous l'appelons *schéma structuré.* C'est une structure beaucoup plus riche que la précédente car elle tient compte cette fois des relations entre classes d'objets (figure 95). Les données enregistrées dans des tables relationnelles ORACLE sont donc décrites selon une de ces deux structures.

Les différentes formats dans lequel les données de l'IGN sont disponibles nous ont conduit à participer au développement d'un chargeur et d'un traducteur de données dans OXYGENE. Ceux-ci permettent d'importer des données stockées dans ORACLE selon la structure *shape,* dans des classes JAVA (et par conséquent dans des tables

relationnelles ORACLE) qui respectent la structure du *schéma structuré* [Mustière 2003]. La traduction implique de dupliquer les objets et leurs attributs et de récréer les relations entre les instances des différentes classes par analyse géométrique.

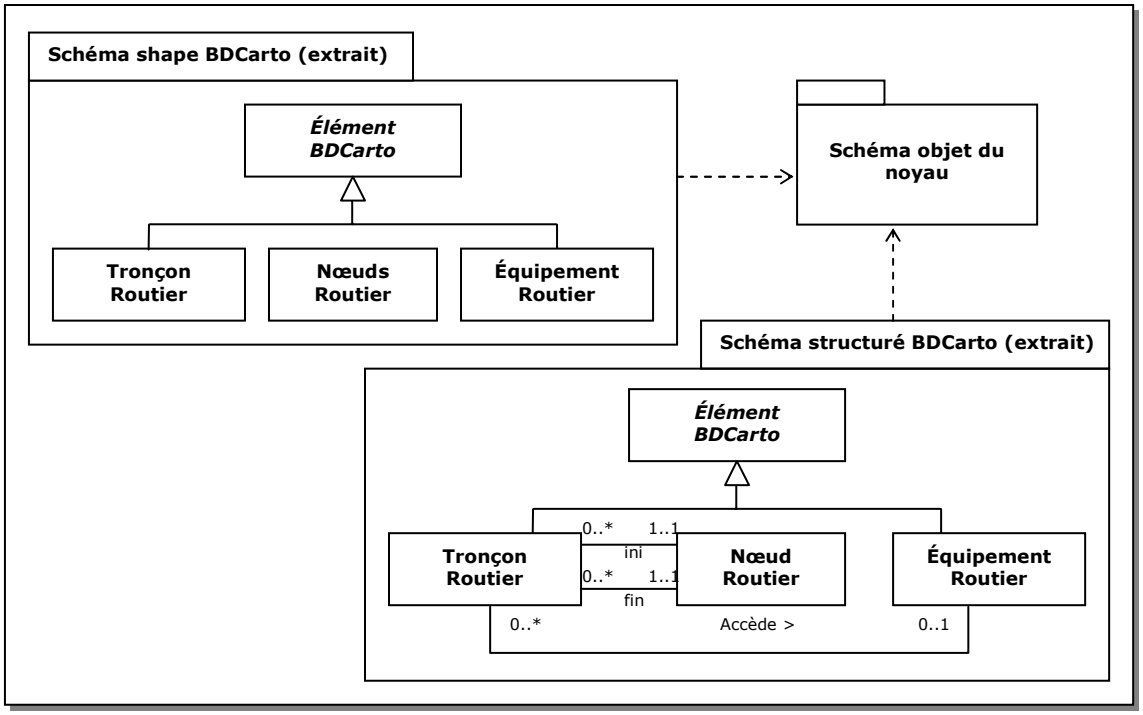


***Figure 95**. Les données dans OXYGENE peuvent être structurées différemment selon le modèle imposé par le format dans lequel elles étaient initialement enregistrées.*

E.2.1.4 SCHEMAS APPLICATIFS

En plus des schémas des BDG, on peut vouloir organiser des données selon un schéma propre à une application développée. Dans ce cas, on définit un schéma applicatif qui fait partie du package *GeoSchema* et dont les classes héritent également de *FT_Feature* ou de ses classes filles. Si on souhaite rajouter des attributs particuliers aux classes d'une BDG par exemple ou que l''on souhaite enrichir les données de nouveaux objets, on peut créer de nouvelles classes correspondantes qui spécialisent les classes de la BDG initiale. C'est de cette manière que nous avons procédé pour développer nos applications. Tous les attributs et les nouvelles classes créées pour mettre en œuvre la méthode d'évaluation *MECO* ont été définies dans un package particulier qui contient le *schéma enrichi* des bases. Un extrait du schéma enrichi de la BDCarto est donné en figure 96.

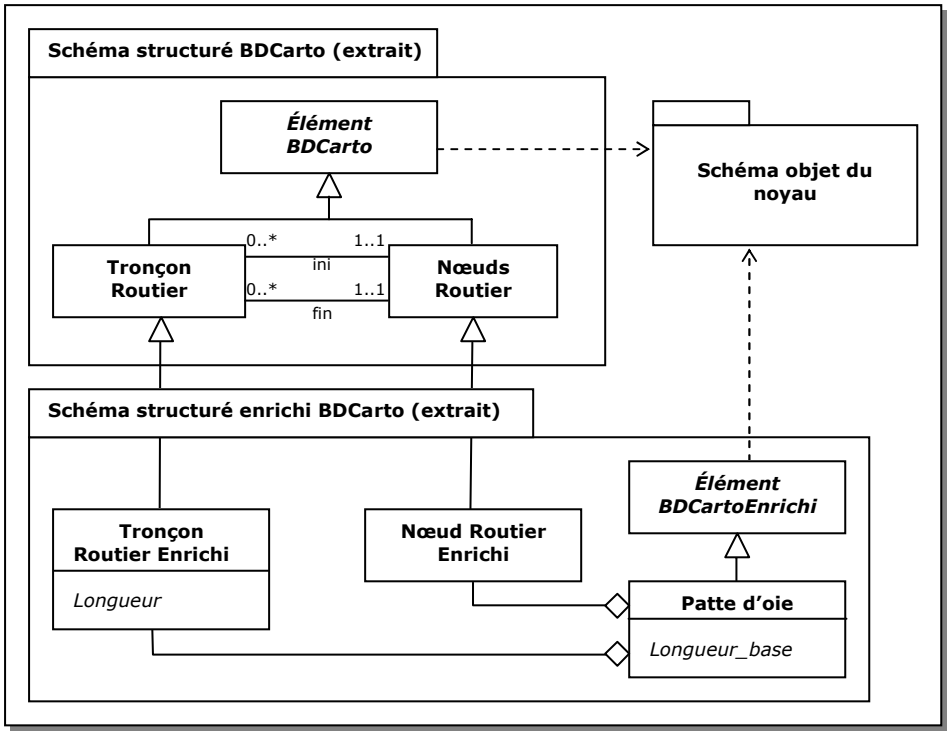


***Figure 96**. Le schéma structuré enrichi correspond à un schéma applicatif définit dans le cadre d'une application particulière.*

De la même manière, la carte topologique que nous avons présenté au paragraphe E.2.1.2. a été spécialisée pour nos besoins (figure 97). Son enrichissement a nécessité de créer un package particulier qui contient les classes *NœudÉvaluatio*n, *ArcÉvaluation*, *FaceÉvaluation* et *GroupeÉvaluation* sur lesquelles ont été définies différentes méthodes. Ces classes héritent de la carte topologique initiale. La structure enrichie a notamment été utilisée pour détecter les ronds-point dans un réseau routier (cf. E.3.) et pour mettre au point la méthode de *clustering* fondée sur le calcul d'une triangulation de Delaunay (cf. E.4.).

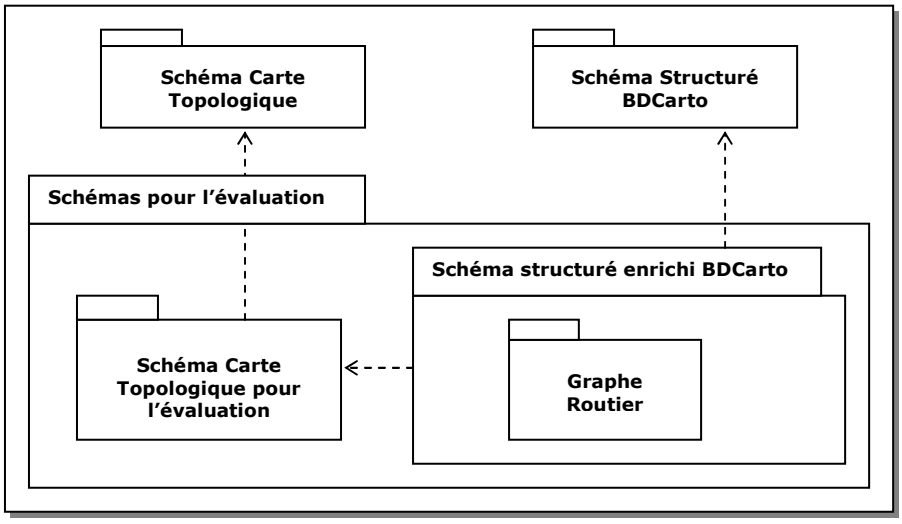


***Figure 97**. Organisation des schémas applicatifs définis pour mettre en œuvre nos expérimentations (exemple de la BDCarto).*

Maintenant que nous venons de donner un aperçu général de l'organisation des données dans notre environnement de travail, nous allons présenter un second

élément de l'architecture de notre prototype : le système-expert fondé sur le moteur JESS.

E.2.2 SYSTEME-EXPERT ET MOTEUR JESS

JESS[33] est l'acronyme de *JAVA Expert System Shell*. C'est un outil de production de système-expert entièrement développé en JAVA par [Friedman-Hill 2003] dans les laboratoires Sandia (Canada). Il est inspiré de CLIPS[34] écrit en langage C et s'appuie sur l'algorithme RETE [Forgy 1982] pour réaliser les inférences. C'est ce système que nous avons utilisé pour automatiser le raisonnement d'évaluation de la cohérence.

L'utilisation de JESS peut prendre plusieurs formes. Dans le cas du développement d'un système-expert, il peut être utilisé de manière autonome, sans l'associer à des programmes JAVA existants. Dans ce cas, les faits et les règles constituant la base de connaissances du système sont entièrement écrits sous forme de scripts à l'aide du langage JESS (une variante du langage LISP). Mais JESS peut également être couplé à des programmes d'application JAVA grâce à l'existence d'une API (« *Application Programming Interface* »). Ceci est particulièrement intéressant puisque cela nous a permis de le relier facilement à la plate-forme OXYGENE. Plusieurs niveaux d'imbrications sont d'ailleurs possibles. Dans notre cas, nous avons choisi de d'initialiser le processus d'inférence à partir d'une application JAVA développée dans la plate-forme. Seules les règles sont directement écrites dans le langage de scripts JESS. Les faits initialement stockés dans les tables ORACLE sont fournis au système-expert grâce à méthodes proposées par l'API. Une fois les règles appliquées sur ces faits, les résultats (conclusions) sont transformés en instances de classes JAVA et stockés dans la base.

La dernière version de JESS (7.0) offre un « *plug-in* » qui peut être ajouté à ECLIPSE, l'interface de développement que nous utilisons pour nos applications JAVA.

E.2.3 LOGICIEL WEKA

Le troisième module de l'architecture du prototype HÉTÉROGENE est le logiciel WEKA développé par l'université Waikato de Nouvelle-Zélande [Witten et Frank 1999][35]. Ce logiciel regroupe un ensemble d'algorithmes d'apprentissage (supervisé et non supervisé) également écrits en JAVA.

Nous avons fait le choix d'utiliser ce logiciel pour deux raisons. La première est qu'il propose tous les algorithmes d'apprentissage supervisé symboliques dont nous avons besoin. Nous avons essentiellement utilisé les versions proposées des algorithmes C4.5. [Quinlan 1993] et RIPPER [Cohen 1995]. Nous en avons déjà discuté au chapitre précédent (D.4.2.1.). La seconde raison est que ces algorithmes peuvent être utilisés par l'intermédiaire d'une API dans un application existante développée en JAVA. Le logiciel peut donc être facilement couplé à OXYGENE. WEKA offre également une interface graphique conviviale et c'est essentiellement par son intermédiaire que nous avons exploité les algorithmes (figure 98). Le logiciel présente en outre plusieurs fonctions très pratiques pour mettre en œuvre des tests

[33] JESS est disponible sur le site : http://herzberg.ca.sandia.gov/jess/index.shtml

[34] CLIPS est disponible sur le site : http://www.ghg.net/clips/CLIPS.html

[35] Le logiciel peut être téléchargé sur le site : http://www.cs.waikato.ac.nz/~ml/weka/

d'apprentissage. Il permet notamment de filtrer automatiquement des descripteurs et des exemples ou de discrétiser des valeurs numériques.

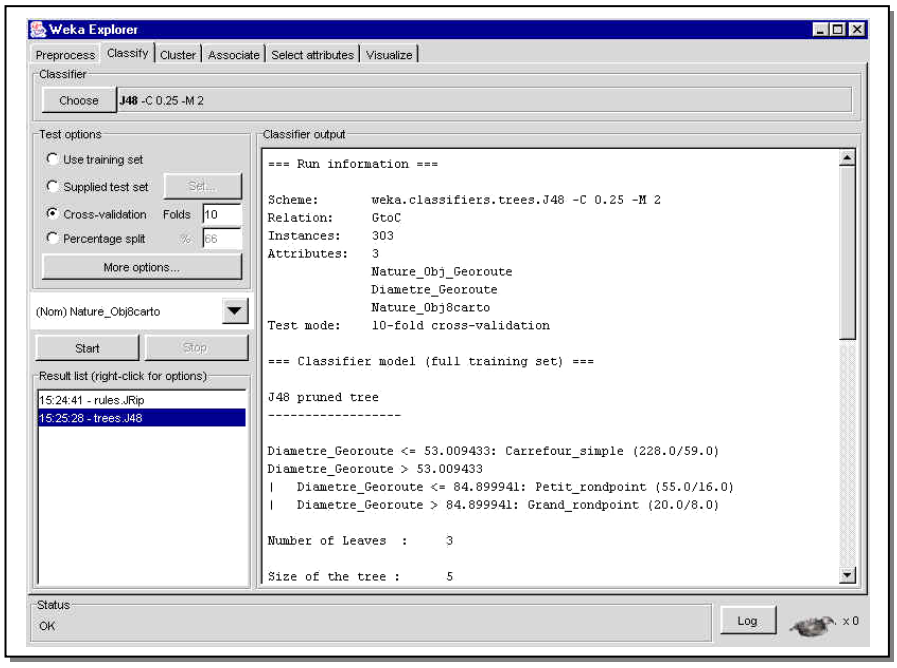


***Figure 98**. Interface du logiciel WEKA destiné à l'apprentissage automatique.*

E.2.4 ARCHITECTURE COMPLETE DU PROTOTYPE HETEROGENE

L'architecture complète de notre prototype peut maintenant être illustrée (figure 99). Elle comprend la plate-forme OXYGENE qui se compose d'un noyau et de schémas de bases de données géographiques de l'IGN sur lesquels se greffent les schémas applicatifs que nous avons définis pour nos expérimentations, ainsi qu'une bibliothèque d'algorithmes. Le système-expert fondé sur le moteur JESS est relié à cette plate-forme par l'intermédiaire d'une API Java. Le logiciel WEKA qui offre des outils d'apprentissage automatique est également associé.

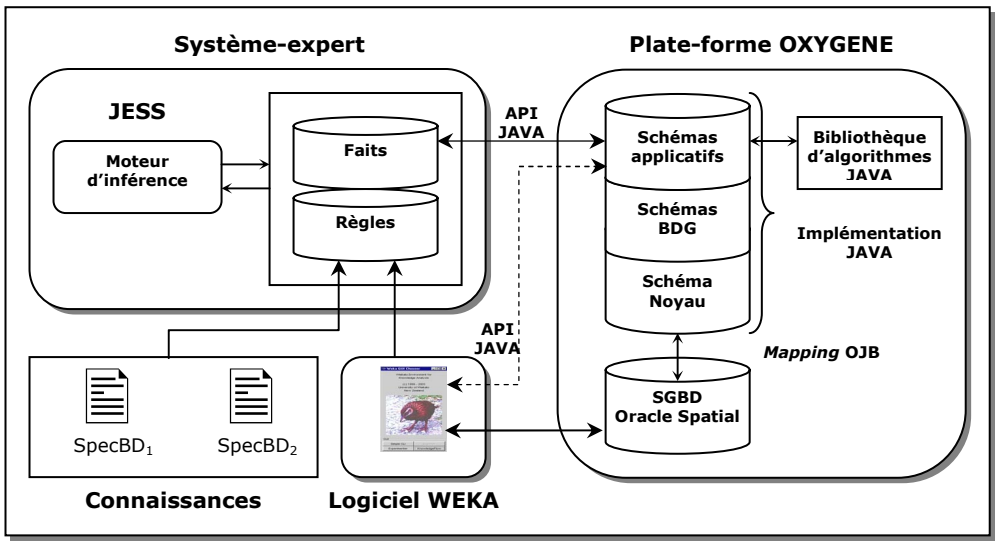


***Figure 99.** Architecture complète du prototype HÉTÉROGENE.*

E.3 ÉTUDE DES DIFFERENCES ENTRE REPRESENTATIONS DE RONDS-POINTS

E.3.1 MOTIVATIONS

La première étude que nous avons menée pour étudier la faisabilité de notre méthodologie concerne les différences entre ronds-points de deux bases de données de l'IGN : la BDCarto et Géoroute. Les ronds-points nous ont semblé particulièrement intéressants à étudier car ils permettent de bien illustrer chaque étape des méthodes *MECO* et *MACO* dans un niveau de complexité suffisamment élevé. Nous détaillons nos expérimentations dans les sections suivantes. Nous présentons au préalable les caractéristiques des bases utilisées.

E.3.2 PRESENTATION DES BASES

Les données utilisées sont issues de bases présentant des niveaux de détail différents mais de même actualité. La BDCarto est une base vectorielle de résolution décamétrique. Elle est constituée d'informations géographiques nécessaires aux activités de réfection et de révision des séries cartographiques IGN à partir du 1:100.000 (TOP100). Elle peut être utilisée pour effectuer des analyses au niveau régional et départemental. Les éléments de cette base proviennent initialement de deux sources différentes : des images SPOT pour l'occupation du sol et des cartes au 1/50.000 pour le reste des objets. Nous donnons un extrait de données en figure 100.

Géoroute est une base vectorielle de résolution métrique en agglomération (figure 100). C'est un produit destiné principalement à la navigation routière et qui n'a pas une vocation cartographique. Il contient une description exhaustive des voies de circulation qui peut servir pour des calculs d'itinéraires ou du géocodage. Les sens de circulation sont ainsi renseignés de même que les restrictions de circulation, les noms des rues et les bornes postales. Géoroute ne se distingue de la BDCarto qu'en milieu urbain, c'est-à-dire pour des agglomérations de plus de 100.000 habitants (seuil abaissé à 10.000 en Île-de-France) recensées par l'INSEE (recensement général de 1990). En milieu inter-urbain, les données de Géoroute sont celles de la BDCarto. Les spécifications des données varient donc à l'intérieur de la zone couverte par le produit et chaque objet porte un attribut indiquant à quelle source il se rapporte (« urbain » ou « inter-urbain »). Pour nos expérimentations, seuls les ronds-points en milieu urbain ont été retenus.

Notre zone d'étude se situe dans le département 77 (Seine-et-Marne). Les jeux de données sélectionnés couvrent une superficie d'environ 3650 km². Pour Géoroute, cela représente approximativement 45.000 objets de la classe *Tronçon Routier* contre 14.200 pour la BDCarto.

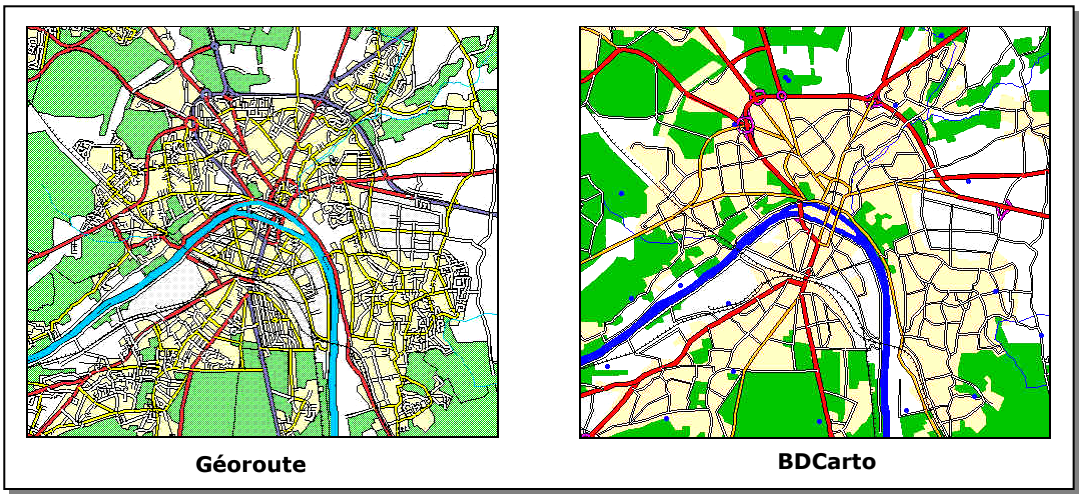


***Figure 100**. Extrait des jeux de données issus des bases de données de l'IGN utilisés pour mener les expérimentations sur les ronds-points.*

E.3.3 ANALYSE DES SPECIFICATIONS

Pour comprendre ce que contiennent les bases de données et déterminer les différences de contenu et de modélisation des objets ronds-points, la première tâche réalisée fut l'analyse des spécifications. C'est la première étape de la méthode *MACO* (cf. D.3.1.).

E.3.3.1 SPECIFICATIONS DE LA BD GEOROUTE

Dans un premier temps, nous avons étudié les documents concernant Géoroute [Géoroute 1999, Géoroute 2000]. Nous avons constaté que les informations relatives aux ronds-points apparaissaient dans deux classes différentes : *Nœud Routier* et *Carrefour Complexe*. Un nœud routier (objet ponctuel) correspond à une extrémité de tronçon routier. Il traduit un carrefour ou une modification des conditions de circulation. Un attribut « nature de l'intersection » précise la nature du nœud routier : il peut s'agir d'une 'intersection simple', d'un 'rond-point simple', d'une 'barrière à péage', d'un 'franchissement', d'un 'nœud d'accès' ou d'un 'nœud à changement de valeur d'attribut'. Un 'rond-point simple' correspond à un « *endroit de l'espace routier où les routes se rejoignent au même niveau, de forme non exclusivement circulaire, possédant un terre-plein central infranchissable et ceinturé par une chaussée à sens unique. Les véhicules ne s'y croisent pas* » [Géoroute 1999, p. 29]. Un rond-point peut donc être représenté par une instance de la classe *Nœud Routier* dont l'attribut « nature de l'intersection » a pour valeur 'rond-point simple'. Sa modélisation est ponctuelle.

Mais une seconde classe se rapporte aux ronds-points, la classe *Carrefour Complexe*, qui n'existe qu'en zone urbaine. Un carrefour complexe (objet surfacique) représente un « *endroit de l'espace routier où les routes se rejoignent ou se coupent au même niveau ou à des niveaux différents... Les carrefours complexes sont des Zones à Trafic Non Structuré (ZTNS), des grands ronds-points ou des carrefours aménagés. Le carrefour complexe possède une emprise totale minimale de 30 mètres de rayon. Si le carrefour a une emprise inférieure, il est traité en intersection simple (nœud du réseau routier)* » [Géoroute 1999, p. 31]. Cette classe possède un attribut « nature du carrefour » qui peut prendre la valeur 'rond-point' définie comme suit :

« *carrefour de forme non exclusivement circulaire, possédant un terre-plein central infranchissable et ceinturé par une chaussée à sens unique. Les voitures ne s'y croisent pas* ». On retrouve la définition proposée pour les nœuds routiers. Un rond-point peut donc également être représenté par une instance de la classe *Carrefour Complexe* dont l'attribut « nature du carrefour » a pour valeur 'rond-point'. Sa modélisation est cette fois surfacique. L'existence de cet objet dans la BD est conditionnée par la taille de son emprise sur le terrain : elle doit être supérieure à 30m. Dans le cas contraire, c'est un objet ponctuel nœud routier qui est saisi.

Il existe cependant une incohérence concernant la taille de l'emprise. Dans la classe *Carrefour complexe*, il semble que le seuil de 30m concerne le *rayon* de l'entité. Cependant, dans la classe *Nœud Routier*, les spécifications indiquent qu'il s'agit de « *toute portion de l'espace routier symbolisant un choix d'au moins trois directions et de diamètre inférieur à 30m lorsqu'on l'assimile à un cercle* » [Géoroute 1999, p. 29]. Il semble donc cette fois que le seuil de 30m se rapporte au *diamètre* de l'entité.

Plusieurs questions se posent :

- Lorsqu'un rond-point est détaillé, c'est-à-dire qu'un objet carrefour complexe est créé, quelle doit être la valeur de l'attribut « nature de l'intersection » des nœuds routiers constitutifs du rond-point ?
- Existe-t-il réellement dans les données des carrefours complexes correspondant à des ronds-points de forme non circulaire ?
- S'agit-il du diamètre ou du rayon de l'entité qui conditionne la saisie d'un carrefour complexe ?
- Que saisit-on lorsqu'on crée un carrefour complexe ? S'agit-il de l'axe de la chaussée ? Du contour extérieur ?

Pour répondre aux deux premières interrogations, nous avons visualisé les données. Nous avons remarqué qu'il existait effectivement des ronds-points détaillés de forme non circulaire, ce qui est néanmoins assez rare. Nous avons également constaté que l'attribut « nature de l'intersection » des nœuds routiers appartenant à des ronds-points détaillés prenait la valeur 'intersection simple' et non 'rond-point simple'.

Les questions suivantes doivent également être éclaircies mais les spécifications de contenu ne suffisent plus pour y répondre et l'analyse interactive des données ne peut aider à les résoudre. Nous avons donc cette fois décidé de prendre connaissance des spécifications de saisie qui sont destinées aux personnes chargées de la production de la base. Elles ne sont jamais fournies aux utilisateurs. Nous avons pu tirer quelques informations supplémentaires au sujet des ronds-points.

D'abord, dans la classe *Nœud Routier*, les spécifications nous ont indiqué que « *les ronds-points permettant de faire demi-tour en bout d'impasses ne sont pas saisis en 'rond-point simple' mais en 'intersection simple'* » [Géoroute 2000]. Cette information est naturellement précieuse pour l'étude de la cohérence, d'autant plus que la construction de ronds-points en bout d'impasses est relativement fréquente dans les nouveaux lotissements. Ensuite, dans la classe *Carrefour Complexe*, il est clairement indiqué que l'emprise minimale de l'entité doit être de 15m de rayon pour créer l'objet. Cela correspond aux 30m de diamètre annoncés dans les spécifications de contenu des nœuds routiers. Nous considérons ainsi que l'incohérence est levée. Les spécifications indiquent encore que le rond-point est caractérisé par un sens giratoire obligatoire. Enfin, il est précisé que le rayon doit être « *mesuré depuis le centre jusqu'à la limite extérieure de la chaussée* » et que « *le rond-point surfacique*

est créé en s'appuyant sur les tronçons qui le forment » [Géoroute 2000]. Cela signifie que la mesure du diamètre pour déterminer la modélisation du rond-point ne s'appuie pas sur les mêmes éléments du monde réel que la saisie effective de l'objet dans la base. D'un côté, le diamètre est mesuré jusqu'aux limites de la chaussée. De l'autre, la saisie s'appuie sur les tronçons de route qui forment le rond-point, ses tronçons étant définis par l'axe des chaussées. Un rond-point dans les données de diamètre égal à 25m devrait ainsi correspondre à une entité sur le terrain de diamètre égal à 30m environ d'emprise totale (25m + 2 x largeur d'une chaussée, soit 2 x 2,5m environ). Un rond-point de 25m dans les données est donc en fait conforme aux spécifications, contrairement à ce qui pourrait être cru en lisant seulement les spécifications de contenu. Nous résumons les spécifications en figure 101.

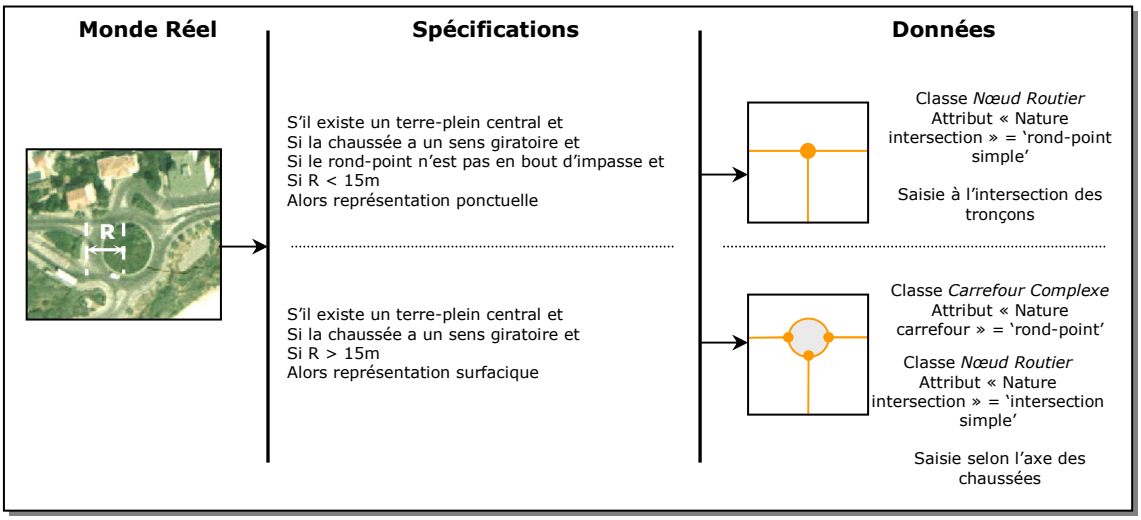


***Figure 101**. Spécifications relatives aux ronds-points dans Géoroute.*

Les règles de saisie que nous venons d'énoncer montrent bien que l'analyse des spécifications n'est pas une tâche évidente. Les règles peuvent être disséminées dans différentes classes, il peut exister des erreurs dans leur description et les spécifications de contenu ne sont pas toujours suffisamment exhaustives. Les spécifications de saisie sont plus riches mais ne contiennent pas non plus toute l'information requise. Une analyse interactive des données peut aider à clarifier certaines interrogations. Nous verrons que l'apprentissage automatique constitue aussi un outil précieux pour y répondre.

E.3.3.2 SPECIFICATIONS DE LA BDCARTO

Pour la BDCarto, les spécifications relatives aux ronds-points sont essentiellement définies dans la classe *Nœud Routier* [BDCarto 2001].

La classe *Nœud Routier* représente les extrémités d'un tronçon de route ou de bac. Un nœud routier (géométrie ponctuelle) traduit un carrefour ou une modification des conditions de circulation. Les spécifications de contenu indiquent qu'il n'y a pas à proprement parler de sélection de ces objets : ils sont déduits à partir de la sélection des tronçons de route et des liaisons maritimes ou des bacs. La classe nœud routier possède un attribut « Type » qui peut prendre plusieurs valeurs. Trois d'entre elles nous intéressent : 'carrefour simple', 'petit rond–point' et 'grand rond-point'. Un 'carrefour simple' peut correspondre à une intersection simple, un cul-de-sac, un carrefour aménagé d'une extension inférieure à 100m ou un rond-point d'un diamètre

inférieur à 50m. On ne distingue donc pas les ronds-points des autres types d'intersection en-deçà d'une emprise de 50m de diamètre. Les ronds-points plus importants sont, par contre, individualisés. La valeur 'petit rond-point' est attribuée si le diamètre est compris entre 50 et 100m. Au delà de 100m, c'est la valeur 'grand rond-point' qui est attribuée et l'objet a deux représentations : une détaillée et une simplifiée. La représentation détaillée est mesurée et saisie d'axe à axe. Cela signifie qu'un rond-point détaillé dans les données de diamètre égal à 95m n'est pas conforme. Il n'existe pas de décalage entre ce qui est saisi et la règle de sélection de l'objet.

La représentation simplifiée du rond-point est déduite de l'intersection du prolongement des tronçons de route. Les tronçons compris entre les représentations détaillées et simplifiées sont qualifiés de tronçons « logiques ». Ils sont fictifs et ont pour seul but de matérialiser la logique de communication (figure 102). Les tronçons constitutifs du rond-point détaillé doivent prendre la valeur 'bretelle' pour l'attribut « vocation de la liaison ».

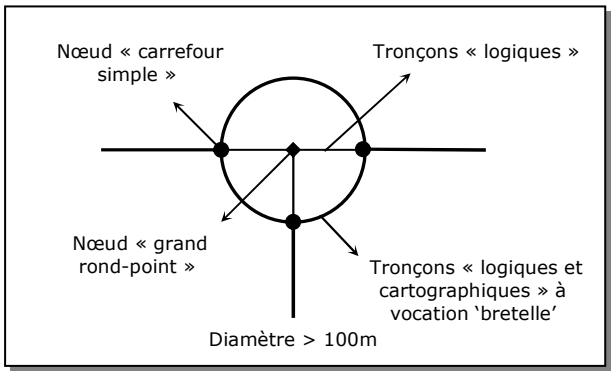


***Figure 102**. Un rond-point de diamètre supérieur à 100m a deux représentations dans la BDCarto : une généralisée et une détaillée.*

Dans cette base, la représentation détaillée du rond-point n'est donc pas directement accessible. Il n'existe pas de géométrie surfacique associée à l'objet. Seul un nœud routier « grand rond-point » permet de les individualiser. Puisque nous aurons à vérifier la conformité du diamètre des objets, nous devrons les recréer lors de l'enrichissement.

Il faut noter qu'il existe également une classe *Carrefour Complexe* dans la BDCarto mais cette classe n'a pas de géométrie. Un carrefour complexe est composé de nœuds routiers appartenant à un rond-point détaillé, un échangeur ou à un carrefour aménagé. Il fait donc le lien entre différents nœuds routier appartenant à une même entité. C'est un objet composé. Son existence pourrait être utile pour reconstruire les ronds-points détaillés mais il n'existe pas assez d'objets de ce type en pratique.

E.3.3.3 COMPARAISON DES SPECIFICATIONS

Nous illustrons en figure 103 les différences de modélisation concernant les ronds-points des bases Géoroute et BDCarto. Puisque les règles de saisie ne sont pas les mêmes, on peut s'attendre à des différences de représentation entre les objets homologues des deux bases. Ces différences ont été mises en évidence après une analyse croisée des spécifications, comme le préconise *MACO* (cf. D.3.1.).

En présence d'un rond-point simple ponctuel dans Géoroute, la seule représentation équivalente possible dans la BDCarto est un nœud routier dont l'attribut « type » porte la valeur 'carrefour simple'. Ensuite, lorsque la représentation est détaillée dans Géoroute, en fonction de la valeur du diamètre de l'objet, la représentation équivalente dans la BDCarto peut correspondre soit à un nœud carrefour simple (diamètre < 50m dans Géoroute), soit à un petit rond-point (diamètre compris entre 50 et 100m dans Géoroute), soit à un grand rond-point (diamètre > 100m dans Géoroute).

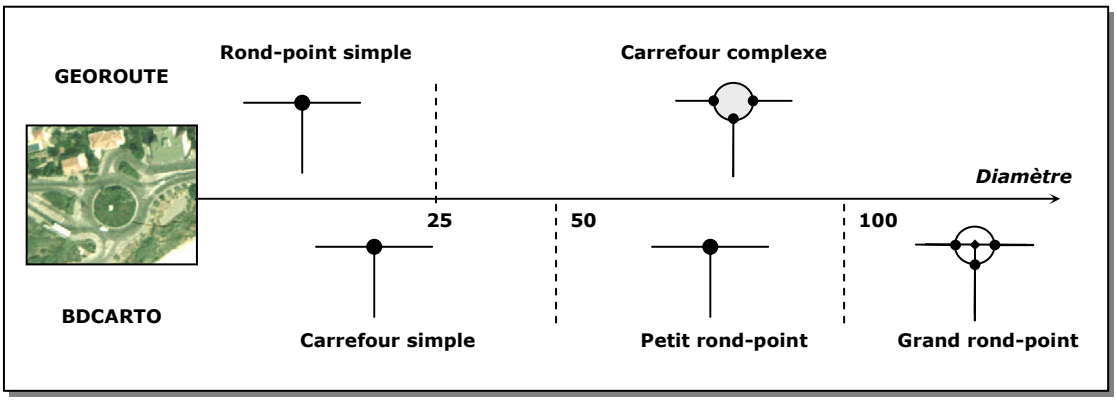


***Figure 103**. Comparaison des contraintes de modélisation des ronds-points dans la BDCarto et Géoroute.*

Puisque dans les deux bases le rond-point est saisi d'axe à axe (figure 104), la valeur du diamètre des objets homologues doit être assez proche, aux écarts de position près (la qualité de la saisie n'est pas la même dans les deux sources). Mais il faut tenir compte du fait que les éléments de référence sur lesquels s'appuie la mesure du rayon de l'entité ne sont pas les mêmes que ceux à partir desquels est saisi l'objet dans la base (cas de Géoroute). C'est ce qui explique le seuil de 25m qui sépare la représentation détaillée du rond-point dans Géoroute de la représentation simplifiée, au lieu du seuil de 30m. Cette différence de 5m est définie arbitrairement mais est fixée en tenant compte de la largeur normalisée d'une chaussée qui correspond à 2,5m.

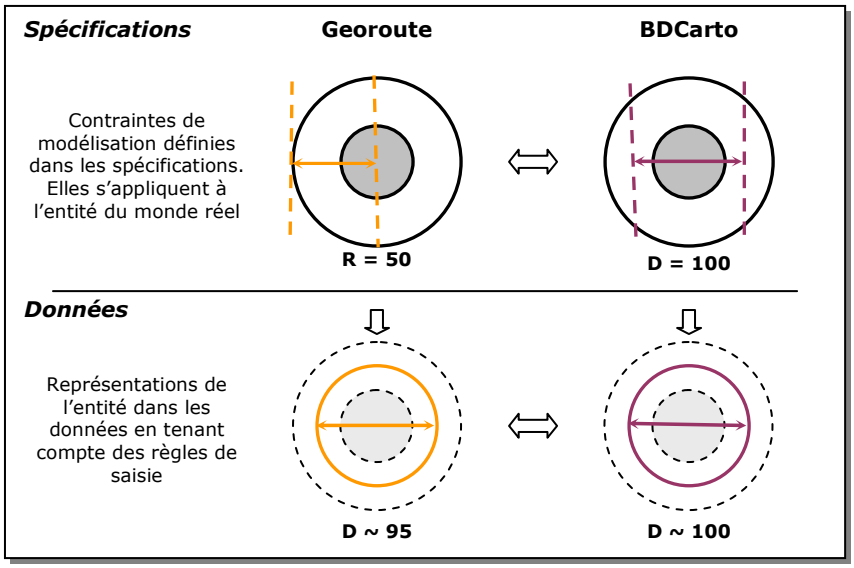


***Figure 104**. Deux représentations comparables dans les données ne s'appuient pas nécessairement sur les mêmes règles de modélisation dans les spécifications.*

Cette analyse des spécifications est une phase d'acquisition de connaissances. C'est la première étape de la méthode *MACO*. Elle nous permet d'identifier les propriétés des objets à contrôler pour juger si les représentations sont conformes ou non. Elle nous aide également à définir les règles pour mener les contrôles intra-base et inter-bases. Elle permet enfin de préciser quels sont les objets à extraire des données et comment il faut les caractériser pour mener les contrôles.

Au terme de cette étape, différentes connaissances ont pu être recueillies : une spécification des outils d'enrichissement et d'appariement (ils seront présentés aux étapes d'enrichissement et d'appariement), et deux bases de règles destinées aux contrôles intra-base et inter-bases. Ces règles ont été obtenues après une phase de reformulation des spécifications. Celles-ci, initialement décrites en langue naturelle, ont été représentées dans un langage de la logique d'ordre 0+, compréhensible par JESS. Les règles définies peuvent être trouvées en annexe 2. Nous les décrirons aux étapes de contrôles intra-base et inter-bases. Nous présentons la phase d'enrichissement ci-dessous.

E.3.4 ENRICHISSEMENT

Suite à l'analyse des spécifications, nous avons pu constater qu'il n'existait pas de classes *Rond-Point* dans les bases. Les ronds-points sont regroupés avec des objets d'autres natures dans des classes moins spécifiques (*Nœud Routier* et *Carrefour Complexe*). En outre, les ronds-points détaillés de la BDCarto n'ont pas une existence explicite dans la base. Ils n'existent qu'à travers la géométrie, ce qui rend impossible leur manipulation. Comme le préconise la méthode *MECO*, la phase d'enrichissement s'impose pour mener l'évaluation de la cohérence. Elle concerne à la fois les schémas et les données.

E.3.4.1 ENRICHISSEMENT DES SCHEMAS

SCHEMA ENRICHI DE LA BDCARTO

Pour la BDCarto, l'enrichissement du schéma se traduit par la création de trois classes : une classe mère abstraite *Rond-Point* et deux classes filles *Rond-Point Simple* et *Rond-Point Complexe* (figure 105). La classe *Rond-Point Simple* fait référence aux ronds-points de modélisation ponctuelle tandis que la classe *Rond-Point Complexe* représente les ronds-points ayant une représentation détaillée.

Les relations entre ces nouvelles classes et les classes initiales sont également définies. Un rond-point simple correspond à un nœud routier dont la valeur de l'attribut « Type » correspond à 'carrefour simple' ou 'petit rond-point'. Un rond-point complexe est composé d'un agrégat de tronçons et de nœud routier, en plus de sa géométrie surfacique. Il a par ailleurs une relation avec un nœud routier de type 'grand rond-point' qui correspond à sa représentation généralisée.

S'ajoutent à ces nouvelles classes des attributs qui ont pour objet de caractériser les données et d'enregistrer les informations nécessaires au contrôle de la conformité des représentations. Pour les ronds-points complexes, cinq attributs nous sont utiles. D'abord le *diamètre*, puisque la représentation détaillée est conditionnée par la longueur de celui-ci. Ensuite l'*indice de circularité* de Miller [Campbell 2000]. Celui-ci va nous permettre d'extraire les objets ronds-points des données. Nous donnerons sa définition plus loin. L'attribut *vocation des tronçons* est défini car nous souhaitons

contrôler que les tronçons constitutifs du rond-point complexe ont bien la vocation de 'bretelle'. Nous enregistrons également l'information concernant la présence ou non d'un nœud 'grand rond-point' dans le cas d'une représentation détaillée (attribut *correspondance GRP*). Normalement, cette double représentation est obligatoire mais un oubli est possible dans les données. Nous devons le vérifier. Enfin, nous enregistrons le nombre de nœuds qui composent le rond-point. Cette information peut être utile pour savoir si le rond-point est dans un cul-de-sac ou pas (bien que les spécifications de la BDCarto ne font pas référence à ces cas).

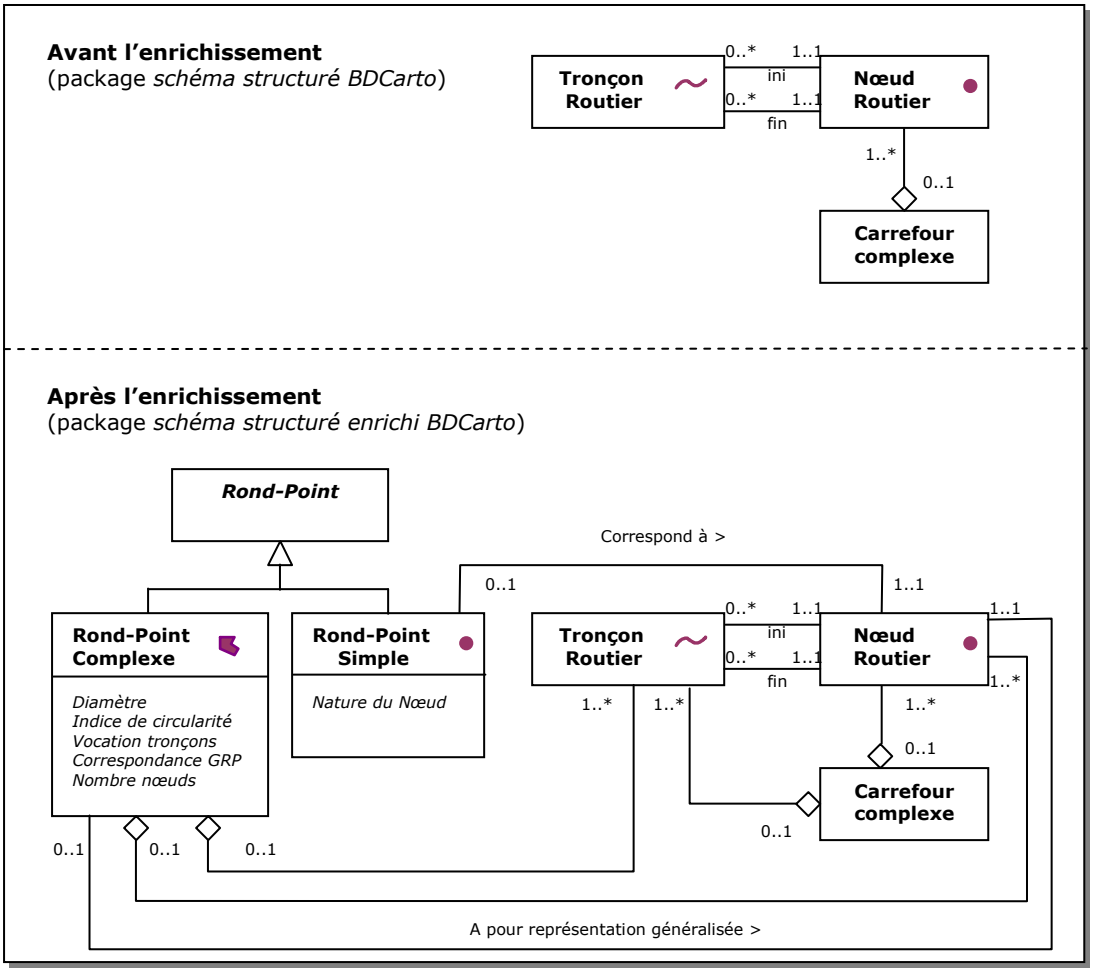


Figure 105. *Enrichissement du schéma de la BDCarto dans le cadre du processus d'évaluation. La notation à base de pictogrammes est inspirée du modèle MADS.*

SCHEMA ENRICHI DE GEOROUTE

L'enrichissement du schéma de Géoroute est assez similaire à celui de la BDCarto. Trois nouvelles classes sont créées : une classe mère abstraite *Rond-Point* et deux classes filles *Rond-Point Simple* et *Rond-Point Complexe* (figure 106). Les relations entre les classes diffèrent cependant. Il n'existe pas une double représentation dans le cas des ronds-points complexes, autrement dit, seule une relation d'agrégation existe entre les nœuds routiers et la classe *Rond-Point Complexe*. Par ailleurs, la classe *Carrefour Complexe* a cette fois une géométrie. Il existe donc une relation entre la classe *Rond-Point Complexe* et *Carrefour Complexe*. Par contre, la classe *Carrefour Complexe* n'est pas reliée aux classes *Nœud Routier* et *Tronçon Routier* dans le schéma structuré initial.

Les attributs créés ne sont pas non plus tous identiques. Nous avons cette fois un attribut *source* qui indique si l'objet rond-point complexe est issu de la classe *Carrefour Complexe* ou s'il a été créé par analyse de la géométrie. Le *sens du cycle* est également repris car nous devons vérifier qu'il est bien direct (sens giratoire). Nous pouvons vérifier ce sens pour Géoroute car les tronçons routiers de cette base contiennent les informations relatives au sens de circulation, ce qui n'est pas le cas de la BDCarto. Un attribut *cul-de-sac* est également défini. Il permet de spécifier que le rond-point simple est un cul-de-sac ou pas. Les attributs *vocation des tronçons* et *correspondance GRP* de la BDCarto ne s'appliquent pas ici.

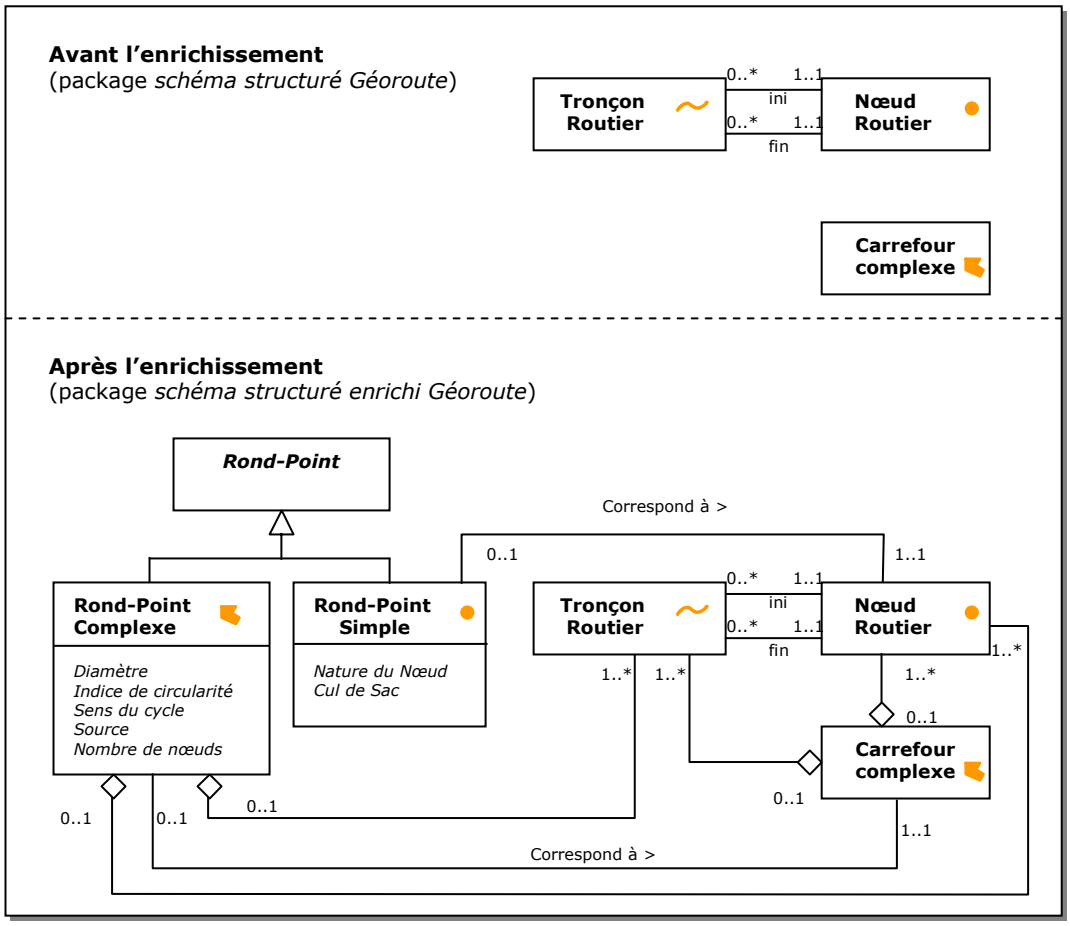


***Figure 106**. Enrichissement du schéma de Géoroute dans le cadre du processus d'évaluation. La notation à base de pictogrammes est inspirée du modèle MADS.*

CORRESPONDANCES ENTRE SCHEMAS

La déclaration des correspondances entre les schémas des bases structurées (non enrichies) est supposée connue avant la mise en œuvre du processus d'évaluation. Pour les ronds-points, ces correspondances (ACI) pourraient être formulées de la manière suivante (d'après le langage défini par [Spaccapietra et al. 1992] étendu par [Devogele 1997], cf. A.3.2.2) :

$\text{SELECTION}_{(\text{BDC.Nœud.Type = 'carrefour simple'})}\text{BDC.Nœud}$
$\supseteq \text{Rond-Point(GEO.SET)} \wedge (25<\text{diametre\_Rond-Point(GEO.SET)}<50)$
GEO.SET([2,N]Nœud,[2,M]Tronçon)

$\wedge \text{SELECTION}_{(\text{BDC.Nœud.Type = 'petit rond-point'})}\text{BDC.Nœud}$
$\equiv \text{Rond-Point(GEO.SET)} \wedge (50<\text{diametre\_Rond-Point(GEO.SET)}<100)$
GEO.SET([2,N]Nœud,[2,M]Tronçon)

$\wedge \text{SELECTION}_{(\text{BDC.Nœud.Type = 'grand rond-point'})}\text{BDC.Nœud}$
$\equiv \text{Rond-Point(GEO.SET)} \wedge (\text{diametre\_Rond-Point(GEO.SET)}>100)$
GEO.SET([2,N]Nœud,[2,M]Tronçon)

$\wedge \text{SELECTION}_{(\text{BDC.Nœud.Type = 'carrefour simple'})}\text{BDC.Nœud}$
$\supseteq_{\text{sinon}}$
$\text{SELECTION}_{(\text{GEO.Nœud.Nature = 'rond-point'})}$ GEO.Nœud

$\text{SELECTION}_{(\text{Rond-Point(BDC.SET)})}\text{BDC.SET}([2,J]\text{Nœud},[2,K]\text{Tronçon}_{\text{where(BDC.Tronçon.Vocation = 'bretelle')}})$
$\equiv \text{Rond-Point(GEO.SET)} \wedge (\text{diametre\_Rond-Point(GEO.SET)}>100)$
GEO.SET([2,N]Nœud,[2,M]Tronçon)

A l'issue de l'enrichissement des schémas, la création des classes *Rond-Point Simple* et *Rond-Point Complexe* a pour effet de simplifier les correspondances. Le contenu des bases est plus homogène. Les conflits de fragmentation exprimés par 'SET' disparaissent :

$\text{SELECTION}_{(\text{BDC.RondPSimple.Nature = 'carrefour simple'})}\text{BDC.RondPSimple}$
$\supseteq$
GEO.RondPSimple OU $\text{SELECTION}_{(\text{GEO.RondPComplexe.Diamètre < 50})}$GEO.RondPComplexe

$\wedge \text{SELECTION}_{(\text{BDC.RondPSimple.Nature = 'petit rond-point'})}\text{BDC.RondPSimple}$
$\equiv$
$\text{SELECTION}_{(50 < \text{GEO.RondPComplexe.Diamètre} < 100)}$GEO.RondPComplexe

$\wedge \text{SELECTION}_{(\text{BDC.Nœud.Type = 'grand rond-point'})}\text{BDC.Nœud}$
$\equiv$
$\text{SELECTION}_{(\text{GEO.RondPComplexe.Diamètre > 100})}$GEO.RondPComplexe

BDC.RondPComplexe
$\equiv$
$\text{SELECTION}_{(\text{GEO.RondPComplexe.Diamètre > 100})}$GEO.RondPComplexe

E.3.4.2 ENRICHISSEMENT DES DONNEES

Maintenant que les deux bases sont préparées à accueillir les ronds-points, nous pouvons envisager l'instanciation des nouvelles classes et des relations définies.

EXTRACTION ET CARACTERISATION DES RONDS-POINTS DANS LA BDCARTO

L'enrichissement des données dans la BDCarto s'est déroulé en deux étapes principales : la création des ronds-points simples et la création des ronds-points complexes.

Les ronds-points simples sont très faciles à créer puisqu'il s'agit de nœuds routiers particuliers ('petit rond-point' et 'carrefour simple'). Il suffit d'effectuer une sélection sur base de la valeur de l'attribut « Type » pour instancier la classe. C'est la

méthode qui a été suivie. Néanmoins, un 'carrefour simple' ne correspond pas toujours à un rond-point. La classe *Rond-Point Simple* renferme donc plus d'objets qu'elle ne devrait en contenir. Les objets sont potentiellement des ronds-points. Cette classe sera filtrée au fur et à mesure du processus.

La création des ronds-points complexes est moins évidente. Cette fois, il s'agit d'extraire les objets des données à l'aide d'outils d'analyse spatiale. L'algorithme que nous avons développé se décompose en trois étapes :

1) Une structure de carte topologique est d'abord reconstruite à partir des tronçons de route et des nœuds routiers.
2) Chaque face est ensuite analysée et celles qui ont la forme d'un rond-point sont retenues (les faces circulaires). Les relations avec les classes *Tronçon Routier* et *Nœud Routier* sont également instanciées.
3) Les ronds-points créés sont finalement combinés aux nœuds routier 'grand rond-point'.

La première étape fait donc appel au schéma de la carte topologique (cf. E.1.1.2.). Tous les tronçons et les nœuds routiers sont d'abord traduits en instances des classes *ARC* et *NŒUD* de cette structure, à l'exception des tronçons logiques (sinon les ronds-points ne pourront pas être créés). La topologie de réseau est ensuite calculée en se basant sur la géométrie des objets (les relations entre les nœuds et les arcs sont instanciées, des nœuds manquants sont créés pour obtenir un graphe planaire, le sens des arcs est défini, etc.). A partir de ce graphe planaire, la topologie *Arc*/*Face* est finalement déduite (définition d'une géométrie surfacique pour chaque face à partir des cycles du graphe, recherche des faces à gauche et à droite de chaque arc). On obtient ainsi une carte topologique.

A la seconde étape, toutes les faces créées sont analysées. Pour ne retenir que les faces circulaires qui ont la forme de ronds-points détaillés, nous avons utilisé l'indice de circularité de Miller [Campbell 2000]. Il s'agit d'un indice qui fait partie d'une famille d'indicateurs décrivant la forme d'un polygone [Agent 1999a], indicateurs souvent construits sur la comparaison avec le périmètre ou la superficie d'une forme de référence (comme un cercle ou un carré). L'indice de Miller a notamment été utilisé par [Weber et al. 2003] pour caractériser des surfaces de visibilité associées à des stations de mesure de pollution. Il a également été utilisé dans le cadre d'un contrôle qualité [Chrisman et Lester 1991]. Il se définit comme le *rapport de la superficie d'une entité à celle du cercle de même périmètre* :

$$I_M = \frac{4\,\pi\,S}{P^2}$$

C'est un indice qui varie entre 0 et 1 (0 = surface dégénérée en ligne, 1 = cercle). Le seuil que nous avons déterminé pour sélectionner les ronds-points est de 0.95. Il a été fixé après une analyse interactive des données et paraît à lui seul suffisant pour extraire les objets voulus. Si plusieurs indicateurs avaient été nécessaires, nous aurions probablement eu recours à l'apprentissage automatique pour déterminer les seuils.

Une fois les faces filtrées et la classe *Rond-Point Complexe* instanciée (y compris les relations avec les autres objets), les valeurs des attributs sont calculées. Le

diamètre correspond à la longueur du plus grand axe de l'objet surfacique (proche d'un cercle) et la *vocation des tronçons* prend la valeur 'bretelle' si tous les arcs constituant le rond-point ont cette valeur dans la base.

La dernière étape dans l'algorithme d'enrichissement de la BDCarto établit les correspondances entre les nouveaux ronds-points créés et les nœuds 'grand rond-point'. Pour ce faire, l'intersection entre la géométrie des ronds-points et la géométrie des nœuds 'grand rond-point' est calculée. Si une intersection existe, la relation est instanciée et l'attribut *correspondance GRP* prend la valeur 'vrai'. Dans le cas contraire, l'attribut prend la valeur 'faux – nœud GRP absent'. L'explication de cette absence sera fournie au cours du processus. Il peut s'agir d'un déficit ou il se peut que la face retenue (même si elle est circulaire) ne représente pas un rond-point. Inversement, un nœud 'grand rond-point' dans la base peut ne pas avoir de correspondant dans la classe *Rond-Point Complexe*, soit parce que le seuil fixé pour filtrer les faces a laissé passer des ronds-points, soit parce que le nœud a des attributs erronés. Dans ce cas, nous importons les faces non retenues dans la classe *Rond-Point Complexe* si le nœud routier 'grand rond-point' est inclus dans une face, que cette face ne fait pas plus de 250 m de diamètre, et que les tronçons constituant cette face sont classés en 'bretelles'. Nous évitons de cette manière d'introduire des erreurs dans la classe *Rond-Point Complexe* (si l'absence est liée à une erreur sémantique sur le nœud). L'attribut *correspondance GRP* prend la valeur 'faux – RPX importé' en cas d'importation. Quelques exemples de ronds-points créés sont donnés en figure 107.

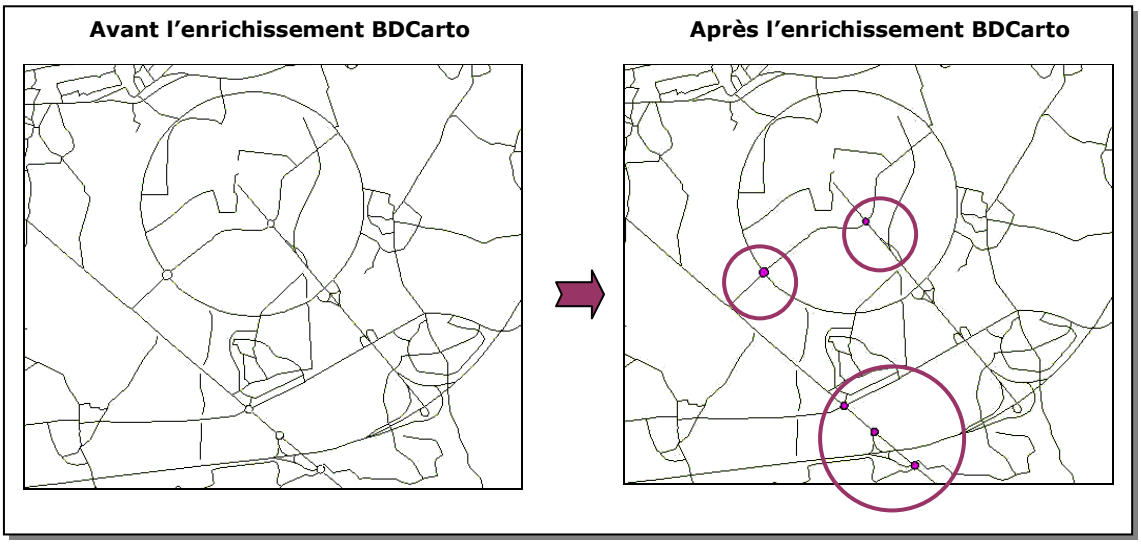


***Figure 107**. Exemples de ronds-points matérialisés automatiquement dans la BDCarto à l'issue de l'enrichissement.*

Au total, 18 ronds-points complexes ont été extraits du jeu de données étudié et 7798 ronds-points simples potentiels (la majorité correspond à des intersections simples et non des ronds-points).

EXTRACTION ET CARACTERISATION DES RONDS-POINTS DANS GEOROUTE

La méthode d'enrichissement des données dans Géoroute est assez proche de celle suivie pour la BDCarto. Bien qu'il existe cette fois des carrefours complexes portant une géométrie à partir desquels nous pourrions instancier la classe *Rond-Point Complexe*, nous avons quand même créé la plupart de nos objets ronds-points. Ceci s'explique pour deux raisons. D'abord, si nous voulons vérifier qu'il n'existe pas de déficit dans la classe *Carrefour Complexe* (ronds-points visuellement identifiables mais

manquants dans la classe), nous devons nécessairement reconstruire les objets. Ensuite, nous avons remarqué que les objets *Carrefour Complexe* dans Géoroute n'étaient pas toujours topologiquement cohérents avec les tronçons de route sur lesquels ils s'appuient. C'est ce que nous illustrons en figure 108. De ce fait, il est préférable de reconstruire ces objets et de les comparer aux éléments stockés dans la base plutôt que de les importer directement depuis la classe *Carrefour Complexe*.

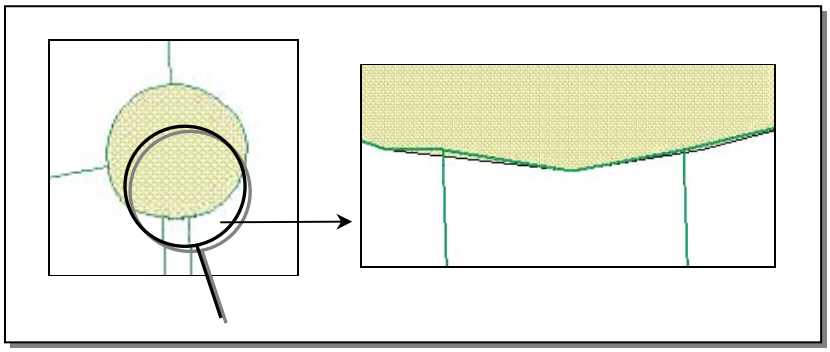


Figure 108. *Incohérence topologique détectée pour un carrefour complexe de Géoroute.*

Le principe de l'algorithme exposé pour l'enrichissement de la BDCarto est assez similaire pour Géoroute. Néanmoins, la comparaison entre les ronds-points complexes créés ne se fait plus avec les nœuds routiers 'grand rond-point' mais avec les objets de la classe *Carrefour Complexe* d'attribut 'rond-point'.

Il est possible qu'un rond-point dans la classe *Carrefour Complexe* n'ait pas de correspondant dans le jeu de ronds-points créé. Ceci s'explique par le fait que des ronds-points de forme non circulaire existent dans Géoroute (cf. spécifications). Dans ce cas, nous avons recherché la face correspondante au rond-point non circulaire dans la carte topologique et importé cette face dans la classe *Rond-Point Complexe*. De cette manière, ces objets sont pris en compte dans notre jeu de ronds-points et leur cohérence topologique est assurée (figure 109). Cette méthode est proche de la comparaison menée entre les ronds-points détaillés et les nœuds routiers 'grand rond-point' pour la BDCarto.

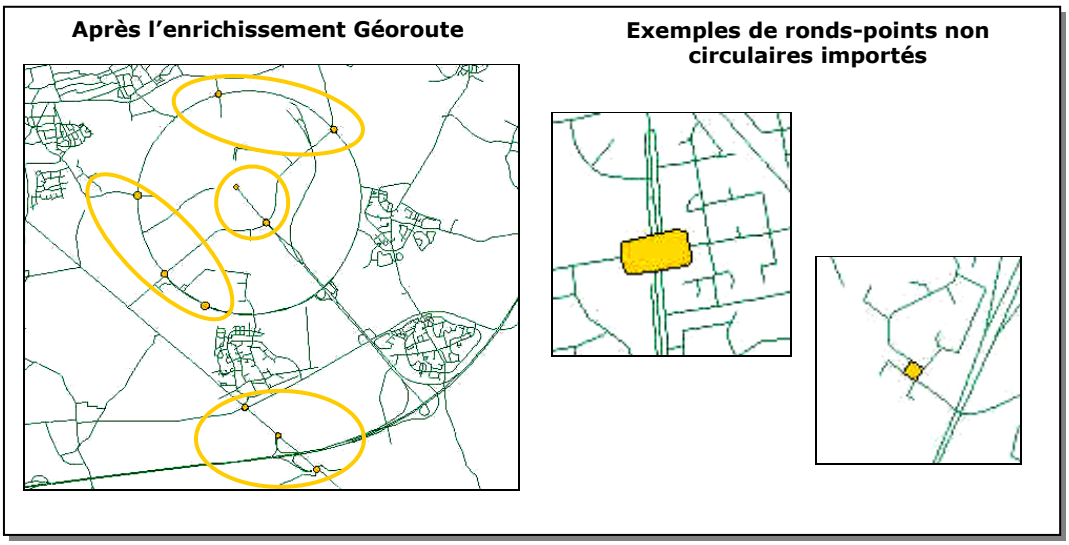


Figure 109*. Création et import des ronds-points complexes dans Géoroute.*

Finalement, une fois la nouvelle classe *Rond-Point Complexe* instanciée, les attributs s'y rapportant sont calculés. Le *sens du cycle* est déterminé à partir du sens

de saisie des arcs et de l'attribut « sens de circulation » des tronçons routiers. Il prend la valeur 'sens giratoire' ou 'sens non giratoire'.

La création des ronds-points simples ne pose aucune difficulté. Il s'agit de sélectionner tous les nœuds routiers qui portent la valeur 'rond-point simple' pour l'attribut « nature de l'intersection ». Cette fois, la classe *Rond-Point Simple* ne contient que des éléments de cette nature, contrairement à la BDCarto. L'instanciation de l'attribut *cul-de-sac* nécessite de faire une analyse du graphe routier calculé. Si le nœud n'est relié qu'à un seul arc, il s'agit d'un cul-de-sac.

Au total, ce sont 170 ronds-points complexes qui ont été créés contre 506 ronds-points simples.

CONCLUSION SUR L'ENRICHISSEMENT

Les méthodes d'enrichissement proposées ne sont pas complètement exemptes d'erreurs puisque la construction des ronds-points détaillés s'appuie uniquement sur les données et que le monde réel n'est pas accessible (un rond-point pourrait être oublié ou créé alors qu'il ne devrait pas l'être). Néanmoins, si erreurs il y a, leur nombre est faible (2 erreurs ont été trouvées). Grâce à l'existence des nœuds routiers 'grand rond-point' et des carrefours complexes 'rond-point', nous pouvons avoir un certain contrôle sur les jeux de ronds-points construits. Par la suite, en comparant les ronds-points des deux bases et en étudiant leur cohérence inter-représentations, certaines erreurs introduites pourront être détectées.

E.3.5 CONTROLE INTRA-BASE

Maintenant que les données ont été préparées aux contrôles de cohérence, nous pouvons envisager la mise en œuvre du contrôle intra-base (2ème étape de *MECO*). Rappelons que l'objectif du contrôle intra-base est de vérifier la conformité des représentations de chacune des bases indépendamment, avant leur mise en correspondance. Ce contrôle est effectué sur les données des bases et est vue comme un contrôle d'intégrité.

E.3.5.1 CONTROLE DES RONDS-POINTS DE LA BDCARTO

Pour la BDCarto, on peut déduire des spécifications plusieurs règles à contrôler. Le contenu des spécifications est suffisamment exhaustif pour mettre en œuvre ce contrôle. Nous exploitons ainsi les seuils fixés dans les documents. Nous apprendrons par la suite d'autres règles pour tenir compte des connaissances implicites utilisées lors de la saisie (cf. E.2.7.2.)

Plusieurs éléments doivent être vérifiés pour les ronds-points complexes :

- Le diamètre des objets : il doit être supérieur à 100m ;
- La vocation des tronçons constitutifs du rond-point : l'attribut doit prendre la valeur 'bretelle' ;
- L'existence d'une double représentation : il doit exister une représentation détaillée et une représentation ponctuelle (nœud 'grand rond-point').

Ces connaissances ont été traduites sous forme de règles de production et introduites dans le système-expert.

Au sujet de la double représentation, nous n'avons pas été capable de juger à ce niveau la conformité de l'existence des objets. En fonction de l'origine du rond-point complexe (créé ou importé), plusieurs interprétations pouvaient être envisagées en cas d'absence de nœud 'grand rond-point'. On peut considérer que l'absence est normale et que le rond-point complexe n'aurait pas du être créé car il ne correspond pas à un rond-point. On peut également supposer qu'il existe une erreur de complétude dans la BD et que le nœud est un oubli anormal. Nous avons attendu le contrôle inter-bases pour lever le doute. Nous avons considéré que si cette absence apparaissait également dans l'autre base (absence du carrefour complexe), il s'agissait d'une erreur de construction du rond-point. Il est peu probable qu'une erreur de complétude apparaisse en effet au même endroit dans l'autre base (même si cela est possible en théorie).

Le tableau ci-dessous résume les résultats du contrôle intra-base mené sur les ronds-points complexes de la BDCarto. Précisons que ces informations sont stockées dans une classe créée à cette fin (figure 110).

***Tableau 2**. Résultats du contrôle intra-base pour les ronds-points complexes de la BDCarto.*

Propriétés contrôlées	Résultats du contrôle
Diamètre	16/18 conformes 2/18 non conforme (valeurs : 82m et 94 m)
Vocation des tronçons	17/18 conformes 1/18 non conforme
Existence de la double représentation	15/18 conformes (correspondance existante) 3/18 classés en 'erreur de déficit du nœud GRP ou erreur de construction du rond-point complexe'

A ce niveau, aucun contrôle ne peut être effectué sur les ronds-points simples. Seule la cohérence inter-représentations est étudiée pour ces objets.

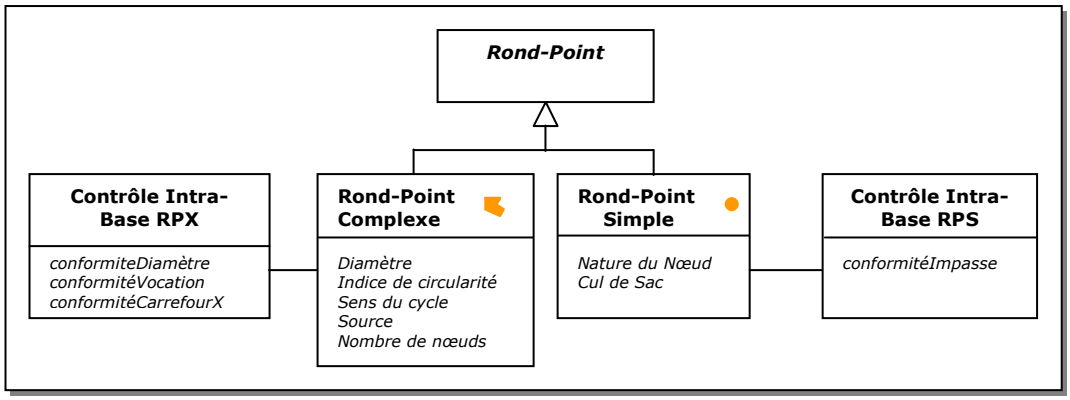


***Figure 110**. Tous les ronds-points créés dans les données sont associés à des classes relatives aux contrôles intra-base et inter-bases.*

E.3.5.2 CONTROLE DES RONDS-POINTS DE GEOROUTE

Le contrôle des ronds-points complexes dans Géoroute doit tenir compte des spécifications suivantes :

- Le diamètre des objets : il doit être supérieur à 25m ;
- Le sens du cycle : il doit être direct (sens giratoire) ;
- L'existence d'un carrefour complexe : il doit exister un carrefour complexe pour chaque rond-point complexe créé.

Ces connaissances sont traduites sous forme de règles dans le système-expert.

Concernant le diamètre, nous avons déjà expliqué la raison pour laquelle nous fixons le seuil à 25m au lieu de 30m. Nous devons tenir compte du fait que la saisie de l'objet dans les données se fait d'axe à axe alors que le seuil fait référence à l'emprise totale. Il ne s'agit pas ici d'une modification de la spécification. Il s'agit plutôt de sa traduction dans l'univers des données.

Le contrôle de la conformité de la correspondance entre le carrefour complexe et le rond-point complexe n'a pas pu se faire à ce niveau, pour les mêmes raisons que celles évoquées pour la BDCarto. La vérification a été réalisée lors du contrôle inter-bases. Nous donnons les résultats obtenus à cette étape dans le tableau 3.

Tableau 3. *Résultats du contrôle intra-base pour les ronds-points complexes de Géoroute.*

Propriétés contrôlées	Résultats du contrôle
Diamètre	165/170 conformes 5/170 non conformes (10m < ∅ < 23m)
Sens du cycle	165/170 conformes 5/170 non conformes
Existence du carrefour complexe	161/170 conformes (correspondance existante) 9/170 classés en 'erreur de déficit du carrefour complexe ou erreur de construction du rond-point complexe'

Au sujet des ronds-points simples, la règle de saisie indiquant qu'un rond-point dans un cul-de-sac doit être codé en 'intersection simple' peut être vérifiée. Sur 506 ronds-points simples, nous avons trouvé 174 erreurs par rapport à cette règle dans les données (environ 34%). Ce nombre est particulièrement élevé et laisse penser que cette règle est peu suivie en pratique.

E.3.6 APPARIEMENT

Maintenant que les données ont été contrôlées de manière indépendante, les correspondances vont être calculées. C'est l'objet de l'étape d'appariement (3ème étape de *MECO*).

Nous proposons une méthode d'appariement qui calcule les liens entre les objets dans les deux sens (BDCarto → Géoroute et Géoroute → BDCarto). Elle se compose des étapes suivantes :

1. Appariement des ronds-points complexes BDCarto et Georoute ;
2. Appariement des ronds-points complexes BDCarto et des ronds-points simples Georoute ;

3. Appariement des ronds-points complexes Georoute et des ronds-points simples BDCarto ;
4. Appariement des ronds-points simples Georoute et des ronds-points simples BDCarto ;
5. Recherche des appariements 1-0 ('Petit Rond-point' BDCarto - Georoute) ;

L'algorithme d'appariement est détaillé en annexe 3. Le principe est simple : l'appariement est fondé sur un critère d'intersection et de proximité. Les couples définis sont ensuite caractérisés à l'aide de plusieurs attributs (cardinalité du lien, nature de l'objet BDCarto, nature de l'objet Géoroute, etc.), en plus des attributs « critère d'appariement » et « confiance dans l'appariement » instanciés au fur et à mesure du calcul des liens.

ÉVALUATION DE L'APPARIEMENT

Notre méthode d'appariement ne garantit pas que tous les couples calculés sont justes, c'est-à-dire que les objets sont bien appariés. Le critère de distance par exemple peut conduire à sélectionner plusieurs candidats, or un seul candidat correspond à l'objet homologue, lequel n'est d'ailleurs pas nécessairement le candidat le plus proche. Inversement, la distance fixée peut être trop courte pour sélectionner l'objet homologue et le lien calculé laissera supposer qu'une erreur de complétude ou d'excédent réside dans une des bases.

Pour évaluer le taux d'erreur introduit dans l'ensemble des couples d'appariement calculés, nous avons sélectionné au hasard un échantillon de 124 correspondances et déterminé si celles-ci étaient justes. Nous avons pu détecter interactivement 8% d'erreurs d'appariement.

Dans l'optique d'automatiser cette évaluation, nous avons alors décidé de mettre en œuvre une seconde méthode d'appariement automatique, celle proposée par [Devogele 1997] dans le cadre de sa thèse. Nous avons recalculé tous les couples de ronds-points et comparé les résultats des deux appariements. Les couples présentant la même réponse ont été jugés certains, les autres ont été qualifiés d'incertains.

La méthode proposée par [Devogele 1997] et redéveloppée dans la plate-forme OXYGENE par Sébastien Mustière est fondée sur d'autres critères que ceux retenus dans notre algorithme d'appariement. C'est une méthode plus complexe qui s'applique à l'ensemble du graphe routier et qui exploite en particulier les relations topologiques entre ses éléments. Cette méthode est exposée en annexe 3. La figure 111 illustre le processus dans son ensemble.

L'appariement mené en utilisant la méthode de [Devogele 1997] donne des résultats un peu moins bons que ceux obtenus avec la première méthode. Ceci s'explique par le fait que le processus de [Devogele 1997] est capable de traiter davantage de carrefours complexes (il permet par exemple d'apparier des pattes d'oie) mais n'est pas adapté spécifiquement aux ronds-points. Les erreurs d'appariement commises ne sont pas toujours les mêmes que celles obtenues avec notre méthode. Nous illustrons quelques exemples de correspondances erronées en figure 112.

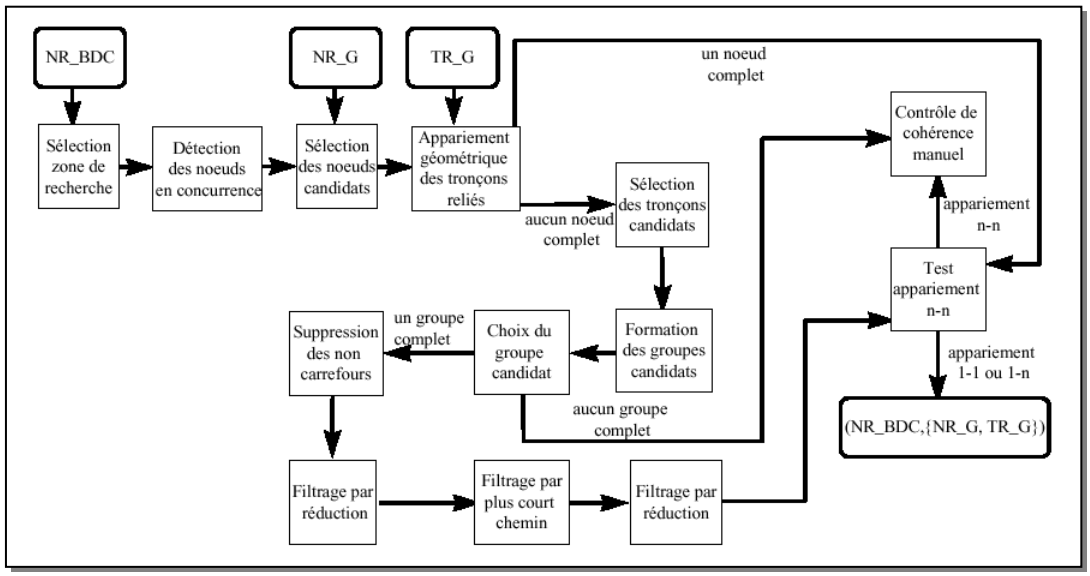


***Figure 111**. Processus d'appariement des graphes routiers de la BDCarto et Géoroute (Source : [Devogele 1997]).*

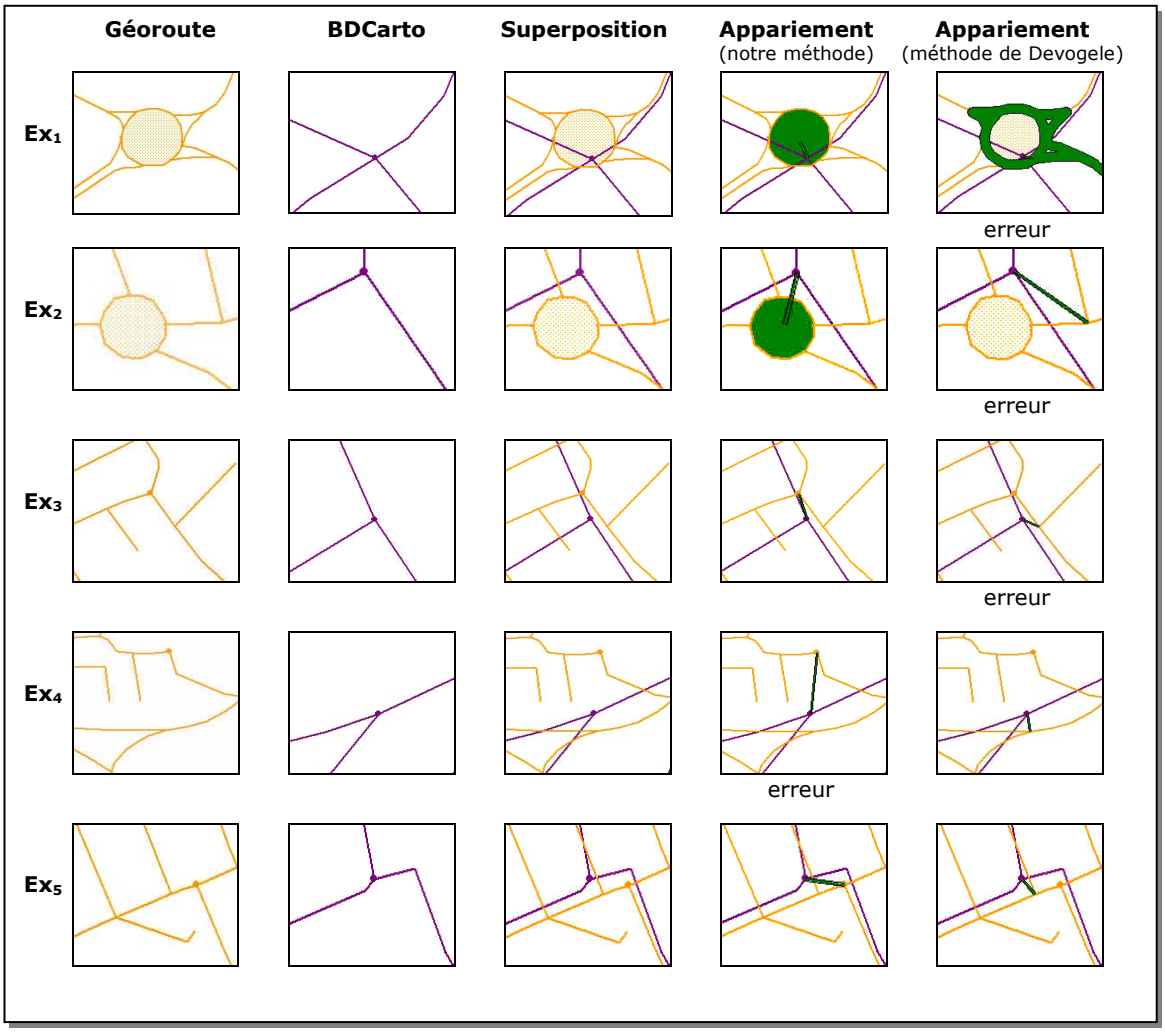


***Figure 112**. Résultats d'appariement selon les deux méthodes utilisées.*

Les deux premiers exemples montrent que le calcul des liens 1-n n'est pas toujours efficace en suivant la méthode de [Devogele 1997]. La création d'un objet

rond-point surfacique rend le calcul des correspondances plus simple et souvent plus juste, surtout lorsque l'objet non détaillé intersecte le polygone.

Le troisième exemple illustre également une erreur dans le couple calculé par la deuxième méthode mais ce type d'erreur est relativement rare. Elle semble même étonnante car l'appariement est réalisé avec un nœud *incomplet* alors que le nœud qui aurait dû être choisi est *complet*. On dit que le nœud est *complet* dans le processus de [Devogele 1997] si chaque tronçon communicant du nœud de la BDCarto s'apparie au moins avec un tronçon communicant de Géoroute. Dans ce cas-ci, il est probable que tous les nœuds aient été jugés incomplets lors du processus et, de ce fait, c'est le nœud le plus proche qui a été choisi. Ceci peut s'expliquer par l'écart angulaire trop important entre deux tronçons homologues. Si l'écart angulaire dépasse un certain seuil (paramètre de l'appariement), les tronçons ne sont pas pré-appariés et ceci a des conséquences sur la qualification des nœuds lors de la validation de l'appariement.

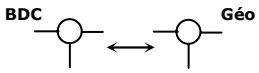
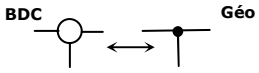
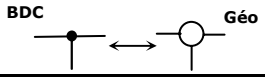
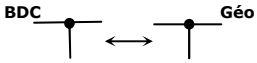
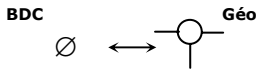
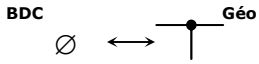
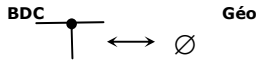
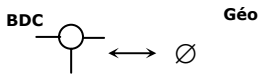
L'exemple suivant illustre une erreur d'appariement dans la première méthode. Cet erreur intervient en raison d'une erreur sémantique d'un nœud dans Géoroute. En principe, si le nœud de Géoroute non apparié (celui à la pointe du triangle qui aurait dû être choisi) avait été bien codé en rond-point simple, le nœud aurait été sélectionné pour être candidat à l'appariement. Dans ce cas, le nœud est codé en intersection simple et, de ce fait, il n'a pas été pris en compte dans le calcul des correspondances. Seuls les objets ronds-points sont sélectionnés pour l'appariement dans la première méthode. Dans le processus de [Devogele 1997], tous les nœuds routiers sont par contre exploités. De ce fait, les objets ont été bien appariés dans ce cas (le nœud étant complet). Ceci montre que la prise en compte de l'ensemble des nœuds routiers rend possible l'appariement de ronds-points dont la sémantique est erronée (incohérence). Mais l'exemple 3 montre également que la sémantique (quand elle est exacte) permet aussi d'éviter des erreurs d'appariement. Cette sémantique n'est pas exploitée dans la méthode de [Devogele 1997]. En fait, nous pensons qu'en couplant les deux méthodes (donc en exploitant tous les nœuds routier, la représentation surfacique des ronds-points, la sémantique des nœuds et les relations topologiques entre les éléments du graphe), le processus d'appariement serait sensiblement plus performant.

Pour conclure, il existe des cas où il est difficile de considérer qu'une des deux méthodes d'appariement a tort bien que les résultats soient différents. C'est ce qui apparaît pour le dernier exemple. D'un point de vue topologique, l'appariement de [Devogele 1997] est juste et donc la correspondance peut être jugée exacte. Mais le lien défini dans la première méthode pourrait également être validé en considérant que le nœud de la BDCarto représente les deux nœuds et le tronçon du carrefour dans Géoroute.

Rappelons que la méthode de [Devogele 1997] a été exploitée dans le but d'évaluer les correspondances calculées avec notre méthode et de détecter les éventuelles erreurs d'appariement. Dans ce sens, tous les couples d'appariement définis dans les deux méthodes ont été comparés automatiquement et les différences ont été notifiées. Un attribut « comparaison des méthodes d'appariement » a été défini à cet effet dans la classe *CoupleAppariement*.

Au total, 690 couples ont été calculés. Parmi ceux-ci, 613 couples sont identiques dans les deux appariements (89%) contre 77 couples différents. Nous avons fait le choix de ne garder que les couples identiques pour la suite du processus bien que dans la liste des couples incertains, beaucoup d'entre eux soient justes (ceux définis par notre méthode). La répartition des couples retenus est donnée dans le tableau ci-dessous. La suite du processus est consacrée au contrôle inter-bases.

***Tableau 4**. Couples d'objets appariés retenus*

Nature du couple	Nombre de couples
Lien 1-1	255 couples
	16
	1
	80
	158
Lien 0-1	354 couples
	40
	314
Lien 1-0	4 couples
	3
	1

E.3.7 CONTROLE INTER-BASES

Le contrôle inter-bases a pour objectif d'étudier la cohérence inter-représentations. C'est à ce niveau que nous avons classé l'ensemble des correspondances calculées en incohérence et en équivalence en analysant conjointement les représentations des deux bases. Il s'agit de la 4ème étape de la méthode *MECO*.

E.3.7.1 DEVELOPPEMENT D'UNE BASE DE REGLES ISSUES DES SPECIFICATIONS

Pour mener ce contrôle, des connaissances sur les bases ont été utilisées comme ce fut le cas aux étapes précédentes. Les premières règles que nous avons développées ont été déduites des spécifications. Les spécifications relatives aux ronds-points sont suffisamment claires et exhaustives dans les documents. De ce fait, nous avons pu écrire les règles manuellement. Celles-ci ont été introduites par la suite dans le système-expert (cf. E.3.3. - étape d'analyse des spécifications).

REGLES DE PREDICTION, COMPARAISON ET CLASSIFICATION

Pour cette expérimentation, nous avons décidé d'organiser les connaissances en suivant l'approche par prédiction (cf. C.5.4.2.) Les deux formes d'expression des connaissances auraient pu s'appliquer mais la classification directe sera illustrée dans un autre contexte (E.4.). Les règles déduites des spécifications et utilisées pour prédire les conditions que doivent respecter les représentations de chacune des bases sont les suivantes :

Prédiction des conditions portant sur les représentations de la BDCarto à partir des représentations de Géoroute :

R_1 **Si** $Objet_{Géo}$ = 'rond-point simple'
ALORS $Objet_{BDC}$ doit être un 'rond-point simple' de type 'carrefour simple'

R_2 **Si** $Objet_{Géo}$ = 'rond-point complexe' et $diamètre\_Objet_{Géo}$ < 50 m
ALORS $Objet_{BDC}$ doit être un 'rond-point simple' de type 'carrefour simple'

R_3 **Si** $Objet_{Géo}$ = 'rond-point complexe' et 50m < $diamètre\_Objet_{Géo}$ < 100 m
ALORS $Objet_{BDC}$ doit être un 'rond-point simple' de type 'petit rond-point'

R_4 **Si** $Objet_{Géo}$ = 'rond-point complexe' et $diamètre\_Objet_{Géo}$ > 100 m
ALORS $Objet_{BDC}$ doit être un 'rond-point complexe' avec $diamètre\_Objet_{BDC}$ > 100 m

Prédiction des conditions portant sur les représentations de Géoroute à partir des représentations de la BDCarto :

R'_1 **Si** $Objet_{BDC}$ = 'rond-point simple' de type 'carrefour simple'
ALORS $Objet_{Géo}$ doit être un 'rond-point simple' ou un 'rond-point complexe' avec 25 m < $diamètre\_Objet_{Géo}$ < 50 m

R'_2 **Si** $Objet_{BDC}$ = 'rond-point simple' de type 'petit rond-point'
ALORS $Objet_{Géo}$ doit être un 'rond-point complexe' avec 50m < $diamètre\_Objet_{Géo}$ < 100 m

R'_3 **Si** $Objet_{BDC}$ = 'rond-point complexe'
ALORS $Objet_{Géo}$ doit être un 'rond-point complexe' avec $diamètre\_Objet_{Géo}$ > 100 m

En comparant ensuite les résultats de l'application de ces règles aux représentations stockées dans chaque base, nous avons pu déterminer si les représentations des ronds-points étaient *équivalentes* ou *incohérentes*.

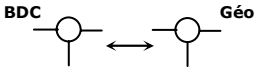
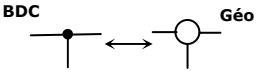
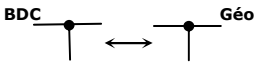
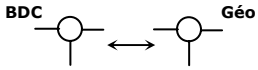
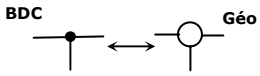
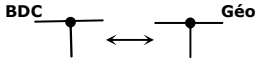
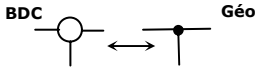
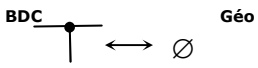
Tous les couples retenus à ce stade du processus ont été soumis à cette évaluation. Précisons que les appariements de type 0-1 (avec 1 correspondant à un objet dans Géoroute) ont dû être écartés. Ce type de lien ne peut pas être analysé à ce niveau car les absences d'intersection dans la BDCarto découlent d'absences de route. Il est donc nécessaire d'étudier la conformité de l'existence des routes pour valider ces liens. Nous présentons dans le tableau 5 les résultats obtenus à la suite de ce contrôle. L'évaluation de la cohérence inter-représentations a porté au total sur 258 couples.

Nous avons trouvé 33% d'incohérences dans les données. Il est possible d'obtenir une incohérence si les deux représentations sont correctes mais que le sens du rond-

point détaillé dans Georoute n'est pas conforme (sens non giratoire). Nous tenons compte ainsi des résultats du contrôle intra-base. De même, si la correspondance avec l'objet « carrefour complexe » est non conforme (déficit ou absence de terre-plein central), cela donne lieu aussi à une incohérence. On peut noter enfin que deux modélisations identiques peuvent être jugées incohérente.

Le nombre d'incohérences détecté est relativement élevé mais on peut considérer, suivant le contexte d'utilisation des BD, que certaines erreurs sont plus graves que d'autres. Pour une application de navigation routière par exemple, une erreur de sens de circulation aura plus d'incidence qu'une erreur de modélisation.

***Tableau 5**. Résultats du contrôle inter-bases (approche par prédiction en exploitant les spécifications)*

Description des couples : - 613 couples jugés certains (89%) - 2 couples composés de « faux » rond-point - 353 couples non traités à ce niveau (lien 0-1) - 258 couples évalués	
Équivalences : 173 couples (67%)	
Type d'équivalence	**Nombre de couples**
	2
	53
	118
Incohérences : 85 couples (33%)	
Type d'incohérence	**Nombre de couples**
	14
	27
	40
	1
	3

Lorsqu'il existe une incohérence entre les représentations et que leur modélisation est ponctuelle, l'erreur peut provenir soit de la BDCarto (le nœud est classé en « petit rond-point » au lieu d'être classé en « carrefour simple), soit de Georoute (la

représentation est ponctuelle bien qu'elle devrait être détaillée). Pour ce cas, on constate qu'il existe une incohérence mais il n'est pas possible de préciser dans quelle BD existe l'erreur. Par contre, dans le cas d'une relation « point - surface » (le point appartenant à la BDCarto), on peut interpréter l'incohérence plus finement. Ainsi, si le nœud est classé en « carrefour simple » et que le diamètre de Georoute mesure 75 mètres, on peut faire l'hypothèse que c'est le nœud qui est mal classé. Il est peu probable que le diamètre soit surévalué par rapport à la réalité, la saisie étant issue de clichés aériens. Les deux résultats possibles ont néanmoins été enregistrés dans la base. Quelques illustrations d'incohérences et d'équivalences sont fournies en figure 113.

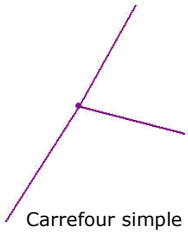
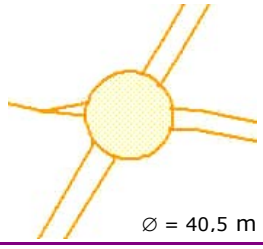
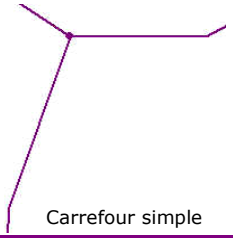
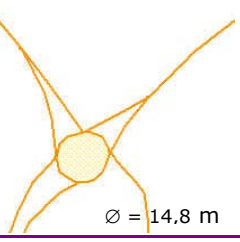
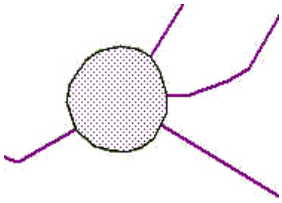
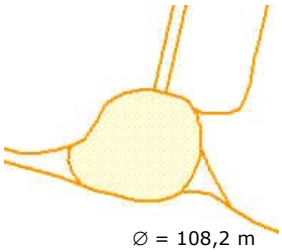
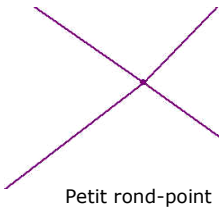
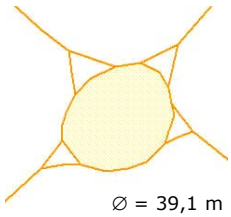
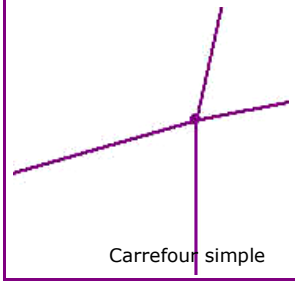
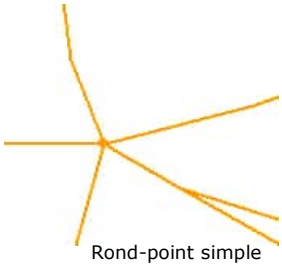
BDCarto	Géoroute	Évaluation
 Carrefour simple	 ∅ = 40,5 m	***Équivalence***
 Carrefour simple	 ∅ = 14,8 m	***Incohérence*** ∅ rond-point complexe Georoute non conforme
 ∅ = 81,8 m	 ∅ = 108,2 m	***Incohérence*** ∅ rond-point complexe BDCarto non conforme
 Petit rond-point	 ∅ = 39,1 m	***Incohérence*** Nature du rond-point simple BDCarto non conforme
 Carrefour simple	 Rond-point simple	***Équivalence***

Figure 113. *Illustrations d'équivalences et d'incohérences*

E.3.7.2 DEVELOPPEMENT D'UNE BASE DE REGLES PAR APPRENTISSAGE AUTOMATIQUE

Après avoir effectué cette première évaluation, nous avons souhaité mener une interprétation en tenant compte de la réalité de la saisie. Par conséquent, nous avons décidé de mettre en œuvre l'apprentissage automatique pour induire les règles de prédiction à partir des données plutôt que de les déduire des spécifications (étape optionnelle de la méthode *MACO*). Comme nous l'avons déjà exposé dans le chapitre D, la mise en œuvre de l'apprentissage pour la prédiction est pratiquement immédiate car les exemples sont créés automatiquement à la suite de l'appariement.

PREDICTION DES CONDITIONS RELATIVES A LA BDCARTO

Pour découvrir les conditions que doivent respecter les représentations de la BDCarto, les exemples d'apprentissage ont été décrits par deux descripteurs : (1) le diamètre du rond-point complexe Géoroute et (2) le type de rond-point, simple ou complexe. Les exemples peuvent être trouvés en annexe 4. Un extrait est présenté dans le tableau ci-dessous (tableau 6). L'étiquette des exemples correspond à la représentation associée de la BDCarto, c'est-à-dire un rond-point simple de type 'carrefour simple', un rond-point simple de type 'petit rond-point' ou un rond-point complexe. L'algorithme d'apprentissage utilisé (C4.5. [Quinlan 1993]) fut appliqué sur l'ensemble des couples appariés jugés certains, soit 258 couples. L'arbre de décision obtenu est représenté en figure 114. Le taux d'erreur réelle estimé par validation croisée (k = 10) est de 29,5%. Nous reviendrons sur ce chiffre par la suite (E.3.7.3.).

***Tableau 6**. Extrait des exemples d'apprentissage (Nombre total d'exemples : 258)*

	Attributs		**Classe**
	Type de rond-point dans Géoroute	**Diamètre du rond-point complexe dans Géoroute**	**Représentation associée du rond-point dans BDCarto**
1	Point	0.0	*Carrefour simple*
2	Point	0.0	*Carrefour simple*
3	surface	42.0	*Petit rond-point*
4	surface	81.28	*Petit rond-point*
5	surface	87.1	*Grand rond-point*
6	surface	40.45	*Carrefour simple*
7	Point	0.0	*Carrefour simple*
8	surface	70.58	*Petit rond-point*
9	point	0.0	*Petit rond-point*
10	surface	91.55	*Petit rond-point*

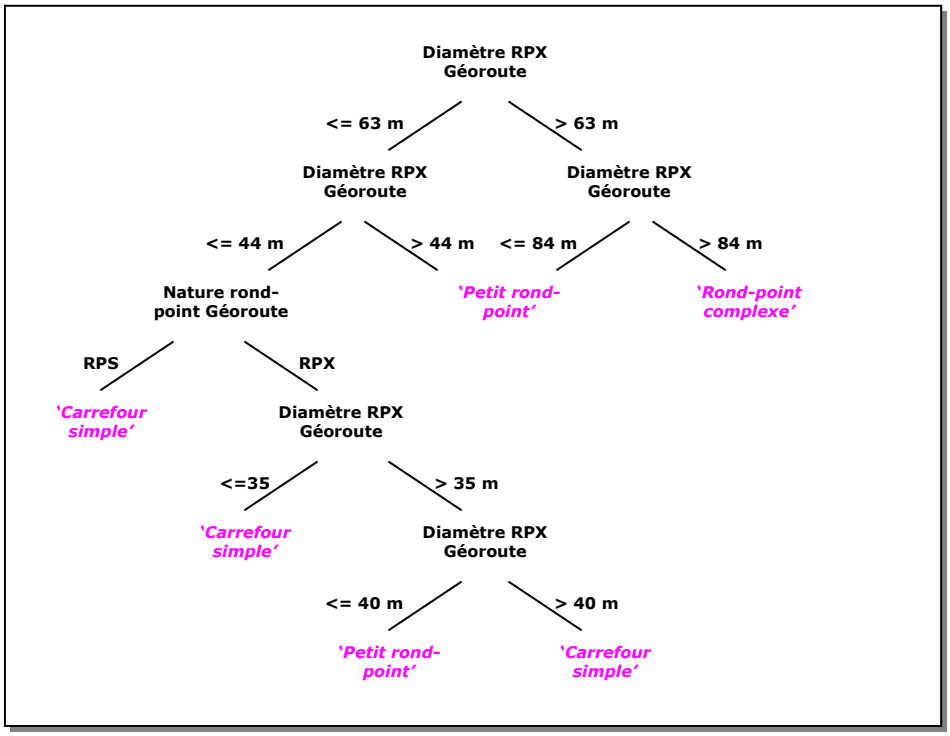


***Figure 114**. Arbre de décision appris déterminant les conditions que doivent respecter les représentations de la BDCarto à partir des représentations de Géoroute.*

ÉVALUATION ET REVISION DES CONNAISSANCES APPRISES POUR LA BDCARTO

Que peut-on conclure sur ces résultats ? Les règles apprises sont-elles cohérentes avec les spécifications des bases ?

Analysons tout d'abord l'arbre de décision obtenu sans tenir compte des couples incertains. On peut constater qu'un rond-point complexe dans la BDCarto est attendu si le diamètre de l'objet homologue dans Géoroute est supérieur à 84m. Cela signifie qu'il existe une différence entre le seuil mentionné dans les spécifications, 100 m, et le seuil appris. D'après les données, il semblerait que les opérateurs de saisie détaillent les ronds-points plus fréquemment qu'ils ne le devraient.

Concernant la classification des ronds-points simples, les branches de l'arbre sont plus complexes. On remarque que pour des valeurs de diamètre comprises entre 44m et 84m, le rond-point simple de la BDCarto devrait correspondre à un 'petit rond-point'. Les spécifications préconisent quant à elles de saisir cette représentation entre 50m et 100m. On constate par ailleurs que la représentation de type 'petit rond-point' devrait s'appliquer entre 35m et 40m d'après l'arbre. En deçà de ces valeurs, le rond-point simple doit être codé en 'carrefour simple', de même qu'au-delà, jusqu'à 44m.

D'après les spécifications des bases, cette dernière règle n'est pas pertinente. Nous pouvons accepter que le seuil soit fixé à 44m pour distinguer les 'carrefours simples' des 'petits ronds-points' mais il n'est pas juste de considérer que les ronds-points simples doivent être représentés par des 'petits ronds-points' lorsque le diamètre de l'objet Géoroute est compris entre 35m et 40m. Il est probable que pour cette règle, l'algorithme d'apprentissage a fait du *sur-apprentissage* (cf. D.4.1.2.).

Pour cette raison, nous avons décidé de réviser l'hypothèse apprise en élaguant les branches qui semblaient trop spécifiques aux données. L'arbre de décision retenu est représenté en figure 115.

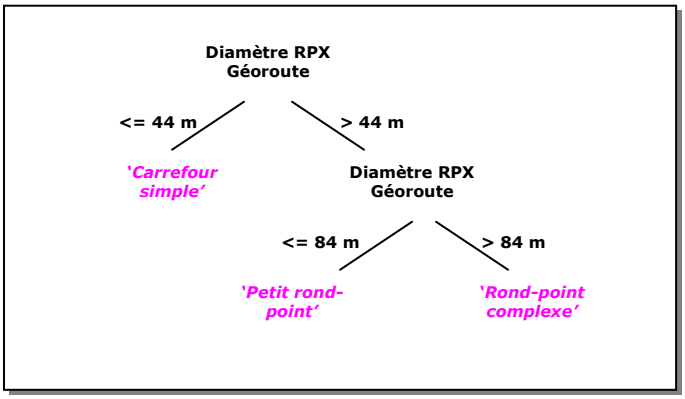


***Figure 115**. Arbre de décision appris révisé.*

Il faut noter que nous avons voulu connaître l'influence de la prise en compte des appariements incertains sur les résultats de l'apprentissage. Nous avons obtenu des règles différentes de celles présentées ci-avant. L'utilisation des erreurs d'appariement a donc une influence sur le résultat de l'apprentissage.

Ce test d'apprentissage est intéressant puisqu'il permet de découvrir l'écart toléré sur les règles de saisie appliquées sur les données par rapport aux spécifications papier. C'est la première interprétation que nous pouvons donner. Nous pouvons également penser que cet écart reflète dans une moindre mesure l'inexactitude de position des limites des ronds-points détaillés. Lors de la saisie des données, des écarts par rapport à la position nominale des limites d'objets sont inévitablement introduits. Si ces écarts sont aléatoires, l'inexactitude de position des limites n'aura que peu d'influence sur la valeur des seuils appris (la somme des écarts devrait s'annuler). En présence d'un biais de saisie (erreur systématique), on peut penser que ce bais sera pris en compte dans la valeur des seuils.

PREDICTION DES CONDITIONS RELATIVES A GEOROUTE

Les tests d'apprentissage ci-dessus ont été réalisés pour apprendre les conditions que devaient respecter les représentations de la BDCarto à partir des ronds-points de Géoroute. Il est maintenant nécessaire de travailler dans le sens inverse. Nous devons apprendre les conditions que doivent respecter les objets de Géoroute en fonction des représentations existantes dans la BDCarto.

Pour découvrir ces connaissances, nous avons utilisé les mêmes exemples d'apprentissage mais cette fois, les descripteurs concernent la BDCarto (diamètre du rond-point complexe et nature du rond-point). La classe des exemples correspond à la représentation des ronds-points dans Géoroute. Les valeurs que peut prendre cette classe classe sont cette fois limitées car les algorithmes d'apprentissage symboliques que nous utilisons n'admettent pas de classes numériques. Pour cette raison, les valeurs de la classe que nous avons attribuées aux exemples se limitent à 'rond-point simple' et 'rond-point complexe'. Nous ne pouvons pas indiquer la valeur du diamètre du rond-point de la base Géoroute lorsque celui-ci est détaillé. Par conséquent, nous ne pouvons pas apprendre les seuils que doivent respecter les ronds-points complexes dans Géoroute en fonction des représentations de la BDCarto (seuils qui apparaissent dans les règles déduites des spécifications). Pour ce faire, nous devrions définir un certain nombre de classes de diamètre par discrétisation mais nous perdrions de ce fait l'intérêt de la méthode. Les seuils qui seraient appris correspondraient aux limites des classes fixées. La définition a priori de ces limites, sans utiliser les connaissances

issues des spécifications, est délicate. Nous donnons l'arbre de décision appris en figure 116.

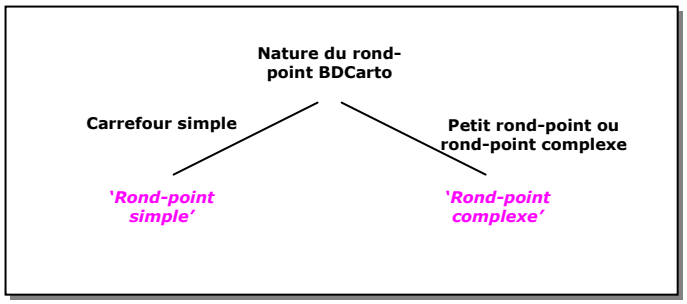


***Figure 116**. Arbre de décision appris déterminant les conditions que doivent respecter les représentations de Géoroute à partir des représentations de la BDCarto.*

ÉVALUATION ET REVISION DES CONNAISSANCES APPRISES POUR GEOROUTE

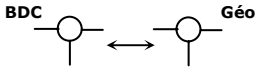
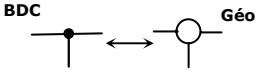
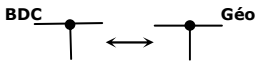
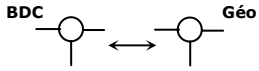
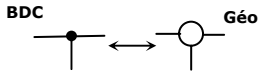
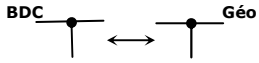
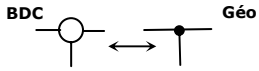
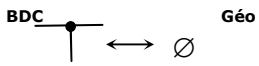
La règle permettant d'attribuer la classe 'rond-point complexe' est vraie (en écartant le fait qu'elle ne précise pas les plages de valeurs du diamètre que doit respecter l'objet dans Géoroute). L'autre règle n'est pas tout à fait juste. En effet, si un 'carrefour simple' existe dans les données de la BDCarto (et que cet objet représente un rond-point), l'objet dans Géoroute peut être un 'rond-point simple' ou un 'rond-point complexe'. Il existe donc une alternative dont il faut tenir compte. C'est ce que nous avons fait en révisant la règle (sans toutefois attribuer un seuil relatif au diamètre du rond-point complexe).

RESULTATS DE L'EVALUATION

Au terme de ce différents apprentissages, nous avons calculé les taux d'incohérences et d'équivalences en tenant compte des connaissances induites des données. Les arbres de décision ont donc été transformés en règles de production dans le système-expert et l'évaluation a été menée en tenant compte de ces connaissances apprises.

Dans le tableau ci-dessous, nous donnons les résultats de l'évaluation effectuée en exploitant les règles apprises. On remarque que le nombre d'équivalences du type « surface – surface » est plus élevé puisque le seuil du diamètre est passé de 100 m à 84 m. C'est également le cas pour la correspondance de type « point – surface » pour laquelle le seuil est passé de 50m à 44m.

***Tableau 7**. Résultats du contrôle inter-bases (approche par prédiction et apprentissage)*

Description des couples : - 613 couples jugés certains (89%) - 2 couples composés de « faux » ronds-points - 353 couples non traités à ce niveau (lien 0-1) - 258 couples évalués	
Équivalences : 184 couples (71%)	
Type d'équivalence	**Nombre de couples**
	10
	56
	118
Incohérences : 74 couples (29%)	
Type d'incohérence	**Nombre de couples**
	6
	24
	40
	1
	3

E.3.7.3 ÉTUDE DE LA RELATION ENTRE LE TAUX D'INCOHERENCES ET LE TAUX D'ERREUR REELLE ESTIME

Le taux d'erreur réelle estimé pour l'hypothèse apprise est de 29,5% pour l'arbre permettant de prédire les conditions que doivent respecter les représentations de la BDCarto. Dans l'autre sens, le taux d'erreur réelle estimé est de 33%. Ils ont été calculés par validation croisée (avec *k* fixé à 10).

Dans le chapitre D, nous avions mentionné le risque d'apprendre des règles erronées en raison de la présence de données bruitées dans le cas de la prédiction. La classe des exemples n'étant pas contrôlée à l'issue de l'appariement, toutes les incohérences inter-représentations sont utilisées pour apprendre. De ce fait, il est possible que l'hypothèse apprise tienne compte de ces correspondances incohérentes.

Nous avons pour cette raison cherché à étudier la relation existant entre le taux d'incohérences présent dans les exemples d'apprentissage (une incohérence étant vue

comme un exemple bruité du point de vue de l'apprentissage) et le taux d'erreur réelle estimé obtenu pour une hypothèse apprise à partir de ces exemples. Nous voulions comprendre si le taux d'erreur réelle estimé s'expliquait par l'erreur d'induction ou la présence des incohérences dans les données. Nous avons pu constater qu'il existait une étroite corrélation entre le taux d'erreur réelle estimé et le taux d'incohérences dans les données, et que ceux-ci évoluaient de la même manière (voir ci-dessous).

CONCEPTION D'UN GENERATEUR DE BRUIT

Pour aboutir à la conclusion que les taux d'incohérences et d'erreurs réelles sont intimement liés, nous avons conçu un générateur de bruit capable de bruiter les exemples d'apprentissage de manière réaliste. Le jeu d'exemples d'apprentissage utilisé à cette fin fut celui constitué des couples d'objets appariés jugés certains permettant d'apprendre les conditions à respecter par la BDCarto. Ce jeu a été corrigé interactivement pour n'avoir que des équivalences. L'introduction de bruit dans les exemples s'est traduite par l'attribution d'une valeur erronée à la classe de ces exemples, autrement dit, à l'affectation d'une mauvaise représentation d'un rond-point de la BDCarto pour une représentation particulière de Géoroute.

Le bruitage des exemples n'a pas été réalisé de manière aléatoire. Nous avons cherché à introduire des incohérences telles qu'elles apparaissaient dans le jeu d'exemples non corrigé (en terme de répartition). Nous avons ainsi tenu compte :

- De la distribution des diamètres des ronds-points de Géoroute ;
- De la distribution des incohérences dans le jeu d'exemples utilisé, toutes classes confondues ;
- Du nombre d'incohérences pour chaque valeur possible de la classe des exemples ('carrefour simple', 'petit rond-point', 'rond-point complexe') pour chaque plage de valeur du diamètre des ronds-points détaillés de Géoroute.

De cette manière, nous avons introduit du bruit dans les plages de valeurs de diamètre pour lesquelles les incohérences apparaissaient le plus fréquemment. De plus, pour chacune de ces plages, le bruit fut déterminé en tenant compte des confusions les plus fréquemment rencontrées (ex : entre 40 et 50m de diamètre, l'erreur la plus fréquente est d'attribuer la valeur 'petit rond-point' dans la BDCarto plutôt que la valeur 'rond-point complexe').

SIMULATIONS ET RESULTAT DU BRUITAGE

Plusieurs simulations ont été réalisées en introduisant chaque fois différents taux d'incohérences (de 0% à 42,5%). Pour chaque simulation, nous avons appliqué un algorithme d'apprentissage sur le jeu d'exemples afin d'estimer le taux d'erreur réelle associé. Les résultats obtenus sont reportés en figure 117.

La droite de régression calculée montre que le taux d'erreur réelle estimé varie globalement de la même manière que le taux d'incohérences introduit (pente positive). De plus, pour un taux d'erreur réelle estimé on obtient un taux d'incohérences identique (la valeur de Y est pratiquement égale à X). Pour ces simulations, nous avons également calculé un coefficient de corrélation positif de 0.98.

Cela nous permet de conclure que pour le jeu d'exemples d'apprentissage utilisé, puisque le taux d'erreur réelle estimé est de 29,5%, le taux d'incohérences dans les

données doit être approximativement de 30%. Autrement dit, le taux d'erreurs que l'on peut estimer lors de la mise en œuvre de l'apprentissage reflète principalement les incohérences dans les données et non l'erreur liée à l'induction. Nous pensons toutefois que pour généraliser cette conclusion à d'autres jeux d'exemples, davantage de tests devraient être effectués.

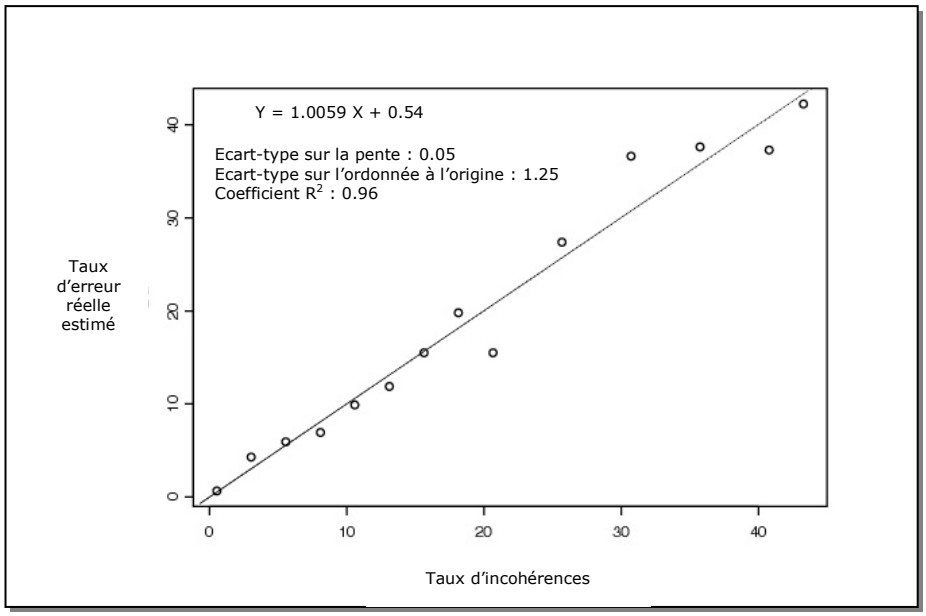


***Figure 117**. Régression linéaire calculée à partir de différents taux d'incohérence (% de données bruitées) et de taux d'erreur réelle estimé.*

E.3.8 PRESENTATION DES RESULTATS

A la suite de ces expérimentations, nous avons développé une interface permettant de visualiser les résultats de l'évaluation. Celle-ci est présentée en figure 118.

Cette interface permet de charger un jeu de couples d'objets appariés dont la cohérence a été évaluée afin d'analyser le résultat de chaque couple, comprendre pourquoi le couple a été jugé incohérent ou équivalent et, au besoin, réviser le résultat de l'évaluation. Elle est donc principalement destinée à des personnes qui seraient chargées de corriger les erreurs dans les bases avant l'intégration des données proprement dite.

La représentation des ronds-points qui est affichée est symbolique : il s'agit d'une représentation ponctuelle ou détaillée. C'est également le cas pour les représentations attendues qui peuvent être affichées grâce aux boutons correspondants. Toutefois, il est possible de visualiser le couple d'objets réels appariés dans les données en cliquant sur le bouton 'Jumelle'. Ce bouton permet d'ouvrir le « *viewer* » d'OXYGENE et d'accéder ainsi au couple et à son environnement.

Les attributs de chaque rond-point sont affichés dans le compartiment prévu à cet effet avec le résultat du contrôle intra-base. Des informations sur le couple sont également reprises : l'identifiant, la cardinalité et la confiance accordée au lien.

Le résultat du contrôle inter-bases est symbolisé par un pictogramme. Lorsque la cohérence inter-représentations n'a pas pu être évaluée (cas des liens 0-1), aucun résultat n'est affiché.

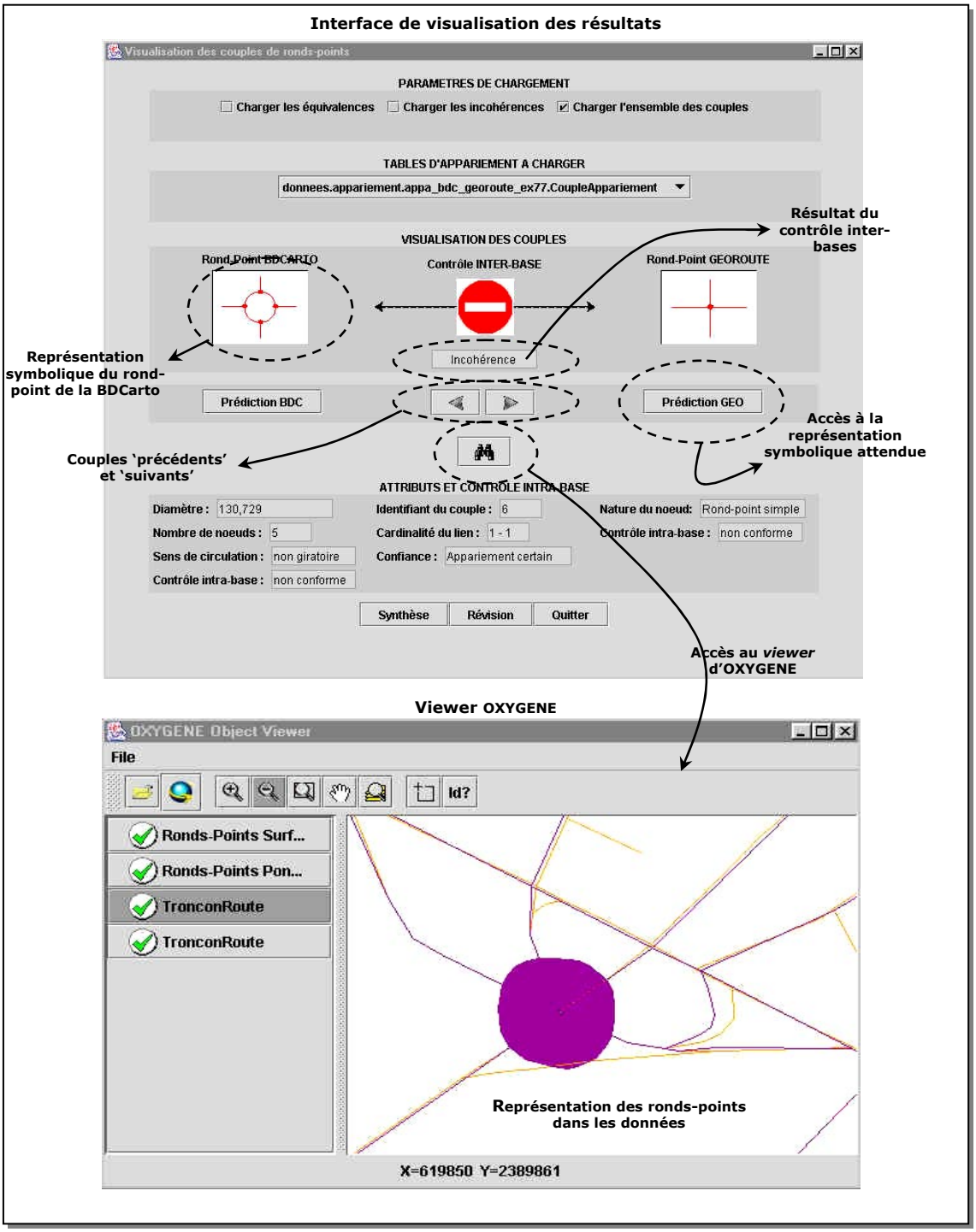


***Figure 118**. Interfaces développées permettant de visualiser les résultats de l'évaluation de la cohérence entre ronds-points.*

Deux autres boutons permettent d'accéder à deux autres interfaces : la 'synthèse' et la 'révision' (figure 119). La première affiche les résultats de l'évaluation dans leur ensemble. La seconde permet de modifier la valeur du résultat du contrôle inter-bases et de préciser la règle informelle utilisée pour aboutir à cette nouvelle interprétation. Ces connaissances pourraient être utiles pour modifier les règles introduites dans le système-expert.

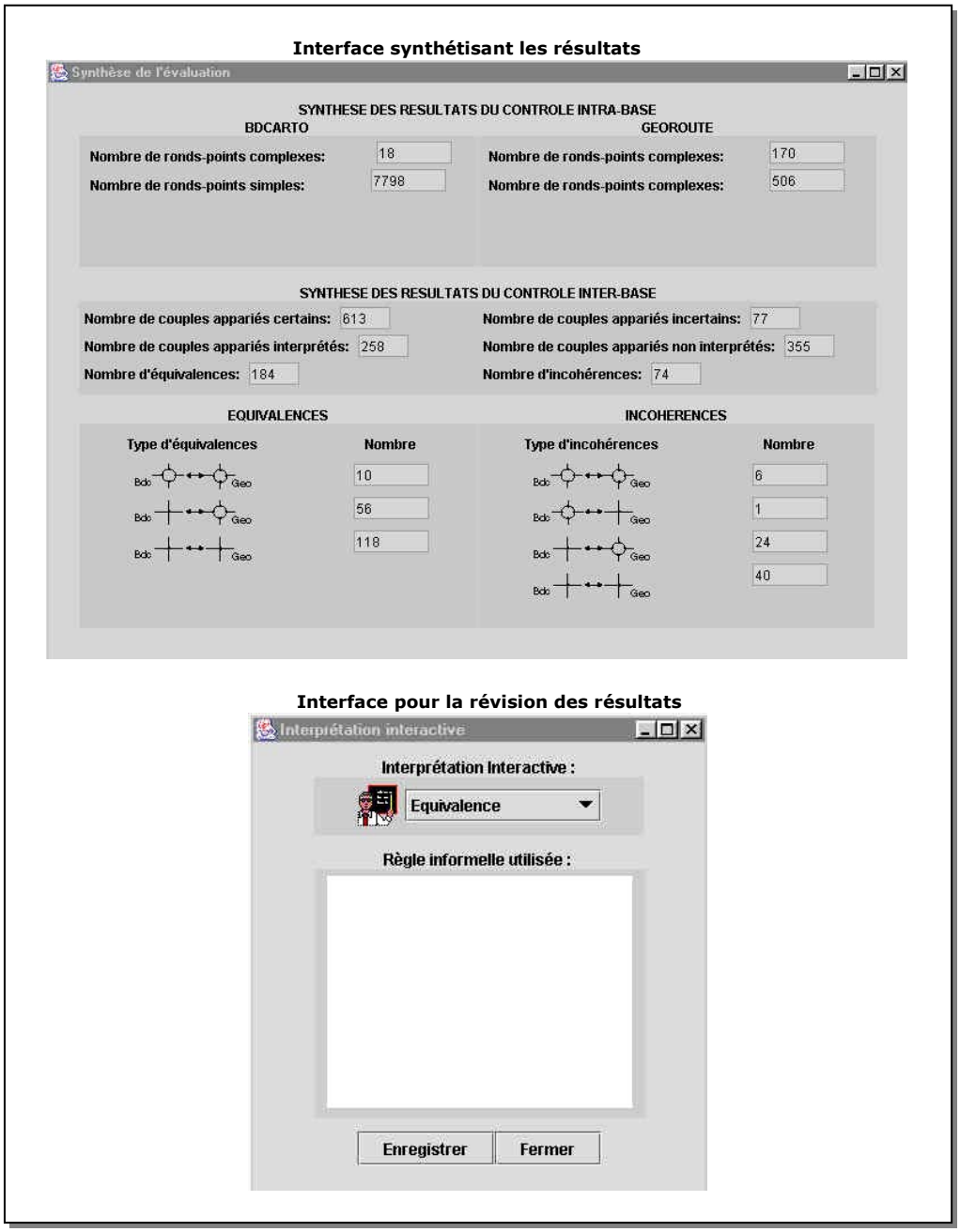


***Figure 119**. Interfaces développées fournissant une synthèse globale des résultats et permettant de les corriger au besoin.*

E.3.9 BILAN DE L'APPLICATION SUR LES RONDS-POINTS

Cette première expérimentation nous a permis d'illustrer la mise en œuvre complète de la méthodologie définie.

Dans un premier temps, nous avons pu voir que l'analyse des spécifications n'était pas une tâche triviale. Les informations à récolter pour mener l'évaluation sont souvent dispersées dans les documents papier et il arrive parfois que les spécifications se contredisent. Néanmoins, les spécifications nous ont permis de réaliser l'évaluation de la cohérence complètement. Ces métadonnées ont ainsi prouvé leur utilité.

Ensuite, nous avons montré que l'étape d'enrichissement était indispensable pour rendre possible la comparaison des données et la vérification des spécifications. Les schémas de données ont d'abord été préparés pour accueillir les nouveaux éléments dans les bases. Les données ont ensuite transformées dans une structure adaptée à leur traitement. Leur transformation a nécessité la mise au point de plusieurs algorithmes géométriques. Finalement, après avoir extrait les ronds-points détaillés des bases, nous les avons caractérisés. L'enrichissement produit ainsi une plus-value sur la quantité d'information enregistrée dans les bases exploitées.

L'étape du contrôle intra-base a suivi. Différents jeux de règles ont été utilisés pour contrôler les données automatiquement grâce au système-expert. Ces règles ont été définies durant l'étape d'analyse des spécifications (étape de *MACO*). Un certain nombre d'erreurs intra-base a ainsi pu être détecté avant la mise en correspondance des données.

L'appariement a fait appel à deux méthodes différentes : une pour calculer les liens et l'autre pour les évaluer. Nous avons retenu les couples pour lesquels nous obtenions les mêmes résultats dans les deux cas. Les erreurs d'appariement constatées nous ont permis de comprendre que chaque méthode avait ses avantages et ses inconvénients. Celles-ci devraient être regroupées pour ne former qu'une seule méthode d'appariement. Cette dernière fournirait probablement des résultats très performants.

Le contrôle inter-bases a ensuite été réalisé. Nous avons d'abord exploité les connaissances déduites des spécifications. Un contrôle automatique fut mis au point. Nous avons appliqué les règles introduites dans les système-expert pour classer les couples retenus en incohérence et équivalence. L'approche par prédiction fut adoptée.

Parallèlement à cette première analyse, nous avons souhaité induire les connaissances contenues dans les données qui pourraient également servir à mener le contrôle inter-bases. Nous avons ainsi réalisé plusieurs tests d'apprentissage et mis en évidence les écarts existant entre ce que les spécifications préconisent de saisir et ce que les données contiennent réellement. Nous avons pu tirer plusieurs enseignements des résultats de l'apprentissage :

- L'apprentissage supervisé symbolique constitue un bon moyen d'extraire des connaissances intelligibles issues des données pour mener l'évaluation ;
- Les règles apprises doivent toutefois être systématiquement examinées avant d'être exploitées. Le taux d'erreur réelle estimé ne doit pas constituer la référence unique pour considérer que les règles sont valables ou non, d'autant plus que ce taux peut-être élevé si les données contiennent de nombreuses incohérences. L'intervention de l'expert est donc systématiquement requise.
- D'autres algorithmes d'apprentissage devraient être également exploités. En particulier, des algorithmes acceptant d'apprendre des classes constituées de valeurs numériques et fournissant des règles interprétables. Ils permettraient d'affiner certaines hypothèses apprises.

Pour conclure, il est possible que la méthode d'évaluation laisse passer certaines erreurs d'évaluation. Celles-ci peuvent se produire car les étapes d'enrichissement des bases et l'appariement, si elles sont automatisées, ne sont jamais parfaitement maîtrisées. Dans la mesure du possible, il faut donc mettre en place des mécanismes qui contrôlent les résultats de ces étapes afin de limiter le nombre d'erreurs et garantir une évaluation correcte.

E.4 ÉTUDE DES DIFFERENCES ENTRE REPRESENTATIONS DE BATIMENTS

E.4.1 MOTIVATIONS

La seconde étude que nous avons réalisée pour mettre en œuvre la méthodologie définie concerne les différences de représentation entre bâtiments de la BDTopo standard et la BDCarto de l'IGN. L'intérêt de cette application est de montrer que la démarche d'évaluation de la cohérence peut être appliquée sur des données présentant des niveaux de détails différents. Par ailleurs, cette expérimentation illustre à nouveau le besoin de recourir à l'analyse spatiale pour enrichir les données des bases et construire des exemples d'apprentissage pertinents. Nous verrons que l'acquisition de connaissances par apprentissage s'est imposée pour ces tests en raison de l'insuffisance des spécifications. Cette application met également en jeu des données qui présentent des actualités différentes. Des différences de mise à jour entre les bases sont donc susceptibles d'apparaître.

E.4.2 PRESENTATION DES BASES

La première source de données exploitées est la BDTopo standard. La BDTopo est une base de résolution métrique. Elle provient de la restitution de photographies aériennes et a été notamment définie pour produire les cartes topographiques à l'échelle du 1/25.000. Il s'agit du produit vecteur le plus détaillé de l'IGN. Cette base a aujourd'hui évolué et correspond à la BDTopo Pays qui fait partie du Référentiel à Grande Échelle (RGE). Le thème que nous avons utilisé pour effectuer nos tests (les bâtiments) date de 1998.

L'autre source de données utilisées est la BDCarto. Les caractéristiques de cette base ont déjà été présentées dans la partie précédente (cf. E.2.2). Les zones d'habitat font partie du thème relatif à l'occupation du sol. Elles proviennent d'une interprétation d'images SPOT dont l'actualité est comprise entre 1989 et 1993.

Notre zone d'étude touche la région d'Orléans. Plusieurs jeux de données ont été extraits. Ils couvrent au total une superficie de 9000 hectares environ. La figure 120 illustre la représentation des bâtiments des deux bases.

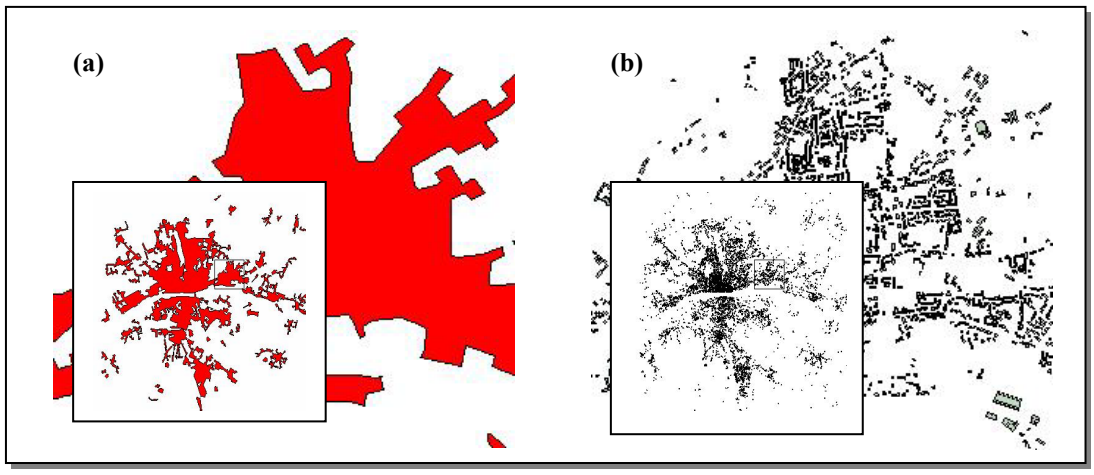


***Figure 120**. Illustration des zones d'habitat de la BDCarto (a) et des bâtiments de la BDTopo (b)*

E.4.3 ANALYSE DES SPECIFICATIONS

Comme notre méthodologie le préconise, la première tâche à réaliser avant de mettre en œuvre le processus d'évaluation est l'analyse des spécifications de chaque base. Nous détaillons ci-dessous les informations contenues dans ces documents.

E.4.3.1 SPECIFICATIONS DE LA BDCARTO

Dans la BDCarto, les bâtiments ne sont pas individualisés. Ceux-ci sont agrégés pour former une zone d'espace bâtie. Celle-ci appartient à la classe *Zone d'occupation du sol* de géométrie surfacique. L'attribut « poste » permet de différencier l'espace bâti des autres phénomènes (zone industrielle, commerciale, zone agraire, forêts et espace semi-naturel, etc.). Les zones d'occupation du sol sont sélectionnées suivant des critères de superficie minimale. Pour l'espace bâti, cette limite est fixée à 8 hectares. La définition de ce poste est la suivante [BDCarto 2001] :

« *Surface à prédominance d'habitat :*

- *Tissu urbain dense, noyaux urbains et faubourgs anciens, bâtiments formant un tissu homogène et continu, y compris les équipements divers inférieurs à 25 hectares ;*
- *Tissu urbain continu mixte, habitat pavillonnaire ou continu bas avec jardins ;*
- *Type faubourg, associant quelques petits secteurs d'activités ;*
- *Grand ensembles, lotissements, cités jardins ;*
- *Villages et hameaux importants en milieu agricole y compris les aménagements associés ;*
- *Cimetière voisin de bâti ou de plus de 8 hectares. »*

S'ajoute à cette définition un certain nombre de règles précisant la manière de sélectionner et de représenter les objets :

« *Bâti :*

- *Villages rues : ils forment une bande continue d'au moins 50 mètres de large sur au moins 1600 mètres de long ;*
- *Petites parcelles de bâti (surfaces inférieures à 8 hectares) : elles sont regroupées si elles sont distantes les unes des autres de moins de 100 mètres, de manière à atteindre les 8 hectares.*
- *Les bâtiments divers : écoles, lycées, universités, hôpitaux, casernes... sont classés dans le poste 11 (bâti).*
- *Cimetières : ils sont classés dans le poste 11 (bâti), sauf quand ils sont isolés et de moins de 8 hectares ; ils sont alors classés dans le thème avoisinant. Cas particuliers : les cimetières (cimetière militaire américain ou autre) de plus de 25 hectares sont classés dans le poste 21 (pelouse) ; les cimetières paysagers où le bois couvre une surface de 8 hectares ou plus, sont classés dans le poste 31 (forêt).*
- *Les parcs, bois, et forêts inférieurs à 8 hectares et associés ou inclus à une zone de bâti de plus de 8 hectares sont classés dans le poste 11 (bâti).*

- *Serres : elles sont classées dans le poste 11 (bâti) quand elles sont incluses dans une zone de bâti ; dans les autres cas, elles sont classées dans le poste 21 (culture) et 22 (verger).* »

Certaines spécifications relatives aux bâtiments apparaissent encore dans la partie des documents consacrés aux zones industrielles, commerciales, de communication ou de loisirs. Ces zones industrielles et commerciales ne sont reprises sous ces termes que si leur superficie est supérieure à 25 hectares. Dans le cas contraire, on note que [BDCarto 2001] :

« *Les surfaces inférieures à 25 hectares et liées à du bâti supérieur à 8 hectares sont incluses dans le poste 11 (bâti).* »

E.4.3.2 SPECIFICATIONS DE LA BDTOPO

L'espace bâti dans la BDTopo est représenté différemment. Tous les bâtiments sont individualisés. Ils sont modélisés par plusieurs classes mais toutes les classes ne sont pas à mettre en correspondance avec l'espace bâti de la BDCarto. Les classes à exploiter sont les suivantes : *Bâtiment quelconque*, *Bâtiment remarquable*, *Construction spéciale*, *Bâtiment religieux*, *Salle de sport*, *Tribune, Cimetière*, *Serre*, *Construction légère*. Les classes *Bâtiment industriel et commercial, Enceinte industrielle, Enceinte commerciale, Silo* sont également concernées lorsque les éléments s'y rapportant ne forment pas un ensemble de bâtiments inférieur à 25 hectares. Tous les bâtiments individualisés de la BDTopo ne sont donc pas pris en compte pour notre étude. Seule l'analyse croisée des spécifications permet de s'en rendre compte.

En principe, nous devrions également inclure pour nos tests des objets de la classe *Occupation du sol* de la BDTopo puisque les zones d'espace bâti de la BDCarto sont des surfaces généralisées qui incluent des éléments comme les parcs, les bois et les forêts si ceux-ci sont inférieurs à 8 hectares et sont compris dans l'espace bâti. En pratique, nous avons agrégé les bâtiments de la BDTopo (cf. *supra*) de sorte qu'il n'est pas nécessaire de tenir compte d'autres éléments de l'occupation du sol. Les espaces vides entre les bâtiments ont été comblés et lorsque des éléments de l'occupation du sol de type 'bois' ou 'forêt' par exemple bordent les bâtiments (en dehors du groupe), ils ne sont généralement pas inclus dans l'espace bâti de la BDCarto (figure 121).

La majorité des objets « bâtiment » de la BDTopo sont issus de la classe *Bâtiment quelconque*. Il s'agit de « *bâtiment en 'dur' dont l'architecture ou l'aspect n'est pas industriel, agricole ou commercial. La modélisation est de type surfacique. En règle générale, les bâtiments ne sont pas généralisés, leur individualité est conservée jusqu'aux limites de la précision planimétrique (1m). On saisit le pourtour extérieur et les cours intérieures. Les cours intérieures sont saisies seulement lorsque la plus grande dimension de celles-ci est au moins de 25m et la plus petite dimension au moins de 10m. L'objet surfacique est alors troué* » [BDTopo 1994].

Les règles de saisie des autres classes citées correspondent la plupart du temps à des contraintes sur la nature des bâtiments. On trouve par exemple les spécifications suivantes : « les bâtiments de type industriel, agricole ou commercial regroupent les ateliers, hangars, entrepôts, supermarchés,… » ; « On ne saisit que les serres pérennes et construites » ; « les bâtiments remarquables regroupent les moulins à vent, les pigeonniers, les tours, les donjons,… ». Elles sont assez peu exploitables pour l'évaluation de la cohérence.

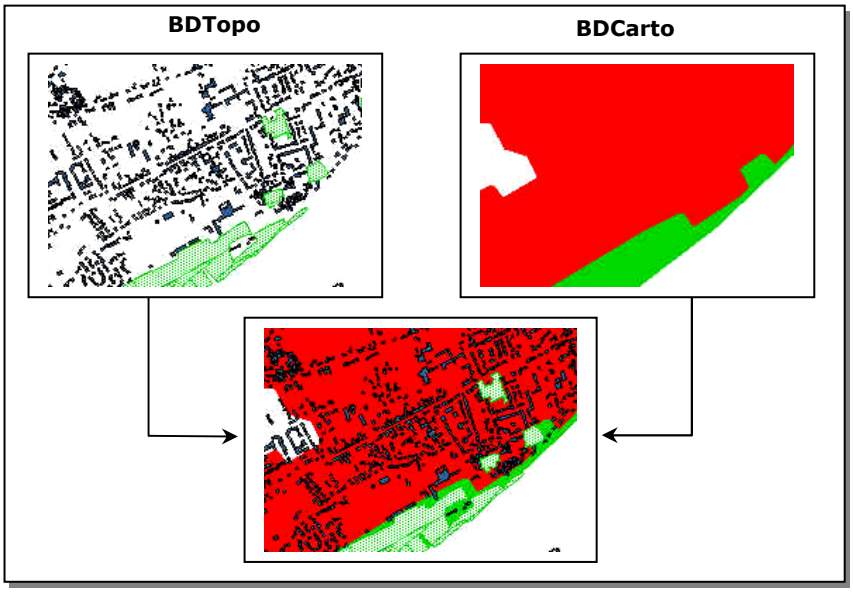


***Figure 121**. Les petites zones en bordure de l'espace bâti ne font pas partie des zones bâties de la BDCarto.*

L'analyse des spécifications nous a permis de comprendre les différences de représentation entre les données susceptibles d'apparaître. Nous avons pu déterminer les outils à utiliser pour enrichir et apparier les données (nous les présenterons ci-après). Par contre, au terme de cette étape, aucune règle n'a pu être formulée pour réaliser le contrôle inter-bases. Seule un règle relative au contrôle intra-base de la BDCarto a été définie. Comme nous le verrons, l'étape d'apprentissage s'est donc imposée.

E.4.4 ENRICHISSEMENT

Suite à l'analyse des spécifications, nous avons préparé les données des bases pour permettre l'évaluation de la cohérence. Cette préparation s'est traduite par un enrichissement et une restructuration des schémas et des données (1ère étape de la méthode *MECO*).

E.4.4.1 ENRICHISSEMENT DES SCHEMAS

SCHEMA ENRICHI DE LA BDCARTO

Pour la BDCarto, l'enrichissement du schéma s'est traduit par la création d'une nouvelle classe *Espace Bâti* qui hérite de la classe *Zone d'occupation du sol* initiale. Cette classe permet d'isoler les zones d'occupation du sol qui présentent un intérêt pour l'étude. Nous lui avons affecté un attribut « superficie » qui servira pour le contrôle intra-base (figure 122).

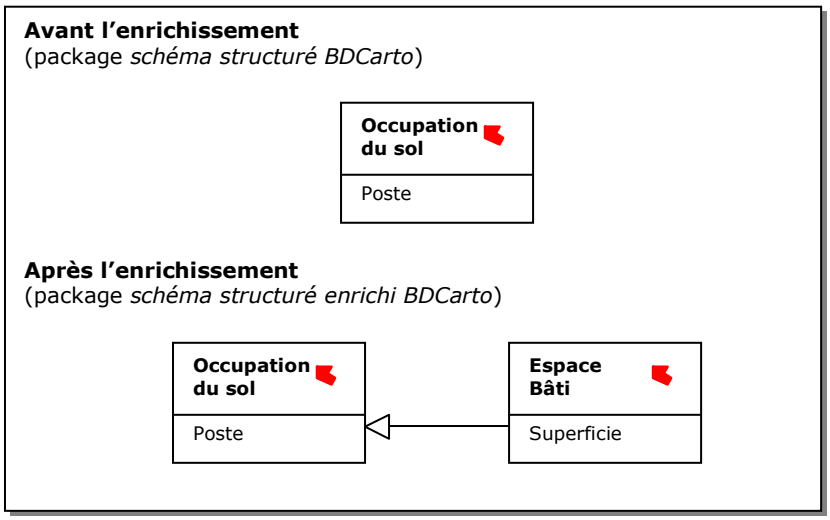


***Figure 122**. Enrichissement du schéma de la BDCarto : création d'une classe « Espace bâti ».*

SCHEMA ENRICHI DE LA BDTOPO

Pour la BDTopo, nous avons introduit une classe *Zone d'espace bâti* dans le schéma qui correspond à une agrégation de l'ensemble des bâtiments individualisés des différentes classes existantes. Cette classe a une représentation surfacique qui généralise l'ensemble des représentations de bâtiments individualisés. Elle est dotée d'un attribut « superficie » (figure 123).

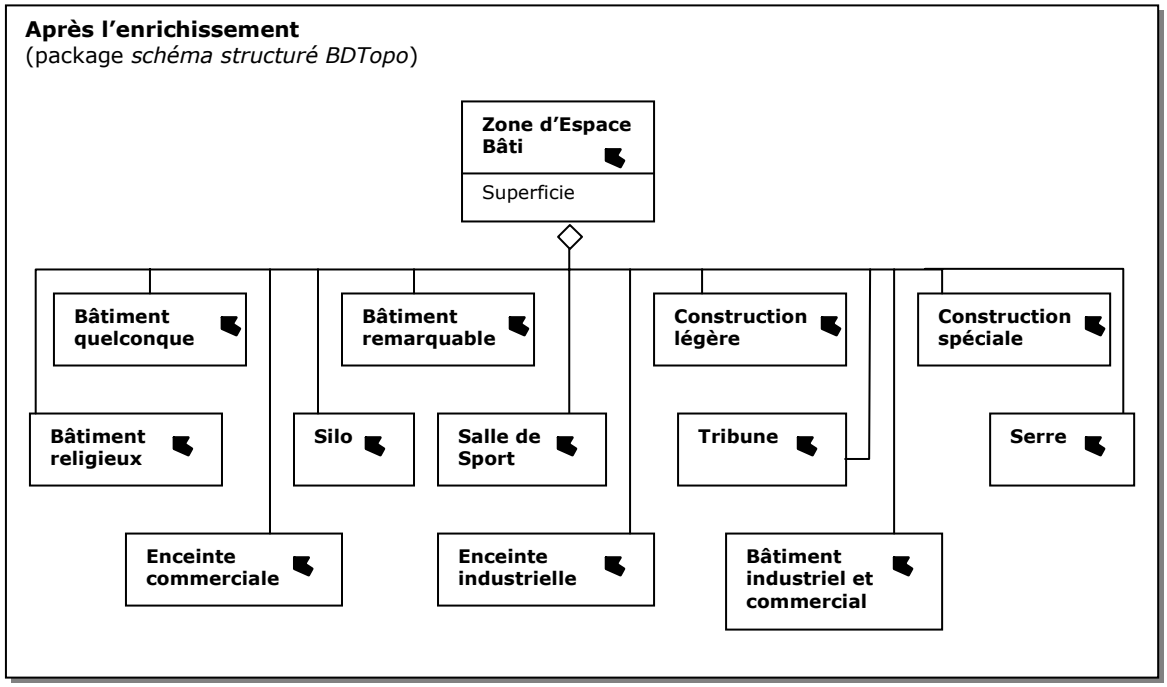


***Figure 123**. Enrichissement du schéma de la BDTopo : création d'une classe « Zone d'espace bâti ».*

E.4.4.2 ENRICHISSEMENT DES DONNEES

Après la définition de ces nouvelles classes, nous les avons instanciées. Pour la BDCarto, la tâche est très simple. Elle consiste à sélectionner tous les éléments de la classe *Zone d'occupation du sol* dont l'attribut « poste » a pour valeur 'bâti' et à

importer ces objets surfaciques dans la classe *Espace Bâti*. L'attribut « superficie » peut ensuite être calculé.

Concernant la BDTopo, l'enrichissement est moins immédiat. Il est nécessaire de construire des objets composés dans la classe *Zone d'espace bâti* à partir des différents types de bâtiments individualisés. Pour aboutir à ces objets composés BDTopo, nous avons exploité les spécifications de la BDCarto. Ceci s'explique par le fait que nous souhaitions créer des zones d'espace bâti qui se rapprochent de l'univers de la BDCarto pour pouvoir comparer les représentations lors du contrôle inter-bases et évaluer ainsi leur cohérence. La transformation a été réalisée de la manière suivante : toutes les maisons distantes de moins de 100m ont été regroupées (cf. spécifications de la BDCarto). Le regroupement s'est fait à l'aide de *zones tampons* correspondant à une expansion des bâtiments d'un rayon égal à 50m (figure 124).

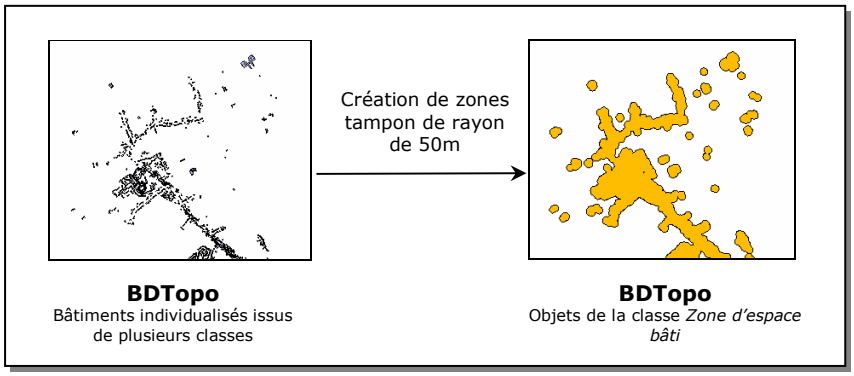


***Figure 124**. Instanciation de la classe « Zone d'espace bâti » de la BDTopo.*

Les maisons agrégées sont principalement issues de la classe *Bâtiment Quelconque.* Quelques bâtiments remarquables et religieux ont également été sélectionnés. Quant aux bâtiments industriels et commerciaux, pour savoir s'ils devaient être pris en compte, nous avons vérifié qu'ils n'étaient pas associés à une zone d'occupation du sol de type industriel ou commercial dans la BDCarto (figure 125).

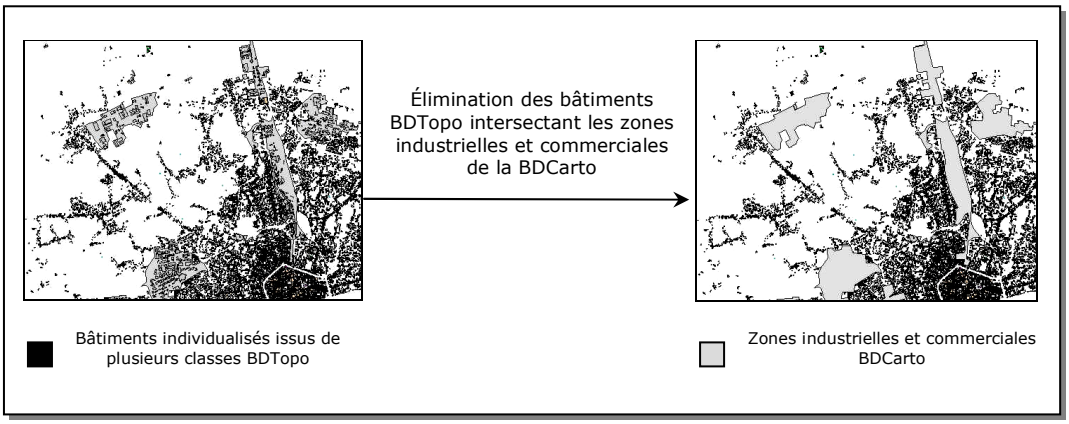


***Figure 125**. Élimination de bâtiments BDTopo appartenant à une zone d'occupation du sol de la BDCarto non traitée.*

Une sélection automatique des bâtiments individualisés a donc été effectuée avant la création des zones tampons. Elle a consisté à soustraire tous les bâtiments de l'ensemble de la BDTopo qui n'intersectaient pas une zone d'occupation du sol de type

'bâti industriel ou commercial' de la BDCarto. Au terme de la création des zones d'espaces bâti de la BDTopo, l'attribut « superficie » a été calculé.

E.4.5 CONTROLE INTRA-BASE

Après l'enrichissement des bases, nous avons réalisé le contrôle intra-base (2ème étape de *MECO*). Cette tâche fut très sommaire pour cette expérimentation. En effet, pour la BDCarto, nous avons seulement pu vérifier que les zones d'espace bâti couvraient bien une superficie supérieure à 8 hectares. Nous ne pouvons pas déterminer à ce niveau si certains bâtiments avaient été pris en compte pour définir les limites de la zone alors qu'ils n'auraient pas dû l'être ou inversement, si des groupes de bâtiments formant une entité supérieure à 8 hectares avaient été oubliés.

Pour la BDTopo, les contraintes de nature qui dictent la sélection des objets n'ont pas pu être vérifiées à ce niveau. Les erreurs de déficit et d'excédent n'ont donc pas été détectées. En principe, elles le sont lors du contrôle inter-bases. Toutefois, dans le cas présent, le contrôle inter-bases ne pourra être que partiel. En effet, si un bâtiment individuel venait à manquer à l'intérieur d'une zone d'espace bâti, nous ne serions pas à même de justifier l'espace libéré par cette absence. Nous ne pourrions pas savoir s'il s'agit d'un oubli ou si l'espace libre est normal dans la BDTopo puisque la zone de la BDCarto correspondante est une surface généralisée.

Le contrôle intra-base n'a mis en évidence aucune erreur à ce niveau. Les zones de la BDCarto ont une superficie largement supérieure à 8 hectares.

E.4.6 APPARIEMENT

L'établissement des liens entre les objets des deux bases peut être envisagé de deux manières différentes. La première peut prendre en compte les objets zonaux de la BDCarto et les bâtiments individualisés de la BDTopo. La seconde peut porter sur les deux ensembles de représentations zonales : les objets de la classe *Espace bâti* de la BDCarto et les objets créés dans la classe *Zone d'espace bâti* de la BDTopo. Pour cette application, nous avons préféré réaliser un appariement en utilisant les bâtiments individualisés. Nous justifions ce choix par le fait qu'il peut exister des groupes de bâtiments appartenant à un objet de la classe *Zone d'espace bâti* de la BDTopo qui n'ont pas de représentation homologue dans la BDCarto bien que les objets zonaux des deux bases soient reliés entre eux (figure 126). Par conséquent, puisqu'un appariement entre objets zonaux seuls impliquerait d'associer tous les bâtiments individualisés de la BDTopo à la zone de la BDCarto, cette approche nous a semblé moins bien adaptée. Elle aurait nécessité d'analyser la forme des objets. La première approche est plus facile à mettre en œuvre.

La méthode d'appariement mise en œuvre fut donc très simple. Nous avons considéré que chaque bâtiment individualisé de la BDTOPO était apparié à une zone de la BDCARTO s'il intersectait celle-ci. A son terme, nous avons obtenu deux types de liens : des liens 1-n (une zone BDCarto est associée à un ensemble de bâtiments BDTopo) et 0-1 (un bâtiment BDTopo n'est pas apparié à une zone BDCarto).

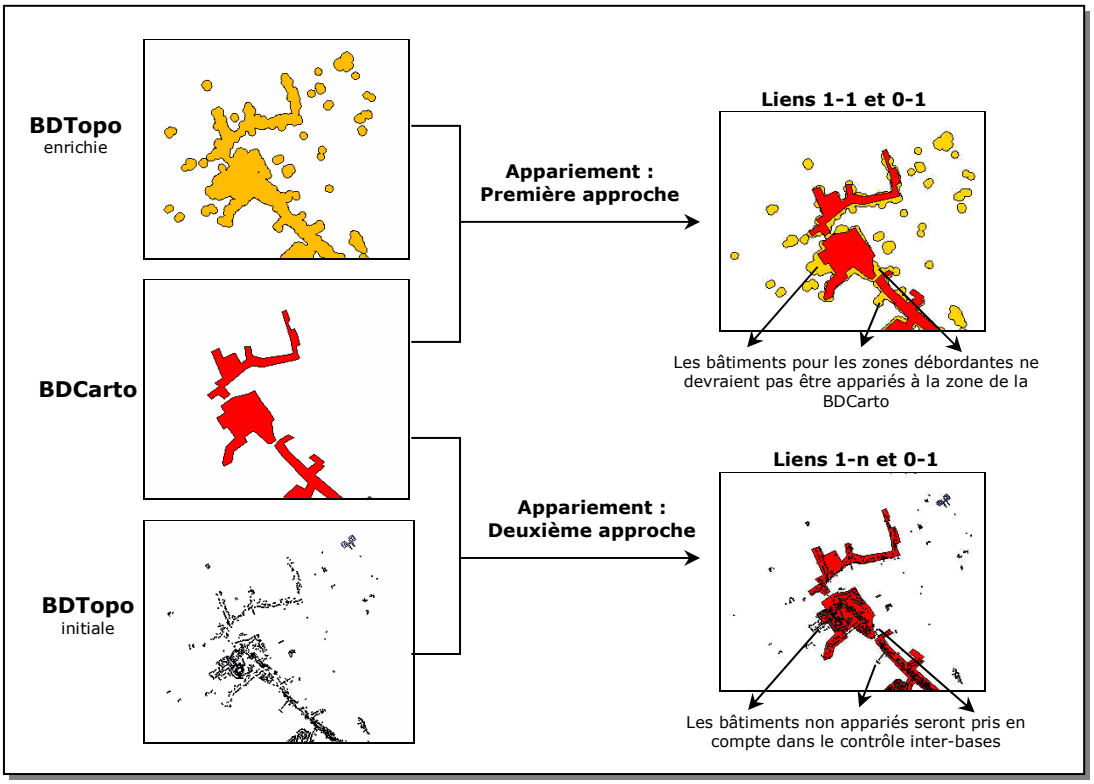


***Figure 126**. Deux approches peuvent être utilisées pour apparier les objets. La seconde a été retenue.*

E.4.7 CONTROLE INTER-BASES

E.4.7.1 DETECTION DES DIFFERENCES DE REPRESENTATION A EVALUER

Les données étant appariées, l'étude de la cohérence inter-représentations a ensuite débuté. La question qui se pose à ce niveau est de savoir quoi comparer. Quelles sont les différences de représentation à interpréter ?

Pour les mettre en évidence, la première tâche réalisée a consisté à détecter un certain nombre d'équivalences parmi les couples calculés (contrôle inter-bases - étape de *MECO*). D'après les spécifications de la BDCarto, il apparaît que des groupes de bâtiments ne formant pas une entité d'une superficie supérieure à 8 hectares ne sont pas saisis dans la base. Cela signifie que tous les groupes de bâtiments créés dans la BDTopo qui ne répondent pas à ce critère correspondent à des équivalences. Il est normal que ces éléments n'apparaissent pas dans la BDCarto. Pour les détecter, nous avons vérifié la superficie de tous les objets de la classe *Zone d'espace bâti* et sélectionné tous les objets dont l'étendue est inférieure à 8 hectares. La représentation abstraite des bâtiments a donc été exploitée (figure 127).

Ensuite, nous avons cherché à extraire tous les groupes d'objets non appariés, et ceci dans les deux directions : liens 0-1 et 1-0. Nous avons donc d'abord classé tous les bâtiments individualisés appariés avec une zone de la BDCarto comme des représentations équivalentes pour découvrir ainsi tous les bâtiments de la BDTopo non appariés à la BDCarto. Ces bâtiments constituaient les différences à interpréter. Il pouvait s'agir d'équivalences, d'incohérences ou de mises à jour (figure 127). Nous avons également soustrait à la zone BDCarto, les zones d'espace bâti de la BDTopo afin de découvrir les portions de zones BDCarto non appariées à la BDTopo. En

principe, dans ce sens, d'après les spécifications des bases, il ne doit pas exister de zones non représentées dans la BDTopo, sauf erreur de saisie dans celle-ci (les différences d'actualité n'intervenant pas ici, la BDTopo étant plus récente). Nous avons quand même mis en évidence un petit nombre de zones de la BDCarto qui n'avaient pas de correspondant dans la BDTopo (figure 127).

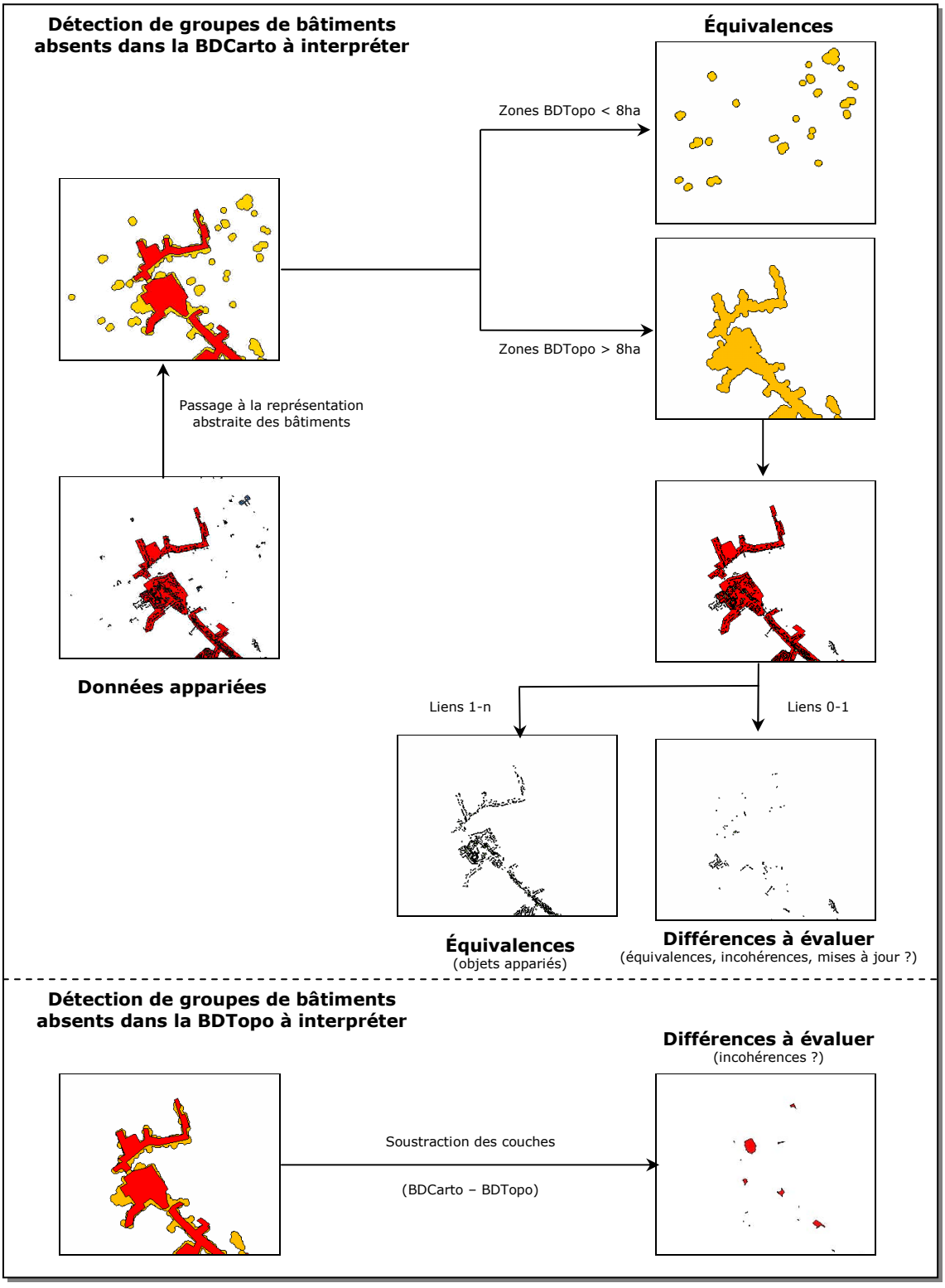


***Figure 127**. Détection des différences de représentation à interpréter dans les deux bases.*

Concernant les zones de la BDCarto détectées sans correspondant dans la BDTopo, nous nous sommes aperçus qu'il ne s'agissait pas de déficit. Il existait bien des bâtiments individualisés dans la BDTopo à ces endroits mais ils étaient classés en 'salle de sport' et 'construction spéciale' et, de ce fait, n'avaient pas été sélectionnés lors de l'enrichissement, bien qu'ils auraient dû l'être. Par conséquent, il ne s'agissait pas d'incohérences. Toute l'étendue des zones de la BDCarto était bien recouverte de bâtiments BDTopo, élimination faite des polygones parasites liés aux différences de construction des objets polygonaux. Il restait donc seulement à expliquer les absences dans la BDCarto de bâtiments individualisés non appariés présents dans la BDTopo.

E.4.7.2 CONSTRUCTION D'UNE BASE DE REGLES PAR APPRENTISSAGE

Pour justifier les différences restantes, l'apprentissage de règles s'est imposée (étape de *MACO*). Les connaissances issues des spécifications n'étaient en effet plus suffisantes. Nous avions déjà exploité toute l'information qui pouvait servir à classer les différences et nous n'étions pas en mesure de définir des règles d'évaluation manuellement (cf. analyse des spécifications). De ce fait, nous avons cherché à construire des exemples d'apprentissage pour apprendre.

La mise en œuvre de l'apprentissage a soulevé le problème important de la définition de la forme des exemples : quelle représentation fallait-il adopter pour rendre possible l'apprentissage ? Pris un à un, les bâtiments ne constituaient pas des exemples pertinents. Aucun descripteur n'aurait permis de différencier par exemple un bâtiment résultant d'une mise à jour (différence d'actualité), d'un bâtiment existant dans une base mais absent dans l'autre en raison des différences de résolution des bases (équivalence). Pour pouvoir distinguer ces cas, il était nécessaire d'agréger les bâtiments de manière à tenir compte du fait que certains d'entre eux appartenaient à un groupe. Nous pouvions ainsi obtenir des représentations plus adéquates pour le processus d'apprentissage supervisé.

RECUEIL D'EXEMPLES D'APPRENTISSAGE

Deux méthodes différentes ont été mises en œuvre pour construire les exemples d'apprentissage. La première méthode d'agrégation fait à nouveau appel la construction de zones tampons. Nous avons regroupé les bâtiments de la BDTopo en définissant des zones tampons de 50 m de rayon et fusionné les éléments ayant des frontières connectées. Nous avons ensuite procédé à leur érosion (création d'une zone tampon inversée de 35 m de rayon) pour élaborer le plus grand nombre de groupes homogènes (du point de vue de l'origine de l'absence des bâtiments dans la BDCarto). Nous avons ainsi obtenu un ensemble d'agrégats que nous avons ensuite caractérisé (en faisant appel à des mesures) et classé interactivement pour construire les exemples d'apprentissage (figure 128).

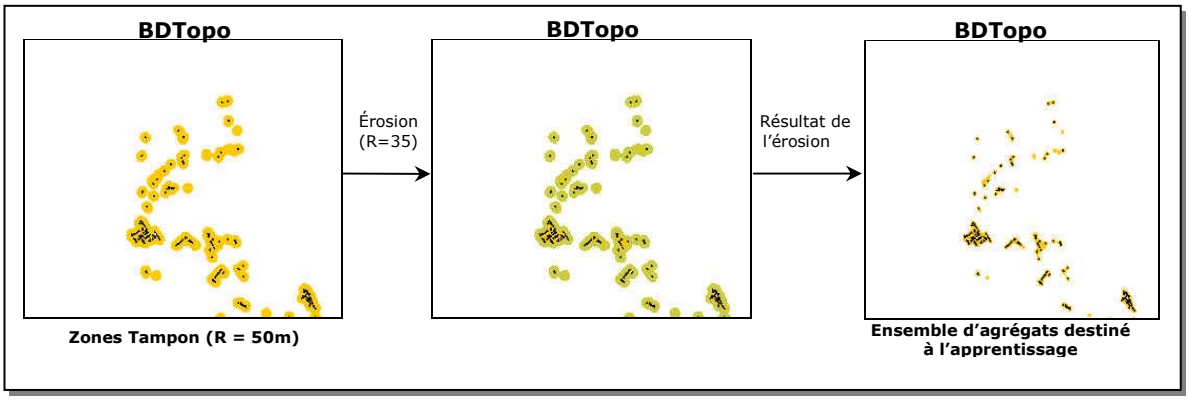


***Figure 128**. Construction d'exemples d'apprentissage : première méthode mise en œuvre.*

Après analyse des différents groupes, il nous a semblé nécessaire d'avoir recours à une autre méthode d'agrégation. Malgré la phase d'érosion, les objets créés présentaient une trop forte hétérogénéité. Plusieurs bâtiments avaient été agrégés alors que certains auraient dû rester séparés (figure 129). Ceci était particulièrement gênant pour la classification des groupes car certains d'entre eux mélangeaient à la fois des équivalences, des incohérences ou des mises à jour. Nous avons donc développé une autre méthode de regroupement.

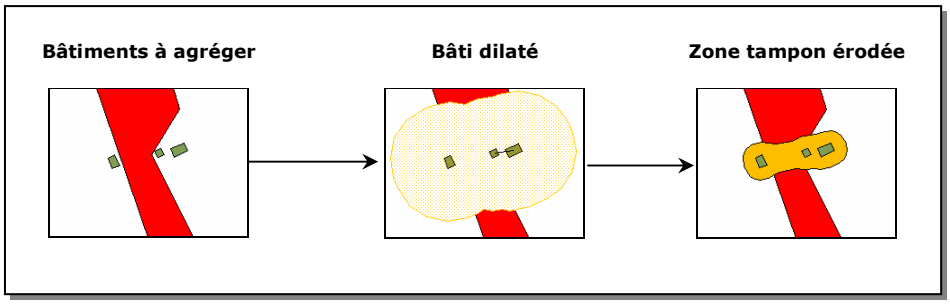


***Figure 129**. Défaut de la première méthode de construction des exemples d'apprentissage*

La seconde méthode mise en œuvre est fondée sur l'utilisation d'une triangulation de Delaunay. Elle peut être comparée aux approches de « *clustering* » basées sur la création de graphes [Anders et al. 1999 ; Estivill-Castro et Lee 2002]. Après avoir déterminé le centre de gravité de chaque bâtiment (uniquement ceux appartenant aux différences à évaluer), nous avons calculé une triangulation de Delaunay sur l'ensemble de ces points (figure 130). Ensuite, afin de créer des groupes homogènes, nous avons filtré ce graphe en utilisant deux critères : un critère de longueur sur les arêtes et un critère d'intersection avec la BDCarto. Toutes les arêtes de longueur supérieure à 115 m (seuil défini empiriquement) ont été supprimées, de même que les arêtes intersectant la BDCarto.

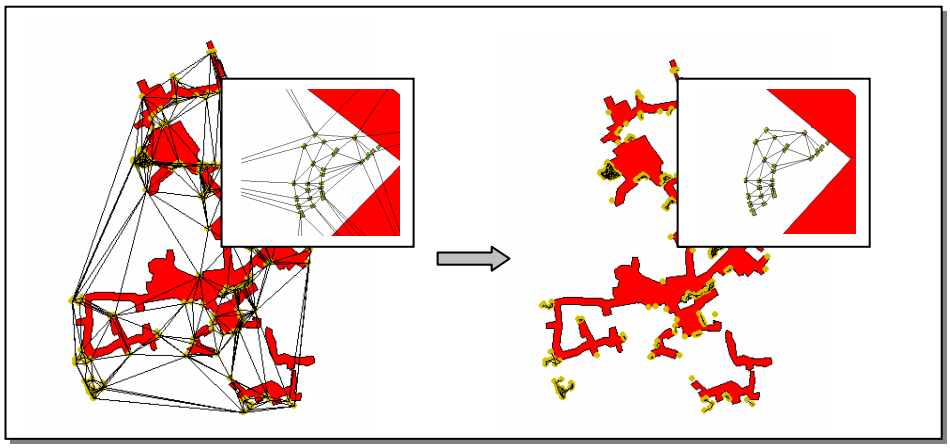


***Figure 130**. Constitution de groupes de bâtiments à partir du triangulation de Delaunay.*

Les défauts de construction de la première méthode ont ainsi été évités (figure 131). Nous avons finalement obtenu un groupe d'agrégats sur lesquels une zone tampon de 15 m de rayon a ensuite été appliquée. La représentation des exemples d'apprentissage était ainsi définie.

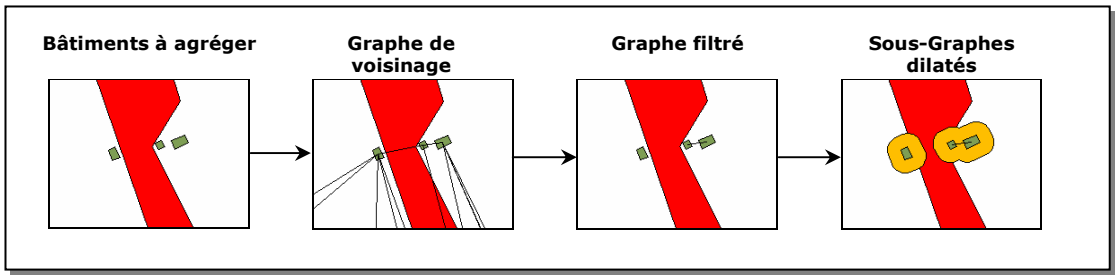


***Figure 131**. La seconde méthode de construction des exemples d'apprentissage élimine les défauts de la première méthode.*

Cette application a été mise en œuvre dans la plate-forme OXYGENE. Les différentes classes relatives à la triangulation ont été définies sur base du *package* de la Carte Topologique (relation d'héritage – spécialisation de la carte topologique). L'algorithme de triangulation proprement dit, *Triangle*, n'a pas été développé par nos soins. Il a été récupéré des travaux de [Shewchuk 1996] et interfacé dans notre environnement de travail.

Une fois la représentation des exemples définie, ceux-ci ont été caractérisés par plusieurs descripteurs. Nous avons décrit chaque groupe créé à l'aide de 8 mesures pour un total de 183 exemples :

- La superficie du groupe ;
- Le périmètre du groupe ;
- La densité des bâtiments individualisés dans le groupe ;
- Le nombre de maisons individualisées dans le groupe ;
- La distance entre le centre de gravité du groupe et la zone d'espace bâti la plus proche (BDCarto) ;
- La distance entre le bâtiment du groupe le plus proche de la zone d'espace bâti (BDCarto) et cette zone d'espace bâti ;

- La distance entre le bâtiment du groupe le plus éloigné de la zone d'espace bâti (BDCarto) et cette zone d'espace bâti ;
- La compacité du groupe. Nous avons utilisé l'indice suivant (proche de l'indice de circularité de Miller) [Coster et Chermant 1989] :

$$I_c = 16\frac{S}{P^2}$$

S représente la surface de l'entité et P correspond à son périmètre.

APPRENTISSAGE DE REGLES PAR CLASSIFICATION DIRECTE

Pour que les exemples d'apprentissage soient complètement décrits, il restait à les étiqueter. Cette fois, l'attribution de l'étiquette n'a pas été automatique. Nous n'avons pas pu adopter l'approche par prédiction pour organiser les connaissances dans la base de règles. Nous nous trouvions en effet en présence de groupes d'objets qui n'avaient pas de correspondants dans l'autre base. Il n'était donc pas possible d'apprendre des conditions qu'auraient dû respecter les représentations de la BDCarto. Nous avons dès lors suivi l'approche par classification directe qui impose l'intervention de l'expert pour classer interactivement les exemples d'apprentissage. Plusieurs sources d'informations ont été utilisées pour mener cette classification :

- Une carte topographique à l'échelle du 1:25.000 révisée en 1991.
- Une carte à l'échelle du 1:100.000 datant de 2001.
- Un carte à l'échelle du 1:50.000 datant de 2000.

La première carte est utile car elle permet de savoir si les éléments dans la BDTopo qui n'ont pas de correspondant dans la BDCarto existaient déjà en 1991. Les cartes au 1:25.000 découlent en effet de la BDTopo et, de ce fait, elles offrent une vision de la base plus ancienne que 1998 (date des données BDTopo utilisées).

La deuxième carte est utile pour le contrôle de la BDCarto. Les cartes au 1:100.000 (TOP100) sont issues de la BDCarto et celle à notre disposition est plus à jour que les données utilisées (saisie entre 1989 et 1993). Nous pouvons donc identifier les mises à jour éventuelles.

En comparant ces deux cartes, nous pouvons également comprendre comment les données ont été généralisées. Si un groupe de bâtiments existe dans la carte au 1:25.000 et qu'il n'apparaît pas dans la carte au 1:100.000, nous pouvons considérer que ce groupe a été volontairement éliminé et qu'il s'agit donc d'une équivalence. Pour vérifier que certaines données ont bien été mises à jour, nous pouvons exploiter la carte au 1:50.000 en complément des deux autres. Elle date cette fois de l'année 2000 et contient les mêmes informations que les cartes au 1:25.000 (la carte au 1:50.000 étant une réduction généralisée de deux cartes au 1:25.000). Elle nous permet donc de contrôler les hypothèses émises au sujet de l'origine des différences de représentation.

La classe des exemples d'apprentissage peut prendre trois valeurs : équivalence (les absences se justifient en raison des différences de résolution des bases), incohérence (il s'agit ici essentiellement d'erreurs d'appariement) ou mise à jour. Nous nous sommes intéressés ici aux mises à jour car les données des bases n'ont pas la même actualité et les informations utilisées pour attribuer la classe des exemples nous

permettaient d'identifier les mises à jour. Nous donnons un extrait des exemples d'apprentissage utilisés dans le tableau ci-dessous.

***Tableau 8**. Extrait des exemples d'apprentissage (Nombre total d'exemples : 183)*

ID	Attributs								Classe
	Nb	Surface	Périmètre	Densité	Distance c_gravité	Distance +proche	Distance +loin	Compacité	
1	2	4501.166	333.149	0.035	52.400	8.585	78.819	0.648	*Équivalence*
2	1	1937.692	160.202	0.114	38.191	40.221	40.221	1.207	*Équivalence*
3	8	11303.30	608.507	0.119	5.332	6.447	55.992	0.488	*Incohérence*
4	1	1654.495	146.191	0.099	8.271	8.039	8.039	1.238	*Mise à jour*
5	2	4501.166	333.149	0.035	51.672	11.968	165.672	0.361	*Equivalence*

Les hypothèses permettant de relier les classes de différences et les mesures caractérisant les groupes ont été apprises à l'aide de deux algorithmes : C4.5. [Quinlan 1993] et Ripper [Cohen 1995]. Nous donnons quelques exemples de règles de décision définies par C4.5. ci-dessous :

R_1 Si Nombre_bâtiments <= 2
Et si Ic <= 1.22
Et si 38.67 < Distance_+_loin < 83.76
ALORS classe *Équivalence*

R_2 Si Densité_groupe <= 0.11
Et si Ic > 0.49
Et si Distance_+_proche < 17.8
ALORS classe *Mise à jour*

R_3 Si Ic > 1.23
Et si 10.67 < Distance_centroide_groupe < 26.4
ALORS classe *Incohérence*

Au total, nous avons appris 19 règles avec C4.5. Plusieurs expérimentations ont été menées avec les deux algorithmes. Nous avons d'abord entrepris un apprentissage direct sur l'ensemble des exemples (183 dont 67 équivalences, 21 mises à jour et 95 incohérences). Nous avons également réalisé un apprentissage en deux étapes en distinguant d'abord les incohérences des *autres* différences, et en apprenant ensuite des règles différenciant les équivalences et les mises à jour. Nous avons finalement testé le « *boosting* » sur les exemples [Cornuéjols et Miclet 2002]. Cette méthode d'apprentissage est destinée à améliorer la performance de l'hypothèse apprise. Le principe est simple : la technique de « *boosting* » fait produire à l'algorithme d'apprentissage plusieurs hypothèses à partir de différents sous-ensembles d'exemples d'apprentissage. Les hypothèses apprises sont ensuite combinées pour ne former qu'un seul modèle plus performant. Le point central du « *boosting* » est de forcer l'apprenant à se concentrer sur les exemples difficiles à classer. Ceci se fait en augmentant à chaque itération le poids des exemples mal classés à l'étape précédente. Pour nos tests, nous avons fixé le nombre de classifieurs à 10. Les résultats obtenus pour les différentes expérimentations sont présentés dans le tableau 9.

Les taux d'erreur donnés ont été calculés par la méthode « *Leave One Out* ». Le principe est le même que celui de la validation croisée mais le nombre de passes est égal au nombre des exemples (on retire donc le premier exemple et on apprend avec les autres ; on retire ensuite le second exemple en remettant le premier et on apprend à nouveau ; on fait ceci pour chaque exemple).

***Tableau 9**. Résultats des tests d'apprentissage*

Algorithme d'apprentissage	Apprentissage direct	Apprentissage en deux étapes	Apprentissage direct avec *boosting* (10)
C4.5.	*Taux d'erreurs (LVO) :* ***41,4%***	***Incohérence et autres.*** *Taux d'erreurs (LVO) :* ***29,8%*** ***Équivalences et mises à jour.*** *Taux d'erreurs (LVO) :* ***28,4%***	*Taux d'erreurs (LVO) :* ***34,8%***
Ripper	*Taux d'erreurs (LVO) :* ***26,5%***	***Incohérence et autres.*** *Taux d'erreurs (LVO) :* ***28,4%*** ***Équivalences et mises à jour.*** *Taux d'erreurs (LVO) :* ***23,3%***	*Taux d'erreurs (LVO) :* ***26,14%***

On constate que les résultats d'apprentissage sont assez différents suivant l'algorithme qu'on utilise. Ainsi, C4.5. fournit un taux d'erreur d'environ 40% pour un apprentissage direct contre 26,5% avec Ripper. Un gain d'environ 10% est obtenu lorsqu'on réalise un apprentissage en deux étapes avec C4.5. Par contre, le résultat reste assez stable avec Ripper. L'hypothèse obtenue avec C4.5. n'est pas concluante en terme de performance pour un apprentissage directe. Ripper semble être mieux adapté pour cette expérimentation mais le taux d'erreur reste quand même relativement élevé.

Comment expliquer la différence de performance des classifieurs pour l'apprentissage directe ?

On peut considérer que l'apprentissage direct et l'apprentissage en deux étapes pour Ripper donnent globalement les mêmes résultats. En fait, lorsque Ripper effectue un apprentissage direct, il réalise implicitement un apprentissage en deux étapes. Ripper cherche en effet d'abord des règles qui séparent la classe la plus fréquente des autres classes. L'algorithme a donc d'abord séparé les incohérences des autres exemples. Ensuite, Ripper s'est focalisé sur le reste des exemples. Il a donc cherché à séparer les mises à jour des équivalences.

L'algorithme C4.5 ne procède pas de la même manière pour déterminer les règles de classification. Il se fonde sur une mesure de désordre pour séparer les exemples des différentes classes. De ce fait, pour l'apprentissage direct, C4.5. s'est davantage focalisé sur la séparation des équivalences et des mises à jour car la distinction entre ces deux classes est plus difficile à réaliser que la séparation avec les incohérences. En menant un apprentissage en deux étapes, l'algorithme est bien plus performant puisque les mises à jour et les équivalences sont cette fois mélangées (pour la première phase). On obtient d'ailleurs des résultats proches de ceux de Ripper.

Les taux d'erreur restent néanmoins assez élevés en réalisant un apprentissage en deux étapes. On peut émettre plusieurs hypothèses pour expliquer ces performances. D'abord, il est possible que les exemples soient trop peu nombreux en comparaison du nombre de descripteurs utilisés ou que ces derniers soient en trop grand nombre. Ensuite, on peut supposer que les exemples ne sont pas suffisamment

bien caractérisés. Ils auraient une description trop pauvre et les mesures ne seraient pas assez pertinentes.

Nous pensons que la faible qualité des résultats est liée ici à deux raisons principales : l'existence de groupes mixtes, c'est-à-dire des exemples qui auraient pu être classés dans différentes catégories, et surtout le manque de mesures pertinentes qui auraient permis de mieux discriminer les exemples.

L'existence de groupes mixtes s'explique par le fait que dans certains cas, il est difficile de distinguer clairement les équivalences des incohérences (erreurs d'appariement). Il y a bien souvent une incertitude sur la classe du groupe et la classification globale est susceptible de varier légèrement d'un expert à l'autre. On pourrait envisager de régler ce problème en créant explicitement les classes mixtes. Ceci suppose néanmoins un nombre suffisant d'exemples dans chaque classe, ce qui n'est pas le cas ici. C'est pour cette raison que nous avons testé l'apprentissage en deux étapes [Dietterich et Bakiri 1995]. Il nous a permis de comprendre que la différenciation des équivalences et des mises à jour n'était pas évidente pour C4.5.

L'utilisation d'autres descripteurs plus pertinents semble aussi nécessaire. Les tests devraient être approfondis avant d'exploiter les connaissances apprises. Nous pensons qu'il serait intéressant d'intégrer une information relative à l'orientation des groupes par exemple. En effet, les groupes correspondant aux erreurs d'appariement sont généralement parallèles aux zones d'habitat de la BDCarto, contrairement aux équivalences (figure 132).

Enfin, la méthode d'appariement pourrait être améliorée afin de limiter la classe des exemples à deux valeurs (mise à jour et équivalence). Nous avons apparié les objets sur la base du seul critère d'intersection des bâtiments. Nous aurions pu également définir un critère de proximité. Cela aurait sans doute fortement réduit le nombre d'erreur d'appariement. Nous aurions dû toutefois définir arbitrairement un seuil sur la distance, ce qui n'est pas évident.

Pour mieux différencier les équivalences des mises à jour, nous pensons qu'il serait utile d'introduire des informations relatives au contexte dans lequel se situe le groupe. L'apparition de nouveaux bâtiments peut s'accompagner de l'apparition de nouvelles routes et ce type d'information pourrait être utilisé pour classer les exemples avec plus de certitude.

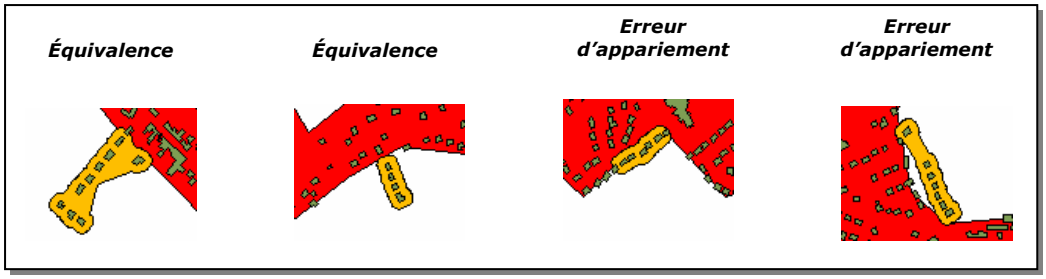


***Figure 132**. L'orientation du groupe : un descripteur susceptible d'améliorer la performance des résultats d'apprentissage.*

E.4.8 BILAN DE L'APPLICATION SUR LES BATIMENTS

Cette seconde application présente des différences avec l'expérimentation réalisée sur les ronds-points. Les particularités suivantes méritent d'être soulignées :

- Nous avons mis en œuvre le processus sur des données qui présentaient cette fois des niveaux d'abstraction très différents. Cette différence nous a contraint à transformer la représentation de la base la plus détaillée dans un niveau de détail équivalent à la représentation de la base la moins détaillée.
- Les jeux de données utilisés n'avaient pas la même actualité. Nous avons pu déceler des différences de représentation entre les bases liées à des mises à jour différées.
- Une seule méthode d'appariement a été calculée. Tous les couples d'objets ont de ce fait été retenus pour le contrôle inter-bases. Les couples d'objets mal appariés ont néanmoins été détectés lors de la phase d'apprentissage.
- L'apprentissage de règles s'est imposé. L'approche par classification directe a été adoptée. La mise en œuvre de l'apprentissage a montré que la définition de la forme des exemples n'était pas toujours immédiate.
- Nous avons eu recours à des techniques d'analyse spatiale pour créer les exemples d'apprentissage mais aussi, pour transformer la représentation de la base la plus détaillée. Les outils de l'analyse spatiale se sont avérées être indispensables pour réaliser ces tests.
- Les résultats de l'apprentissage pourraient être améliorés. Les tests entrepris mériteraient d'être approfondis mais l'apprentissage a prouvé son utilité. Sans utiliser ces techniques, aucune règle pour le contrôle inter-bases n'aurait pu être définie.
- Pour mieux identifier les mises à jour, des connaissances relatives à l'environnement des objets devraient être exploitées.

E.5 APPRENTISSAGE DE CORRESPONDANCES ENTRE VALEURS D'ATTRIBUTS DE TRONÇONS DE ROUTE

E.5.1 MOTIVATIONS

Nous n'avons pas traité jusqu'ici la cohérence entre valeurs d'attributs de classes. Pourtant, le problème du maintien de la cohérence dans le cadre de l'intégration se pose tout autant pour les attributs que pour les représentations géométriques des objets. Cette expérimentation vise à montrer que la démarche d'acquisition de connaissances que nous proposons, fondée sur l'apprentissage automatique, peut également s'appliquer sur les attributs. L'apprentissage peut aider à découvrir des règles de correspondances entre les valeurs d'attributs de deux bases à intégrer.

E.5.2 ATTRIBUTS ETUDIES ET SPECIFICATIONS

Nous avons décidé de réaliser cette étude sur certains attributs des tronçons routiers de la BDCarto et Géoroute. Nous nous sommes ainsi concentrés sur les correspondances entre les attributs « Vocation de la liaison » de la BDCarto et « Classement physique » et « Classement fonctionnel » de Géoroute. Ces attributs nous ont semblé intéressants à étudier car la définition a priori des correspondances entre eux (sur la base des spécifications) est incertaine.

L'attribut « Vocation de la liaison » de la BDCarto matérialise une hiérarchisation du réseau routier non pas sur un critère administratif, mais sur l'importance des tronçons de route pour le trafic routier. Cet attribut peut prendre différentes valeurs [BDCarto 2001] :

- 'Autoroutier' : elle s'applique aux autoroutes ou routes à chaussées séparées et carrefours dénivelés si leur longueur est supérieure à 5 km.
- 'liaison principale' : cette valeur est attribuée pour refléter la densification du maillage routier défini par les tronçons de type autoroutier. Les liaisons principales ont notamment pour fonction d'assurer les liaisons à fort trafic entre agglomérations importantes. Elles permettent également de relier les agglomérations importantes au réseau autoroutier. Les liaisons principales offrent aussi une alternative aux autoroutes lorsque celles-ci sont payantes et proposent des itinéraires de contournement des agglomérations.
- 'liaison régionale' : elle s'applique aux des liaisons qui ont pour fonction (quand celle-ci n'est pas assurée par des itinéraires de vocation plus élevée) de relier les communes de moindre importance entre elles ; de relier des voies de vocation plus élevée ; de structurer la circulation en agglomération et de desservir entre autres les localités et sites touristiques importants.
- 'liaison locale' : cette valeur est attribuée par exclusion des autres valeurs de l'attribut.
- 'bretelle' : cette valeur correspond aux tronçons de route qui définissent la description détaillée des échangeurs, des carrefours aménagés ou des ronds-points d'une extension supérieure à 100m ;
- 'piste cyclable' ;

La classification des tronçons dans Géoroute est un peu différente. Deux attributs peuvent aider à définir les correspondances avec la BDCarto. D'abord, l'attribut « Classement physique ». Celui-ci accepte plusieurs valeurs [Géoroute 1999] :

- 'autoroute' : on attribue ce classement si le tronçon de route appartient effectivement à cette catégorie d'après le décret du Conseil d'État (classement officiel) ;
- 'quasi-autoroute' : il s'agit de routes importantes de type 'autoroute' mais qui ne sont pas classées officiellement dans cette catégorie ;
- 'bretelle' : cette valeur est donnée pour des tronçons qui permettent la communication entre routes dont l'une passe sous l'autre ;
- 'route à 2 chaussées' : aucune définition n'est indiquée pour ce type de route ;
- 'route à 1 chaussée' : aucune définition n'est indiquée pour ce type de route ;
- 'chemin' : il s'agit d'une voie circulable, empierrée ou non ;
- 'Escalier ou passerelle' : il s'agit d'un escalier ou d'une passerelle directement relié au réseau routier. La passerelle doit supporter une allée.

Le second attribut de Géoroute à prendre en compte est l'attribut « Classement fonctionnel ». Les valeurs permises sont les suivantes [Géoroute 1999] :

- 'principal' : cette valeur est attribuée pour les liaisons inter-métropoles, en général des autoroutes et parfois des routes nationales.

- 'primaire' : cette valeur concerne les liaisons entre départements (grandes routes nationales par exemple mais aussi les quais de Seine…)
- 'secondaire' : elle concerne les liaisons entre villes à l'intérieur d'un département. Il s'agit de routes départementales.
- 'tertiaire' : cette valeur est affectée aux voies intra-villes permettant de se déplacer rapidement à l'intérieur des villes.
- 'desserte' : par exclusion, il s'agit de toutes les autres voies qui ne sont pas classées à un niveau supérieur.

Les spécifications des bases montrent qu'il existe un *conflit de description n-aires* entre les attributs [Devogele 1997]. L'information contenue par un ou plusieurs attributs de la BDCarto correspond en effet à l'information apportée par plusieurs attributs de Géoroute et il n'est pas évident de définir précisément les équivalences entre les valeurs d'attributs. Les relations entre les valeurs ne sont pas de type '1-n' ou 'n-1' mais plutôt de type 'n-m'. De nombreux attributs sont portés par les tronçons de route et on peut supposer qu'ils sont corrélés. Les attributs « Nombre de chaussées » et « Nombre de voies » de la BDCarto ne sont pas complètement indépendants par exemple. Ceux-ci peuvent en outre faire le lien avec les valeurs 'route à 1 chaussée' et 'route à 2 chaussées' de l'attribut « Classement physique » de Géoroute. Dans cette expérimentation, nous nous sommes limités à apprendre des correspondances de type '1-n' ou 'n-1'. Les algorithmes d'apprentissage que nous utilisons ne nous permettent pas de définir des classes composées de plusieurs valeurs faisant référence à plusieurs attributs. Mais cette limite n'est pas bloquante. Nous avons quand même pu apprendre des relations intéressantes entre les attributs.

E.5.3 APPARIEMENT DES TRONÇONS

L'étape incontournable avant de comparer les représentations des deux bases est l'appariement (cf. *MECO*). Les tronçons de route ont d'abord été appariés avant de mettre en œuvre l'apprentissage.

La méthode d'appariement utilisée est celle définie par [Devogele 1997] (voir annexe 3). C'est elle qui a servi à contrôler les correspondances calculées entre les ronds-points (cf. E.3.6.). La zone étudiée ici est d'ailleurs celle utilisée pour l'application sur les ronds-points. Au total, 6991 couples d'objets ont été définis.

E.5.4 APPRENTISSAGE DES CORRESPONDANCES ENTRE ATTRIBUTS

Tous les couples obtenus ont joué le rôle d'exemple d'apprentissage pour nos tests. Nous avons donc utilisé une grande quantité d'exemples. Le nombre d'exemples est un paramètre important dans l'apprentissage. Il peut fortement influencer la qualité de l'hypothèse apprise. Tous les couples d'objets appariés ont pu être pris en compte car nous avons adopté l'approche par prédiction. Nous n'avons donc pas attribué la classe des exemples manuellement.

APPRENTISSAGE DES CONDITIONS A RESPECTER PAR L'ATTRIBUT « CLASSEMENT PHYSIQUE » DE GEOROUTE

Nous avons d'abord cherché à apprendre les conditions que doivent respecter les valeurs de l'attribut « Classement physique » de Géoroute à partir des attributs « Vocation de la liaison », « Nombre de chaussées » et « Nombre de voies » de la

BDCarto. Nous avons ainsi appliqué l'algorithme d'induction C4.5. et obtenu l'arbre de décision reporté en figure 133. Nous présentons d'abord ci-dessous un extrait des exemples utilisés (tableau 10).

***Tableau 10**. Extrait des exemples d'apprentissage (Nombre total d'exemples : 6991)*

	Attributs			Classe
	Vocation liaison (tronçon BDCarto)	**Nombre de chaussées (tronçon BDCarto)**	**Nombre de voies (tronçon BDCarto)**	**Classement physique (tronçon Géoroute)**
1	bretelle	1	Sans objet	*Bretelle*
2	Liaison locale	1	1 voie ou 2 voies étroites	*Route à 1 chaussée*
3	Liaison principale	1	1 voie ou 2 voies étroites	*Route à 1 chaussée*
4	Liaison principale	1	2 voies larges	*Route à 1 chaussée*
5	Liaison principale	1	1 voie ou 2 voies étroites	*Route à 2 chaussées*
6	Liaison principale	1	3 voies	*Route à 1 chaussée*
7	Liaison principale	1	4 voies	*Route à 2 chaussées*
8	Liaison régionale	1	1 voie ou 2 voies étroites	*Route à 1 chaussée*
9	Liaison régionale	1	1 voie ou 2 voies étroites	*Route à 1 chaussée*
10	Liaison régionale	1	2 voies larges	*Route à 1 chaussée*
11	Autoroutier	1	Sans objet	*Autoroute*
12	Autoroutier	1	Sans objet	*Autoroute*
13	Autoroutier	1	Sans objet	*Quasi-autoroute*
14	Liaison régionale	1	2 voies larges	*Route à 1 chaussée*
15	Liaison principale	1	2 voies larges	*Route à 1 chaussée*
16	Liaison principale	1	4 voies	*Route à 2 chaussées*
17	Liaison locale	1	1 voie ou 2 voies étroites	*Route à 1 chaussée*
18	Liaison locale	1	1 voie ou 2 voies étroites	*Route à 1 chaussée*
18	Liaison locale	1	1 voie ou 2 voies étroites	*Route à 1 chaussée*
20	bretelle	1	Sans objet	*Route à 1 chaussée*

Cette hypothèse est globalement cohérente avec ce qu'on aurait pu annoncer en analysant les spécifications. Le taux d'erreur réelle calculé par validation croisée (avec k=10) est de 2%, ce qui est faible.

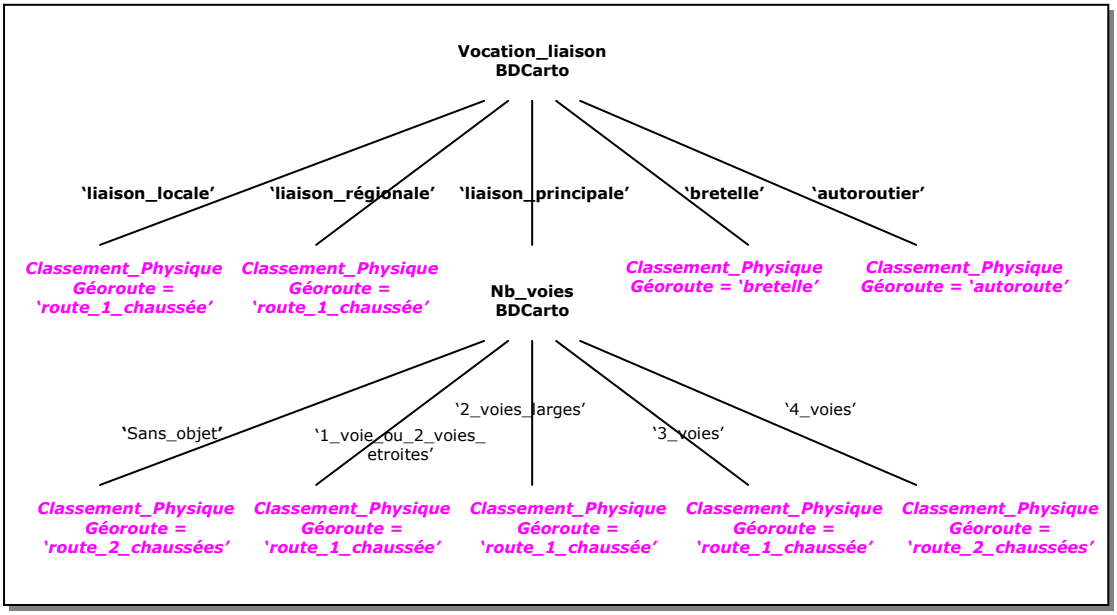


***Figure 133**. Arbre de décision appris déterminant les conditions que doivent respecter les valeurs de l'attribut « classement physique » de Géoroute à partir des attributs de la BDCarto.*

A la suite de cet apprentissage, nous avons quand même voulu examiner dans quels cas les règles n'étaient pas vérifiées dans les données de manière à savoir si l'hypothèse apprise devait être révisée. Un faible taux d'erreur réelle n'exclut pas en effet une révision même partielle des règles.

Nous nous sommes d'abord concentrés sur les autoroutes. La règle apprise par C4.5. introduit 8 erreurs pour les autoroutes sur un total de 87 objets de cette nature

dans la BDCarto. Une erreur apparaît si les objets correspondants dans Géoroute n'ont pas de classement physique de type 'autoroute' (règle non vérifiée). Parmi les 8 erreurs, 6 objets sont en effet étiquetés en 'quasi-autoroute', les 2 autres étant des erreurs d'appariement. Nous pensons qu'il ne s'agit pas d'incohérences pour les 'quasi-autoroutes' mais plutôt d'équivalences car les définitions concernant les autoroutes dans les spécifications des bases ne sont pas tout à fait les mêmes. Puisque les correspondances avec les 'quasi-autoroutes' sont très marginales, l'algorithme d'apprentissage n'a pas tenu compte de ces exemples pour déterminer son hypothèse. Néanmoins, dans ce cas précis, nous devons tenir compte de ces équivalences possibles. Il s'agit de cas relativement exceptionnels mais leur présence implique de modifier la règle apprise. Les conditions à respecter par l'attribut de Géoroute sont donc les suivantes :

Règle : **Si** Vocation_Liaison BDCarto = ('autoroutier')
Alors Classement_Physique Géoroute ∈ {'autoroute','quasi-autoroute'}

En ce qui concerne les bretelles, 33% des exemples ne vérifient pas la règle apprise (20/60). Parmi eux, 5 erreurs d'appariement ont été détectées contre 15 exemples classés en 'route à 1 chaussée'. Là aussi la règle doit être révisée. Dans la BDCarto, en présence de ronds-points détaillés, la valeur 'bretelle' est attribuée aux tronçons de route. Dans Géoroute, ce n'est pas le cas. De ce fait, des différences entre valeurs d'attributs sont possibles et doivent être considérées comme équivalences (les 15 tronçons erronés appartiennent à des ronds-points). La règle révisée est la suivante :

Règle : **Si** Vocation_Liaison BDCarto = ('bretelle')
Alors Classement_Physique Géoroute ∈ {'bretelle','route à 1 chaussée'}

Dans le cas d'une 'liaison locale' dans la BDCarto, nous avons appris que le classement physique de Géoroute devait avoir pour valeur 'route à 1 chaussée'. Seuls 32 cas ne vérifient pas cette règle sur 5017 couples de ce type. Nous pensons cette fois qu'il s'agit d'incohérences. Nous avons donc décidé de ne pas réviser la règle. Les cas erronés ont pour valeur 'route à 2 chaussées'. Si des routes de ce type existaient effectivement sur le terrain, l'attribut « Nombre de chaussées » de la BDCarto devrait également prendre la valeur '2 chaussées'. Or, ce n'est pas le cas. Il est donc probable qu'il s'agisse d'incohérences.

La règle concernant les 'liaisons régionales' n'a pas non plus été révisée. Sur 991 couples, 24 cas ne satisfont pas la règle et il s'agit d'incohérences.

Au sujet des liaisons principales, les règles apprises sont globalement satisfaisantes. Si le nombre de voies de la BDCarto a pour valeur 'sans objet', la règle indique que le classement physique de Géoroute doit être égal à 'route à 2 chaussées'. Cette règle est exacte. D'après les spécifications, la valeur 'sans objet' dans la BDCarto est définie obligatoirement pour les voies à deux chaussées et les bretelles d'échangeur. Il existe 18 couples qui ne vérifient pas cette règle sur un total de 82 paires. Ces couples prennent la valeur 'route à 1 chaussée'. Ils correspondent à des incohérences (erreurs dans la BDCarto ou Géoroute).

Les autres règles, celles pour les valeurs '1 ou 2 voies étroites', '2 voies larges' et '3 voies' sont également valables. Les exemples qui ne les satisfont pas correspondent bien à des incohérences (12 erreurs sur 334 dans le premier cas, 7 sur 301 dans le

deuxième et 1 sur 68 dans le troisième). Par contre, lorsque le nombre de voies est fixé à 4 dans la BDCarto, les valeurs possibles de l'attribut « classement physique » de Géoroute sont 'route à 2 chaussées' ou 'route à 1 chaussée'. La règle apprise doit donc être complétée :

Règle : **Si** Vocation_Liaison BDCarto = ('liaison principale') ∧ Nombre_Voies BDCarto = ('4 voies')
Alors Classement_Physique Géoroute ∈ {'route à 2 chaussées','route à 1 chaussée'}

APPRENTISSAGE DES CONDITIONS A RESPECTER PAR L'ATTRIBUT « VOCATION DE LA LIAISON » DE LA BDCARTO

Après ce test, nous avons mené l'apprentissage dans le sens inverse, autrement dit, nous avons cherché à apprendre des règles permettant de spécifier les valeurs à respecter par l'attribut « vocation de la liaison » de la BDCarto à partir des attributs « classement physique » et « classement fonctionnel » de Géoroute.

L'arbre de décision appris a été transformé sous forme de règles de production. Elles sont listées ci-dessous. Le taux d'erreur réelle estimé pour cette hypothèse est de 11%.

R_1. **Si** Classement_Fonctionnel Géoroute = ('primaire') ∧ Classement_Physique Géoroute = ('bretelle')
Alors Vocation_Liaison BDCarto = ('bretelle')

R_2. **Si** Classement_Fonctionnel Géoroute = ('primaire') ∧ Classement_Physique Géoroute = ('route à 1 chaussée')
Alors Vocation_Liaison BDCarto = ('liaison principale')

R_3. **Si** Classement_Fonctionnel Géoroute = ('primaire') ∧ Classement_Physique Géoroute = ('route à 2 chaussées')
Alors Vocation_Liaison BDCarto = ('liaison principale')

R_4. **Si** Classement_Fonctionnel Géoroute = ('primaire') ∧ Classement_Physique Géoroute = ('autoroute')
Alors Vocation_Liaison BDCarto = ('autoroutier')

R_5. **Si** Classement_Fonctionnel Géoroute = ('secondaire') ∧
Classement_Physique Géoroute = ('bretelle')
Alors Vocation_Liaison BDCarto = ('bretelle')

R_6. **Si** Classement_Fonctionnel Géoroute = ('secondaire') ∧
Classement_Physique Géoroute = ('route à 1 chaussée')
Alors Vocation_Liaison BDCarto = ('liaison régionale')

R_7. **Si** Classement_Fonctionnel Géoroute = ('secondaire') ∧
Classement_Physique Géoroute = ('route à 2 chaussées')
Alors Vocation_Liaison BDCarto = ('liaison principale')

R_8. **Si** Classement_Fonctionnel Géoroute = ('tertiaire') ∧
Classement_Physique Géoroute = ('bretelle')
Alors Vocation_Liaison BDCarto = ('bretelle')

R_9. **Si** Classement_Fonctionnel Géoroute = ('tertiaire') ∧
Classement_Physique Géoroute = ('route à 1 chaussée')
Alors Vocation_Liaison BDCarto = ('liaison locale')

R_{10}. **Si** Classement_Fonctionnel Géoroute = ('tertiaire') ∧
Classement_Physique Géoroute = ('route à 2 chaussées')
Alors Vocation_Liaison BDCarto = ('liaison locale')

R_{11}. **Si** Classement_Fonctionnel Géoroute = ('desserte')
Alors Vocation_Liaison BDCarto = ('liaison locale')

R_{12}. **Si** Classement_Fonctionnel Géoroute = ('principal')
Alors Vocation_Liaison BDCarto = ('autoroutier')

R_{13}. **Si** Classement_Fonctionnel Géoroute = ('sans objet')
Alors Vocation_Liaison BDCarto = ('liaison locale')

Nous avons révisé deux règles : les règles 6 et 7. La première est trop restrictive (579 cas ne la vérifient pas sur 1452 couples de ce type). Lorsqu'une 'route secondaire' de Géoroute est une 'route à 1 chaussée', il est possible d'avoir dans la BDCarto une route de type 'liaison régionale' mais aussi, 'liaison principale', 'liaison locale' et même 'bretelle'. Pour la règle 7, la plage de valeurs possibles dans la BDCarto doit également être étendue. Il est possible d'avoir une 'liaison principale' mais également une 'liaison régionale' ou une 'liaison locale'. La règle doit donc également être révisée.

E.5.5 BILAN DE L'APPLICATION SUR LES ATTRIBUTS

Pour cette expérimentation, nous n'avons pas présenté toutes les étapes des deux méthodes *MECO* et *MACO*. Néanmoins, l'étape d'appariement a été réalisée pour construire les exemples d'apprentissage. Nous nous sommes ensuite limités à étudier les possibilités d'utilisation de l'apprentissage supervisé pour la découverte de règles d'évaluation de la cohérence entre valeurs d'attributs.

Au regard de ces tests, nous considérons que l'apprentissage facilite la définition des correspondances mais nous soulignons à nouveau l'importance d'effectuer une analyse des hypothèses apprises. Les techniques d'induction utilisées n'ont pas permis d'extraire des correspondances peu fréquentes. Celles-ci ont été assimilées à des exemples bruités (non significatifs pour l'apprentissage) alors que la plupart d'entre elles étaient exactes. Plusieurs règles ont donc été révisées malgré de faibles taux d'erreur réelle estimés. Les règles apprises étaient cohérentes mais incomplètes.

Pour certaines correspondances (conflit de description n-aire), des techniques d'apprentissage acceptant des classes composées de plusieurs valeurs faisant référence à plusieurs attributs seraient mieux adaptées. De nombreux attributs sont en effet corrélés et il est parfois souhaitable d'apprendre des relations de type ensemble à ensemble entre les valeurs.

E.6 BILAN GENERAL

Les trois applications développées ont permis d'étudier la faisabilité de notre approche pour évaluer la cohérence inter-représentations.

La première expérimentation a montré l'intérêt de toutes les étapes de la méthode *MECO*. Les spécifications ont prouvé leur utilité. Leur analyse a suffit à recueillir toutes les connaissances permettant la réalisation des différentes étapes.

L'apprentissage a également été appliqué (étape de *MACO*). Il a permis de mettre en évidence l'écart existant entre les spécifications fournies par les producteurs de données et celles constatées dans les données.

La seconde expérimentation a montré l'intérêt d'avoir recours à des outils d'analyse spatiale pour enrichir les données et pour créer des exemples d'apprentissage pertinents. L'apprentissage s'est imposé pour cette application. Les spécifications ne nous ont pas permis de définir des règles permettant le contrôle inter-bases.

La dernière expérimentation a illustré la mise en œuvre de l'apprentissage automatique pour la découverte de correspondances entre valeurs d'attributs de routes de deux BDG. Plusieurs milliers d'exemples d'apprentissage ont été construits automatiquement en adoptant l'approche par prédiction. Au vu des résultats obtenus, on peut considérer que notre méthodologie convient à la fois pour l'étude de la cohérence entre représentations d'entités géographiques et pour l'étude de la cohérence entre valeurs d'attributs d'entités géographiques.

Tous ces tests ont été développés dans le prototype HéTéROGENE réalisé à cette fin. Avec ce prototype, il devrait être possible d'étudier la cohérence entre tous les ronds-points existants dans les bases BDCarto et Géoroute sur la France entière (plusieurs dizaines de milliers d'objets). La bibliothèque d'algorithmes que nous avons conçue pour nos expérimentations peut également être réutilisée pour d'autres applications.

CONCLUSION ET PERSPECTIVES

« Lorsqu'on s'achemine vers un but, l'important est de soigneusement observer sa route, car elle nous enseigne infailliblement la manière de l'atteindre. En plus, à chaque pas, le chemin nous montre ses richesses. »

P. Coelho

1. CONCLUSION

1.1 RAPPEL DE L'OBJECTIF

Le problème de recherche étudié dans cette thèse avait pour cadre l'intégration des bases de données spatiales et en particulier, l'intégration au niveau des données. Le sujet traité a porté sur l'évaluation de la cohérence inter-représentations. Notre objectif était de proposer une méthodologie permettant d'analyser, le plus automatiquement possible, chaque différence de représentation entre des données appariées issues de plusieurs bases. Cet objectif était motivé par le besoin de garantir la cohérence lors de l'intégration de données. Cette évaluation permettra d'améliorer la qualité des bases.

1.2 CONTRIBUTIONS

DEFINITION DES NOTIONS D'EQUIVALENCE ET D'INCOHERENCE

L'approche que nous avons adoptée pour étudier la cohérence entre les données repose sur l'utilisation des *spécifications* des bases de données spatiales. Les spécifications sont des métadonnées qui définissent les critères de saisie des objets des bases. Ils permettent de juger si les représentations des bases sont *équivalentes* ou *incohérentes*. Si les différences de représentation se justifient par les spécifications, nous considérons que ces différences sont normales et que les représentations sont équivalentes. Si les différences de représentation ne se justifient pas par les spécifications, en raison de la présence d'une erreur de saisie ou de différences d'actualité, nous considérons que les différences sont anormales et que les représentations sont *incohérentes*.

METHODOLOGIE D'EVALUATION DE LA COHERENCE INTER-REPRESENTATIONS

Nous proposons deux méthodes complémentaires qui définissent la méthodologie générale d'évaluation de la cohérence : la méthode *MECO* (Méthode d'Évaluation de la COhérence) et la méthode *MACO* (Méthode d'Acquisition de connaissances pour l'évaluation de la COhérence). A l'interface de ces deux méthodes se trouvent des connaissances. Celles-ci sont utilisées par *MECO* et produites par *MACO*.

- **Méthode *MECO***

La méthode *MECO* est destinée à réaliser l'évaluation de la cohérence. A partir des données des bases indépendantes, l'application de *MECO* doit permettre d'obtenir un ensemble de couples d'objets appariés qualifiés comme incohérent ou équivalent. Les étapes de la démarche de *MECO* sont les suivantes : l'enrichissement, le contrôle intra-base, l'appariement, le contrôle inter-bases, l'évaluation globale.

L'enrichissement vise à préparer les bases et les rapprocher pour le contrôle de la cohérence des représentations. Il touche à la fois les schémas et les données. Cette étape se caractérise par l'extraction d'information implicite et la caractérisation des objets. Elle fait appel à des outils d'analyse spatiale qui sont définis après une analyse individuelle et croisée des spécifications (étape de la méthode *MACO*).

Le contrôle intra-base est destiné à identifier un certain nombre d'erreurs avant la mise en correspondance des données. Il est réalisé sur chaque base indépendante et correspond à un contrôle d'intégrité. Ce contrôle s'appuie sur les caractéristiques des objets en particulier issues de la phase d'enrichissement précédente. Il est réalisé automatiquement, au moyen d'un système-expert. Les connaissances introduites dans le système-expert découlent des spécifications des bases. Ces connaissances sont acquises après l'étape d'analyse des spécifications (étape de la méthode *MACO*). Elles sont décrites dans un langage à base de règles pour les rendre utilisables par le système-expert.

L'appariement relie les objets homologues des différentes bases. La stratégie d'appariement retenue est de définir les liens entre les objets sans se soucier de la cohérence de leur représentation. L'usage de connaissances spécifiques sur les bases est ainsi réduit. Les objets homologues sont principalement identifiés en s'appuyant sur la géométrie et la position des objets. Cette stratégie permet l'emploi d'une boîte à outils d'appariement générique (outils d'analyse spatiale). Dans la mesure du possible, les résultats d'appariement doivent être analysés pour distinguer les couples bien appariés des erreurs potentielles d'appariement.

Le contrôle inter-bases permet de classer les représentations de chaque couple d'objets appariés comme des *incohérences* ou des *équivalences*. Il vérifie la cohérence inter-représentations. Les règles d'évaluation utilisées sont celles déduites des spécifications ou induites des données par apprentissage automatique (étapes de la méthode *MACO*). Ces règles sont décrites dans le langage à base de règles du système-expert.

Au terme du contrôle inter-bases, une évaluation globale des résultats est fournie. Ces résultats peuvent être exploités pour corriger les données des bases ou signaler à l'utilisateur quelles sont les représentations incohérentes, pourquoi elles sont incohérentes et où elles se trouvent.

- **Méthode *MACO***

La méthode *MACO* est destinée à produire l'ensemble des connaissances utiles à *MECO*. Les sources de connaissances qu'elle exploite sont les spécifications, décrites dans des documents papier, et les données. Les étapes de la démarche de *MACO* sont : l'analyse des spécifications (systématique) et l'acquisition de connaissances par apprentissage automatique supervisé (optionnelle).

L'analyse des spécifications est la première étape à mettre en œuvre. C'est elle qui permet de comprendre le contenu des bases et d'identifier les règles de saisie des objets. Une analyse individuelle et croisée des spécifications doit être réalisée. C'est une étape qui aujourd'hui est interactive et à la charge d'un expert du domaine. A son terme, un ensemble de connaissances doit être produit : la définition des concepts et des propriétés à extraire pendant l'enrichissement (par exemple, les ronds-points, le diamètre), la spécification des outils à utiliser pour enrichir les données des bases et les apparier (étapes de *MECO*) et une liste de règles reformulées dans le langage du système-expert choisi pour réaliser les contrôles intra-base et inter-bases (étapes de *MECO*). Les règles pour le contrôle inter-bases sont définies manuellement par l'expert si les spécifications le permettent, et si l'expert souhaite tenir compte des spécifications fournies par le producteur de données pour l'évaluation. Dans le cas contraire, l'expert applique l'étape d'apprentissage automatique supervisé (seconde étape de *MACO*, cf. ci-dessous).

L'analyse des spécifications peut être facilitée par la description normalisée du contenu des documents. Cette normalisation s'est traduite dans cette thèse par la définition d'un modèle de représentation des spécifications (fruit d'un travail réalisé en commun avec Nils Gesbert notamment). La formalisation des spécifications permet de comparer plus facilement les documents. La formalisation aide aussi à mettre en évidence les contraintes imprécises qui requièrent un examen des données pour les expliciter. Plus généralement, le modèle des spécifications est un bon outil pour analyser les spécifications de chaque base, même en dehors du cadre de l'intégration.

La seconde étape de *MACO* est l'apprentissage automatique à partir de données. Elle sert à recueillir des règles pour le contrôle inter-bases (étape de *MECO*). L'apprentissage est appliqué si les spécifications papier sont insuffisantes pour réaliser le contrôle ou si l'on souhaite réaliser l'évaluation de la cohérence en utilisant les spécifications constatées dans les données. Deux approches sont proposées pour décrire la forme des règles d'évaluation. Du point de vue de l'apprentissage, la mise en pratique de ces approches présente des différences importantes.

La première approche est ce que nous avons appelé l'approche par *classification directe*. Elle consiste a définir des règles permettant de classer directement les représentations de chaque couple d'objets appariés comme des incohérences ou des équivalences. Lorsque ces règles sont acquises par apprentissage, l'intervention de l'expert est requise pour classer les exemples (un exemple correspond à un couple d'objets appariés classé comme incohérent ou équivalent). Les connaissances apprises reflètent alors les connaissances de l'expert. C'est une approche qui est bien adaptée à la description des règles d'évaluation mais elle est fastidieuse à mettre en œuvre. La construction des exemples n'est pas automatique.

La seconde approche proposée vise à réduire l'intervention de l'expert et acquérir des connaissances qui sont plus proches des données : c'est l'approche par *prédiction, comparaison et classification.* Le principe est d'apprendre d'abord les conditions que doit respecter la représentation des objets d'une base à partir de la représentation des objets de l'autre base. Les exemples d'apprentissage (un exemple correspond à la représentation caractérisée de la première base avec pour classe la représentation homologue de la seconde base) peuvent être créés automatiquement, à l'issue de l'étape d'appariement. Les connaissances apprises reflètent les connaissances contenues dans les données, c'est-à-dire celles des personnes ayant produit les objets des bases. Cette approche est plus facile à mettre en œuvre puisque l'intervention de l'expert n'est pas nécessaire pour classer les exemples d'apprentissage. En contrepartie, le contrôle des règles apprises est impératif car l'apprentissage est réalisé avec des données bruitées. Les incohérences sont en effet considérées comme du bruit pour l'apprentissage. Il faut donc s'assurer que l'algorithme d'apprentissage n'apprend pas le bruit. L'intervention de l'expert est requise à ce niveau.

Notre méthodologie est synthétisée à la figure 134.

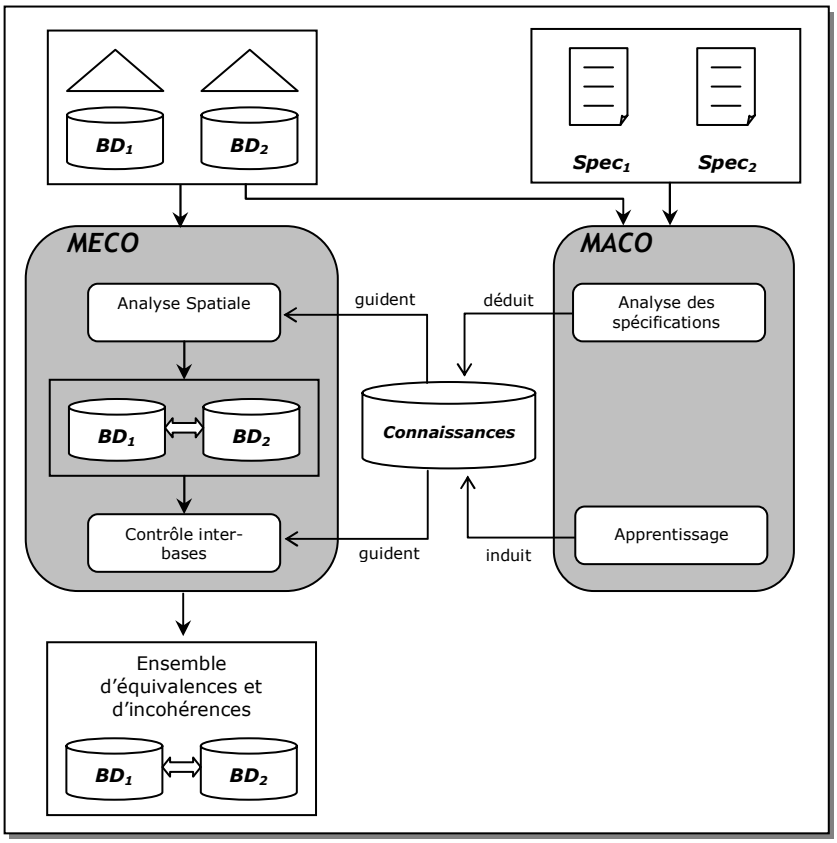


***Figure 134**. Synthèse de la méthodologie d'évaluation de la cohérence proposée.*

APPLICATION DE LA METHODOLOGIE

Plusieurs applications ont été développées pour évaluer la méthodologie. Un prototype a été mis au point à cette fin : le prototype HéTéROGENE.

Nous avons d'abord montré l'intérêt des différentes étapes de la méthode *MECO* grâce à l'expérimentation réalisée sur les ronds-points. Les spécifications ont prouvé leur utilité. Leur analyse a suffi à recueillir la plupart des connaissances utiles à l'évaluation de la cohérence. Nous avons également mis en évidence l'intérêt d'utiliser l'apprentissage automatique supervisé. Ces techniques nous ont permis d'acquérir des règles reflétant la réalité de la saisie des données. Nous avons pu constater l'écart existant entre les spécifications fournies par les producteurs de données et celles suivies en pratique par les personnes chargées de la production des bases. Grâce à la découverte de ces connaissances, une évaluation plus tolérante de la cohérence a pu être réalisée.

La seconde application développée a porté sur l'étude des différences entre bâtiments. Nous avons étudié la cohérence des représentations de bâtiments présentant des différences de modélisation importantes. Nous avons souligné le rôle que devaient jouer les méthodes d'analyse spatiale pour enrichir les données et construire des exemples d'apprentissage pertinents. Nous avons également montré l'intérêt d'utiliser l'apprentissage automatique dans un contexte où les spécifications ne permettaient pas de définir des règles pour le contrôle inter-bases.

La dernière expérimentation réalisée a illustré l'application de la méthode *MACO* et en particulier, l'étape d'apprentissage automatique, pour la découverte de règles de correspondances entre valeurs d'attributs de routes. L'approche par prédiction fut

retenue pour définir la forme des exemples d'apprentissage. Plusieurs milliers d'exemples d'apprentissage ont ainsi pu être construits automatiquement. Les résultats obtenus nous permettent de considérer que la méthodologie convient aussi pour l'étude de la cohérence entre valeurs d'attributs d'entités géographiques.

Nous avons pu tirer plusieurs enseignements de ces tests. Le premier est que les spécifications constituent des métadonnées particulièrement utiles pour l'intégration. Toutefois, l'étape d'analyse des spécifications reste assez longue à réaliser aujourd'hui. Son automatisation est souhaitée. Ensuite, au vu des résultats obtenus, nous considérons que l'apprentissage automatique supervisé est une bonne solution pour acquérir des connaissances à partir des données. Cependant, il s'agit d'être prudent dans l'application des algorithmes d'apprentissage. Si en amont le problème d'apprentissage est mal posé, les hypothèses apprises risquent de ne pas être cohérentes avec la réalité et ne seront donc pas d'une grande utilité. En aval, il est impératif de vérifier les hypothèses apprises et, au besoin, de les réviser. Cette analyse est d'autant plus importante que dans notre contexte, suivant la forme des exemples d'apprentissage retenue, de nombreux exemples peuvent être bruités.

EXPLOITATION DES RESULTATS

Puisque notre méthodologie permet d'identifier des erreurs de saisie dans les bases, les résultats de l'évaluation pourraient servir à améliorer la qualité des données, tant du point de vue de la géométrie des objets que de leurs attributs.

En réalisant davantage de tests d'apprentissage, on pourrait également envisager d'enrichir les spécifications papier. On pourrait par exemple distinguer deux types de seuil pour les règles de saisie : les seuils « théoriques », et les seuils « constatés » dans les données. Cela permettrait également de mieux préciser certaines règles imprécises.

2. PISTES DE RECHERCHE

La méthodologie proposée pourrait être améliorée sur différents aspects touchant à la fois la méthode *MECO* et la méthode *MACO*. Nous mentionnons ci-dessous plusieurs perspectives de recherche.

2.1 PERSPECTIVES POUR LA METHODE *MACO*

RAFFINER LE MODELE DES SPECIFICATIONS

Le modèle que nous avons proposé pour formaliser la représentation des spécifications a été défini à partir de documents provenant uniquement des bases de l'IGN. Il serait utile de le confronter à davantage de documents provenant d'autres producteurs de données géographiques pour valider son contenu.

On peut déjà mentionner quelques améliorations possibles à apporter à ce modèle. En premier lieu, comme le propose [Gesbert et al. 2004], il est nécessaire de mieux distinguer les règles qui portent sur les entités géographiques du monde réel de celles qui concernent les objets de la base. En définissant explicitement une classe « Entité géographique » qui représente un concept du monde réel, il est possible de créer une ontologie du domaine qui peut être particulièrement utile pour l'intégration.

Dans notre contexte, cette distinction permettrait de mettre plus facilement en évidence toutes les contraintes d'intégrité que doivent respecter les objets de la base. Nous pourrions aussi plus facilement traduire les spécifications qui font référence aux entités du monde réel, dans l'univers des données.

Plus simplement, sans remettre en cause la forme générale du modèle, celui-ci pourrait être raffiné. Notre modèle ne permet pas, par exemple, de définir une règle spécifiant la distance minimale à respecter entre deux points constitutifs d'une ligne (objet d'une classe de la base). Il est également difficile de traduire la règle indiquée à la figure 135. Cette règle est d'ailleurs représentée sous forme de figure car il est difficile de l'exprimer par des phrases. Au sujet des relations spatiales, il serait utile de définir une classification des relations entre objets lorsque ceux-ci sont disjoints. La classe « *Autre Contrainte de Relation Spatiale* » devrait être raffinée. Par exemple, on trouve dans les spécifications des relations du type : *être le long de*, *être à côté*, *être devant*, *être au milieu*, etc. Certaines de ces relations peuvent être traduites en relation métrique après une analyse des données, mais toutes ne peuvent pas l'être. Une classification permettrait de les spécialiser.

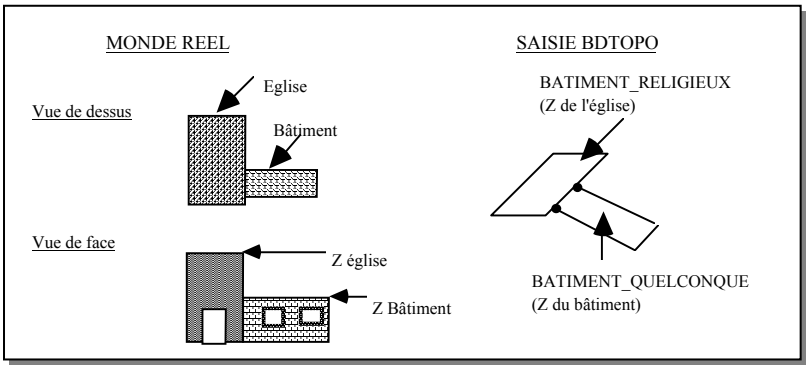


Figure 135. *Exemple de règle de modélisation difficilement traduisible dans le modèle des spécifications proposé (Source : [BDTopo 1994]).*

MANIPULER AUTOMATIQUEMENT LES SPECIFICATIONS FORMELLES

Le principal défaut de la méthode d'analyse des spécifications de *MACO* est qu'elle reste aujourd'hui complètement interactive. S'il est difficile d'imaginer que cette étape soit complètement automatisable, on peut envisager à l'avenir une traduction automatique des spécifications formalisées sous forme de règles de production pour réaliser les contrôles intra-base et inter-bases. Il serait utile de définir des outils capables de reformuler automatiquement les spécifications formalisées en contraintes d'intégrité (contrôle intra-base) et en règles de classification directe ou de prédiction (contrôle inter-bases).

On peut se demander par ailleurs si le langage orienté-objet adopté pour représenter les spécifications formalisées est bien adapté. Il serait intéressant à ce sujet de tester les logiques de description pour représenter les spécifications, à l'image des travaux réalisés sur la représentation et la manipulation des ontologies [Hakimpour 2003]. Ces logiques pourraient être utiles pour réaliser des inférences sur les classes « Entités géographiques » que nous mentionnions dans la section précédente. Mais, ces mécanismes d'inférence ne présenteraient sans doute pas un grand intérêt pour manipuler les contraintes de saisie proprement dites. En fait, plusieurs langages de représentation seraient peut-être nécessaires car les spécifications doivent aussi rester facilement compréhensibles par un utilisateur.

REPRESENTER GRAPHIQUEMENT LES DIFFERENCES DE SPECIFICATIONS

La représentation des spécifications dans un langage formalisé pourrait également permettre de développer des outils de comparaison automatique des spécifications. Ces outils seraient utiles pour déclarer les correspondances entre les schémas et identifier les différences de représentation possibles entre les données. Il s'agit d'une autre perspective de recherche.

Dans le même ordre d'idée, il serait utile d'étudier les possibilités de représenter graphiquement les différences de spécifications (figure 136). Un stage au laboratoire COGIT a déjà été réalisé dans ce sens [Goder 2003]. Le travail mériterait d'être poursuivi.

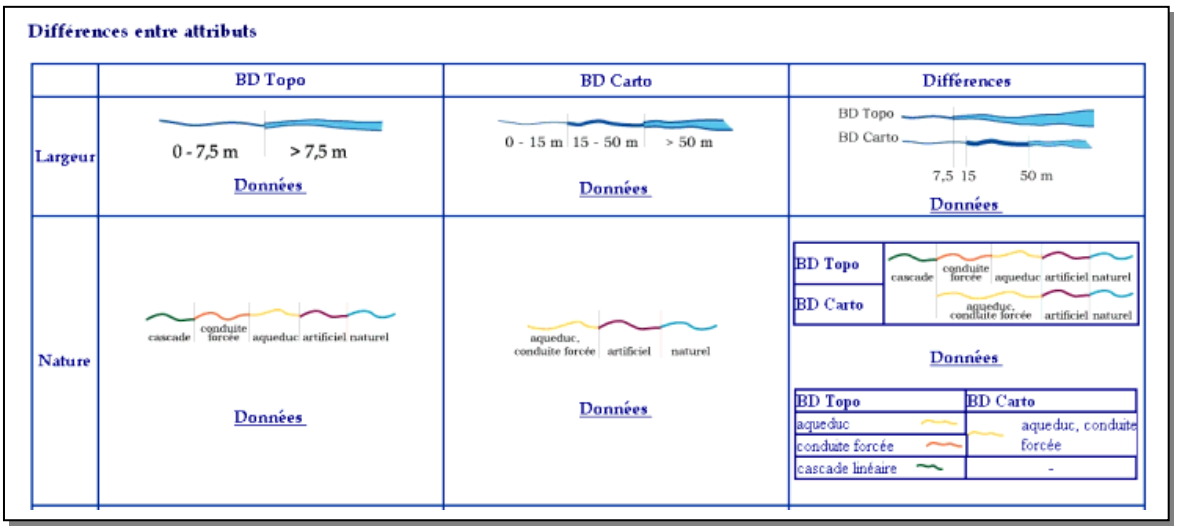


***Figure 136**. Représentation de différences de spécifications entre attributs de deux BDG (Source : [Goder 2003])*

ÉTENDRE *MACO* A D'AUTRES TECHNIQUES D'APPRENTISSAGE AUTOMATIQUE

Concernant l'étape d'apprentissage de *MACO*, nous nous sommes volontairement restreints à utiliser des algorithmes d'apprentissage automatique symbolique dont le langage de représentation des exemples était de type attribut/valeur. A la suite des différents tests, nous avons pu apprécier les limites des algorithmes. Il serait maintenant utile d'explorer d'autres techniques d'apprentissage automatique.

D'abord, des algorithmes acceptant des valeurs numériques dans la définition de la classe des exemples devraient être utilisés, en gardant à l'esprit que les hypothèses apprises doivent être compréhensibles par un expert du domaine. Ensuite, des techniques plus robustes au bruit devraient être testées, en particulier, lorsque l'approche par prédiction est adoptée. Il serait également intéressant d'étudier les méthodes d'apprentissage capables de prendre en compte l'avis de plusieurs experts [Richardson et Domingos 2003]. Dans notre cas, cette multiplicité serait particulièrement intéressante car les connaissances implicites utilisées lors de la saisie des données varient légèrement d'un expert à l'autre. En adoptant l'approche par classification directe, les règles apprises ne représentent donc l'avis que d'un seul expert, lequel n'est pas complètement représentatif de l'ensemble des opérateurs de saisie.

Dans le souci de faciliter la tâche de recueil des exemples pour l'approche par classification directe, on pourrait également étudier l'apprentissage actif. Ce type

d'apprentissage se caractérise par l'existence d'une interaction entre l'expert et l'apprenant. Ce dernier tente d'identifier la meilleure hypothèse en posant un minimum de questions à l'expert. Cet apprentissage pourrait permettre de définir une hypothèse en réduisant la taille de l'échantillon d'exemples. Ceci est intéressant car dans notre contexte, il est possible de devoir apprendre avec peu d'exemples pour traiter les cas particuliers peu fréquents.

Enfin, il serait profitable d'utiliser des algorithmes d'apprentissage capables de prendre en compte les caractéristiques spatiales des objets et, en particulier, les relations spatiales qu'ils entretiennent avec d'autres objets [Ester et al. 1997, Koperski et al. 1998]. L'information contextuelle des objets paraît pertinente pour distinguer les différences résultant d'erreurs de saisie de celles provenant de mises à jour.

2.2 PERSPECTIVES POUR LA METHODE *MECO*

Les améliorations qui mériteraient d'être effectuées touchent également la méthode *MECO*. Nous les présentons ci-dessous.

ÉVALUER LES RESULTATS DE L'APPARIEMENT

Dans l'approche que nous proposons, nous préconisons d'évaluer les résultats de l'appariement avant de mettre en œuvre le contrôle inter-bases. Il faut être capable de juger si le couple d'objets appariés est probablement juste ou pas. L'évaluation peut être réalisée en exploitant une autre méthode d'appariement, fondée sur d'autres critères. C'est une solution que nous avons adoptée pour l'application sur les ronds-points mais elle n'est pas optimale. Dans certains cas, il ne sera pas possible d'utiliser une seconde méthode d'appariement. Dès lors, il s'agira de définir d'autres solutions pour évaluer les résultats de l'appariement automatiquement. Cet aspect de la méthode *MECO* devrait être approfondi.

ÉTUDIER LA COHERENCE DES RELATIONS TOPOLOGIQUES

Dans cette thèse, nous nous sommes principalement intéressés aux différences de modélisation de la géométrie des objets. Nous avons également réalisé une application assez simple sur les attributs. Mais la cohérence touche également les relations topologiques entre les objets. A des échelles différentes, les relations entre deux objets peuvent différer, d'autant que le mode d'implantation des objets peut avoir changé. Une étude sur la cohérence des relations topologiques entre objets à différentes échelles devrait être réalisée. Le travail de [Paiva 1998] devrait être exploité à ce sujet.

ÉTUDIER LA COHERENCE DE POSITION

Nous avons également assez peu étudié la cohérence de position des objets. Nous avons préconisé de fixer un seuil de recherche des candidats assez grand lors de l'appariement afin de relier des objets homologues anormalement éloignés, mais les différences de position n'ont pas été étudiées. Il serait utile de compléter les applications dans ce sens.

REPRESENTER L'IMPRECISION DES SPECIFICATIONS A L'AIDE DE REGLES FLOUES

Grâce aux techniques d'apprentissage automatique, nous avons vu qu'il était possible d'identifier l'écart existant entre les seuils définis dans les spécifications fournies par les producteurs de données et celles constatées dans les données. Cette zone d'imprécision existe car les spécifications papier sont là pour guider les opérateurs de saisie mais les seuils ne sont pas rigoureusement respectés. Ils donnent une ordre de grandeur aux opérateurs.

Dans ce travail, nous avons adopté un mode d'évaluation binaire. Les représentations ont été jugées soit incohérentes, soit équivalentes. A l'avenir, il serait plus juste de juger les différences de représentation d'une manière plus souple. Des modificateurs linguistiques permettant de moduler la qualification des différences de représentation devraient être utilisés. On pourrait par exemple qualifier les différences comme totalement incohérentes, plutôt incohérentes, plutôt équivalentes ou totalement équivalentes. Pour une évaluation de ce type, la logique floue semble bien adaptée [Bouchon-Meunier 1999]. Les limites des fonctions d'appartenance qu'il est nécessaire de définir dans ce cadre pourraient être fixées en exploitant les seuils des spécifications papier et ceux obtenus par apprentissage automatique. Il existe également des algorithmes d'apprentissage qui permettent d'induire des règles floues [Guillaume 2001]. Ce type d'outil serait également intéressant à étudier. Ces règles pourraient être facilement introduites dans notre système-expert car JESS est doté d'une extension permettant de manipuler des règles floues (FuzzyJESS).

CARTOGRAPHIER LES INCOHERENCES ET LES EQUIVALENCES

En terme de présentation des résultats (pour l'étape d'évaluation globale de *MECO*), il serait utile de définir une sémiologie adaptée à la représentation de la cohérence. La solution pour obtenir une représentation efficace n'est cependant pas évidente. La définition de la règle de sémiologie à appliquer est triviale mais il n'est pas facile de rendre lisible l'information, les objets appariés pouvant se superposer. Ce problème se pose également pour la représentation des liens d'appariement. Des études pourraient être menées à ce sujet.

2.3 PERSPECTIVES POUR L'INTEGRATION DE BASES DE DONNEES SPATIALES

DEFINIR UN AGL DEDIE A L'INTEGRATION DE BASES DE DONNEES SPATIALES

Pour faciliter l'intégration des BD spatiales, il serait très utile de concevoir un AGL spécifiquement conçu à cette fin. Celui-ci devrait permettre de déclarer les correspondances entre les schémas graphiquement, et dans un langage comme celui proposé par exemple par [Spaccapietra et al. 1992]. Les schémas ne devraient pas être déconnectés des données afin de pouvoir mener une intégration conjointe des schémas et des données. Les spécifications des BD devraient être également associées. Les travaux réalisés dans le cadre des projets MurMur [MurMur 2002] et Amber [Sotnykova 2003] vont déjà dans ce sens. L'outil proposé par [Cockcroft 2004] pour définir et enregistrer les contraintes d'intégrité spatiales constitue également un bon point de départ.

D'une manière plus générale, il serait utile d'associer systématiquement les spécifications des bases de données géographiques aux schémas et aux données des bases. Les AGL devraient permettre de saisir les spécifications en même temps que la définition des schémas des BDG.

APPRENDRE LES CORRESPONDANCES ENTRE LES SCHEMAS A PARTIR DES CORRESPONDANCES ENTRE LES DONNEES

Du point de vue de l'intégration, plutôt que de définir d'abord les correspondances entre les schémas et puis les constater dans les données, on pourrait envisager d'exploiter les correspondances entre les données appariées pour déclarer automatiquement les correspondances au niveau des schémas. Les algorithmes d'apprentissage automatique pourraient être utilisés à cette fin. L'approche par prédiction que nous avons proposée prend déjà cette direction.

Dans le futur, on peut imaginer être capable d'intégrer des bases de données spatiales le plus automatiquement possible. Des outils de comparaison de spécifications pourraient aider à définir les relations entre les schémas qui seraient traduites automatiquement en assertions de correspondances inter-schémas (ACI). Les spécifications des bases pourraient également être reformulées automatiquement sous forme de règles d'évaluation de la cohérence. Après avoir apparié les données des bases au moyen d'algorithmes d'appariement géométrique, la cohérence serait automatiquement contrôlée. Les données des bases pourraient ensuite être corrigées et finalement intégrées en suivant les spécifications de la nouvelle base.

REFERENCES BIBLIOGRAPHIQUES

[Abbas 1994] Abbas I. 1994. *Base de données vectorielles et erreur cartographique : problèmes posés par le contrôle ponctuel. Une méthode alternative fondée sur la distance de Hausdorff : le contrôle linéaire*, Thèse de doctorat de l'université Paris 7.

[Abdelmoty et Jones 1997] Abdelmoty A.I. and Jones C.B. 1997. Towards Maintaining Consistency in Spatial Databases, In *Proceedings of the 6th International Conference on Information and Knowledge Management* (*CIKM'97*), pp 293-300.

[Agent 1999a] Agent 1999. *Selection of basic measures*, Report DC1 of the AGENT project, ESPRIT/LTR/24939.

[Agent 1999b] Agent 1999. *Specifications for measures on MESO level and organisations*, Report DC4 of the AGENT project, ESPRIT/LTR/24939.

[Ahmed et al. 1991] Ahmed R., Smedt P.D., Du W., Kent W., Ketabchi M.A. and Litwin W.A. 1991. The Pegasus Heterogeneous Multidatabase System, *IEEE Computer*, 24(12), pp. 19-27.

[Alchourron et al. 1985] Alchourron C., Gardenfors P. and Makinson D. 1985. On the logic of theory change : partial meet functions for contraction and revision, *Journal of Symbolic Logic*, 50(2), pp. 510-530.

[Anders et al. 1999] Anders K-H., Sester M. and Fritsch D. 1999. Analysis of settlement structures by graph-based clustering, In *Proceedings of the Semantic Modelling for the Acquisition of Topographic Information from Images and Maps Conference (SMATI'99)*, pp. 41-49.

[Anokhin et Motro 2001] Anokhin P. and Motro A. 2001. Data Integration: Inconsistency Detection and Resolution Based on Source Properties, In *Proceedings of the International Workshop on Foundations of Models for Information Integration (FMII'01)*.

[Appice et al. 2003] Appice A., Ceci M., Lanza A., Lisi F.A. and Malerba D. 2003. Discovery of spatial association rules in geo-referenced census data: A relational mining approach, *Intelligent Data Analysis*, 7(6), pp. 541-566.

[Arens et al. 1996] Arens Y., Knoblock C.A. and Shen W.-M. 1996. Query reformulation for dynamic information integration, *Journal of Intelligent Information Systems*, Special Issue on Intelligent Information Integration, 6(2-3), pp. 99-130.

[Aufaure et al. 2000] Aufaure M.A., Yeh L. et Zeitouni K. 2000. Fouille de Données Spatiales, In Prade H., Jeansoulin R. et Garbay C. (Eds.) : *Le temps, l'espace et l'évolutif en Sciences du Traitement de l'Information*. Toulouse : CEPADUES-éditions, pp. 319-328.

[Aussenac 1989] Aussenac N. 1989*. Conception d'une méthodologie et d'un outil d'acquisition de connaissances expertes*, Thèse de doctorat, Université Paul Sabatier de Toulouse.

[Ayel et Rousset 1990] Ayel M. et M.-C. Rousset 1990*. La cohérence dans les bases de connaissances*. Toulouse : CEPADUES-éditions, Collection Intelligence artificielle, 106 p.

[Azé et Kodratoff 2002] Azé J. and Kodratoff Y. 2002. A study of the effect of noisy data in rule extraction systemes, In *Proceedings of the 16th European Meeting on Cybernetics and Systems Research (EMCSR'02)*, vol. 2, pp. 781-786.

[Badard 2000] Badard T. 2000. *Propagation des mises à jour dans les bases de données géographiques multi-représentations par analyse des changements géographiques*, Thèse de doctorat en Sciences de l'Information Géographique, Université de Marne-la-Vallée, 114 p.

[Badard et al. 2001] Badard T., Braun A. et Raynal L. 2001. *Projet SGME, Synthèse de l'étude sur la conception d'un serveur géographique multi-échelles*, Rapport SGME/2500/017.

[Badard et Lemarié 2002] Badard T. et Lemarié C. 2002. Associer les données : l'appariement, In Ruas A. (Ed.), *op. cit.*, chapitre 9, pp. 163-183.

[Badard et Braun 2003] Badard T. and Braun A. 2003. OXYGENE : an open framework for the deployment of geographic web services, In *Proceedings of the International Cartographic Conference* (*ICC'2003*), pp. 994-1003.

[Balley et al. 2004] Balley S, Parent C. and Spaccapietra S. 2004. Modeling geographic data with multiple representations, *International Journal of Geographical Information Science*, 18(4), pp. 329-354.

[Bard 2000] Bard S. 2000. *Révision d'une base de connaissances, Application à la généralisation cartographique*, Rapport de stage du DESS « Méthodes Quantitatives en Gestion et Aménagement de l'Espace », Université de Metz, 62 p. + annexes.

[Bard 2004] Bard S. *Méthode d'évaluation de la qualité de données géographiques généralisées*, Application aux données urbaines, Thèse de doctorat en Informatique, Université Paris 6, 206 p.

[Barillot 2002] Barillot X. 2002. Mesures et structures d'analyse, In Ruas A. (Ed.), *op. cit.*, chapitre 10, pp. 187-201.

[Barillot et Plazanet 2002] Barillot X. et Plazanet C. 2002. Analyse des formes des routes, In Ruas A. (Ed.), *op. cit.*, chapitre 11, pp. 203-223.

[Batini et al. 1986] Batini C., Lenzerini M and Navathe S.B. 1986. A comparative analysis of methodologies for database schema integration, *ACM Computing Surveys*, 18(4), pp. 323-364.

[BDCarto 2001] Spécifications de contenu de la BDCARTO®, version 2.0., IGN, Saint-Mandé.

[BDPays 2001] Spécifications de contenu de la BDTopo Pays, version 1.1., IGN, Saint- Mandé.

[BDTopo 1994] Spécifications détaillées de la BDTOPO®, version 3.1., IGN, Saint- Mandé.

[Bédard 1999] Bédard Y. 1999. Visual Modeling of Spatial Databases Towards Spatial Extensions and UML, *Geomatica*, 53(2), pp. 169-186.

[Bédard et al. 2002] Bédard Y., Bernier E. et Devillers R. 2002. La métastructure vuel et la gestion des représentations multiples, In Ruas A. (Ed.), *op. cit.*, chapitre 8, pp. 149-162.

[Bel Hadj Ali 2001] Bel Hadj Ali A. 2001. *Qualité géométrique des entités surfaciques. Application à l'appariement et définition d'une typologie des écarts géométriques*, Thèse de doctorat en Sciences de l'Information Géographique, Université de Marne-la-Vallée, 210 p.

[Bellahsène et Baril 2001] Bellahsène Z. et Baril X. 2001. XML et les systèmes d'intégration de données, Interopérabilité et intégration des systèmes d'information, *Revue ISI : Ingénierie des Systèmes d'Information*, 6(3), pp. 11-32.

[Berlin et Motro 2002] Berlin J. and A. Motro 2002. Database Schema Matching Using Machine Learning with Feature Selection, In *Proceedings of the 14th International Conference on Advanced Information Systems Engineering (CAiSE'02)*, Lecture Notes in Computer Science 2348, pp. 452-466.

[Bertin 1973] Bertin J. 1973. *Sémiologie graphique. Les diagrammes, les réseaux, les cartes*. Paris – La Haye : Mouton et Gauthier-Villars, 400 p.

[Beynon-Davies et al. 1997] Beynon-Davies P., Bonde L., McPhee D. and Jones C.B. 1997. A Collaborative Schema Integration System, *Computer Supported Cooperative Work*, 6(1), pp. 1-18.

[Bishr 1997] Bishr Y. 1997. *Semantic Aspects of Interoperable GIS*, PhD Thesis, International Institute for Geo-Information Science and Earth Observation (ITC), 154 p.

[Boffet 2001] Boffet A. 2001. *Méthode de création d'informations multi-niveaux pour la généralisation cartographique de l'urbain,* Thèse de doctorat en Sciences de l'Information Géographique, Université de Marne-la-Vallée, 234 p.

[Bonjour et al. 1994] Bonjour M., Falquet G. et Léonard M. 1994. Bases de concepts et intégration de bases de données, In *Actes du 10ème Congrès INFORSID*.

[Borges et al. 2002] Borges K.A.V., Davis C.A and Laender A.H.F. 1997. Integrity Constraints in Spatial Databases, In Doorn J.H. and Rivero LC. (Eds.) : *Database Integrity : Challenges and Solutions*, pp. 144-171.

[Boucelma et Lacroix 2001] Boucelma O. et Lacroix Z. 2001. InterMed : Interface de médiation pour les systèmes d'information, *Revue ISI : Ingénierie des Systèmes d'Information*, 6(3), pp. 33-60.

[Bouchon-Meunier 1999] Bouchon-Meunier B. 1999. *La logique floue*, Collection Que sais-je ?. Paris : Presses Universitaires de France, 3ème édition, 127 p.

[Branki et Defude 1998] Branki T. and Defude B. 1998. Data and Metadata: two-dimensional integration of heterogeneous spatial databases, In *Proceedings of the 8th International Symposium on Spatial Data Handling* (*SDH'98*), pp. 172-179.

[Brassel et Weibel 1988] Brassel K. et Weibel R. 1988. A review and conceptual framework of automated map generalization, *International Journal of Geographical Information Systems*, 2(3), pp. 229-244.

[Braun 2004] Braun A. 2004. *Plate-forme OXYGENE : prise en main et utilisation*, Document interne du laboratoire COGIT, IGN - Service de la Recherche, Saint-Mandé, 52 p.

[Breiman et al. 1984] Breiman L., Friedman J.H., Olshen R.A. and Stone C.J. 1984. *Classification and regression trees*, Technical report, Wadsworth International.

[Brodeur 2004] Brodeur J. 2004. *Interopérabilité des données Géospatiales : élaboration du concept de proximité géosémantique*, Thèse de doctorat en Géomatique, Université Laval, 247 p.

[Bruns et Egenhofer 1996] Bruns T.H. and Egenhofer M.J. 1996. Similarity of spatial scenes, In *Proceedings of the 7th International Symposium on Spatial Data Handling* (*SDH'96*), pp. 173-184.

[Buchanan et al. 1969] Buchanan B.G., Sutherland G.L. and Feigenbaum E.A. 1969. Heuristic DENDRAL : a program for generating explanatory hyptheses in organic chemistry, In Meltzer B., Michie D, Swann M (Eds.) : *Machine Intelligence 4*, Edinburgh University Press, pp. 209-254.

[Burrough et Franck 1996] Burrough, P. A. and Frank A. U. (Eds.) 1996. *Geographic Objects with Indeterminate Boundaries*, GISDATA Series. London : Taylor & Francis.

[Busse et al. 1999] Busse S., Kutsche R.-D., Leser U. and Weber H. 1999. *Federated Information Systems: Concepts, Terminology and Architectures*, Technical Report n°99-9, Technical University of Berlin, 38 p.

[Busse et al. 2000] Busse S., Kutsche R. and Leser U. 2000. Strategies for the Conceptual Design of Federated Information Systems, In *Proceedings of the 3rd International Workshop on Engineering Federated Information Systems (EFIS'00)*, pp. 23-32.

[Buttenfield et Delotto 1989] Buttenfield B.P. and Delotto J.S. 1989. *Multiple representations*, Report for the specialists meeting, National Center for Geographic Information and Analysis (NCGIA), Technical paper 89-3, 1989.

[Calvanese et al. 1998] Calvanese D., De Giacomo G., Lenzerini M., Nardi D. and Rosati R. 1998. Knowledge Representation Approach to Information Integration, In *Proceedings of the American Association for Artificial Intelligence (AAAI) Workshop on AI and Information Integration*, pp. 58-65.

[Calvanese et al. 2001] Calvanese D., De Giacomo G., Lenzerini M., Nardi D. and Rosati R. 2001. Data Integration in Data Warehousing, *International Journal of Cooperative Information Systems*, 10(3), pp. 237-271.

[Campbell 2000] Campbell J. 2000. *Map Use and Analysis*. Dubuque : McGraw-Hill, 4th edition.

[Car et Frank 1994] Car A. and Frank A. 1994. Modelling a Hierarchy of Space Applied to Large Road Networks, In *Proceedings of the International Workshop on Advanced Research in Geographic Information Systems (IGIS'94)*, Lecture Notes in Computer Science 884, Springer-Verlag, pp.15-24.

[Chawla et al. 2001] Chawla S., Shekha S., Wu W. and Ozesmi U. 2001. Modelling spatial dependencies for mining geospatial data, In Miller H. J. and Han J. (Eds.), *op. cit.,* chapter 6, pp. 131-159.

[Chrisman et Lester 1991] Chrisman N.R. and Lester M.K. 1991. A diagnostic test for error in categorical maps, In *Proceedings of the 10th International Symposium on Computer-Assisted Cartography (Auto-Carto 10)*, pp. 330-348.

[Christophe et Ruas 2002] Christophe S. and Ruas A. 2002. Detecting Building Alignments for Generalisation Purposes, In *Proceedings of 10th International Symposium on Spatial Data Handling (SDH'02)*, pp 419-432.

[Clancey 1983] Clancey W. 1983. The Epistemology of a Rule Based Expert System – A framework for Explanation*. Artificial Intelligence journal*, 20(3), pp.215-251.

[Clancey 1985] Clancey W. 1985. *Heuristic Classification*, Technical report, Stanford University (ref. STAN-CS-85-1066).

[Cleach et Fort 2003] Cleach A. et Fort J. 2003. *Intégration de bases de données spatiales, constitution d'une BD multi-représentation hydrographique*, Rapport de stage, École Navale, Brest, 50 p.

[Cockcroft 1997] Cockcroft S. 1997. A Taxonomy of Spatial Data Integrity Constraints, *GeoInformatica*, 1(4), pp. 327-343.

[Cockcroft 2004] Cockcroft S. 2004. The Design and Implementation of a Repository for the Management of Spatial Data Integrity Constraints, *GeoInformatica*, 8(1), pp. 49-69.

[Cohen 1995] Cohen W. 1995. Fast Effective Rule Induction, In *Proceedings of the 12th International Conference on Machine Learning* (*ICML'95*), pp. 115-123.

[Cohen 1998] Cohen W. 1998. Integration of Heterogeneous Databases without Common Domains Using Queries Based on Textual Similarity, In *Proceedings of the International Conference on Management of Data (SIGMOD'98)*, pp. 201-212.

[Colomb 1997] Colomb R.A. 1997. Impact of Semantic Heterogeneity on Federating Databases, *The Computer Journal*, 40(5), pp. 235-244.

[Congalton 1991] Congalton R.G. 1991. A review of assessing the accuracy of classifications of remotely sensed data, *Remote Sensing of Environment,* 37(1), pp. 35-46.

[Conrad et al. 1997] Conrad S., Eaglestone B., Hasselbring W., Roantree M., Saltor F., Schonhoff M., Strassler M. and Vermeer M. 1997. Research issues in federated database systems : report of EDBIS'97 Workshop, *SIGMOD Record*, 26(4), pp. 54-56.

[Cornuéjols et Miclet 2002] Cornuéjols A. et Miclet L. 2002*. Apprentissage artificiel, concepts et algorithmes*. Paris : Eyrolles, 591 p.

[Coster et Chermant 1989] Coster M. et Chermant J.L. 1989. *Précis d'analyse d'images.* Presses du CNRS, chap. 9, pp. 291-339.

[Cressie 1993] Cressie N.A. 1993. *Statistics for Spatial Data*. New York : John Wiley & Sons, revised edition, 928 p.

[Cruz et al. 2002] Cruz I.F., Rajendran A., Sunna W. and Wiegand N. 2002. Handling semantic heterogeneities using declarative agreements, In *Proceedings of the 10th International Symposium on Advances in Geographic Information Systems (ACM-GIS'02)*, pp. 168-174.

[Cuenin 1972] Cuenin, R. 1972. *Cartographie Générale. Notions générales et principes d'élaboration*. Paris : Eyrolles, Tome 1, 323 p.

[Cullot et al. 2003] Cullot N., Parent C., Spaccapietra S. et Vangenot C. 2003. Des ontologies pour données géographiques, *Revue internationale de Géomatique*, 13(3), pp. 285-306.

[David et Fasquel 1997] David B. & Fasquel P. 1997. Qualité d'une base de données géographiques : concepts et terminologie, *Bulletin d'Information de l'IGN*, 67, 51 p.

[David et al. 1993a] David B., Raynal L., Schorter G. and Mansart V. 1992. GeO_2 : Why objects in a geographical DBMS ?, In *Proceedings of the 3rd International Symposium on Advances in Spatial Databases (SSD'93)*, Lecture Notes in Computer Science 692, Springer-Verlag, pp. 264-276.

[David et al. 1993b] David J.-M., Krivine J.-P. and Simmons R. (Eds.) 1993. *Second generation Expert Systems*. Berlin : Springer-Verlag.

[Dayal 1983] Dayal U. 1983. Processing queries over generalized hierarchies in a multidatabase systems, In *Proceedings of the 9th International Conference on Very Large Data Bases (VLDB'83),* pp. 342-353.

[DB-Main 2004] DB-Main 2004. Atelier de Genie Logiciel DB-Main : http://www.db-main.be/

[Decatur 1997] Decatur S.E. 1997. Pac learning with constant-partition classification noise and applications to decision tree induction, In *Proceedings of the 14th International Conference on Machine Learning (ICML'97)*, pp. 83-91.

[Delobel et al. 2003] Delobel C., Reynaud C., Rousset M.-C., Sirot J.-P. and Vodislav D. 2003. Semantic Integration in Xyleme: a Uniform Tree-based Approach, *Journal on Data & Knowledge Engineering*, 44(2), pp 267-298.

[Devogele et al. 1996] Devogele T., Trévisan J. and Raynal L. 1995. Building a multi-scale database with scale-transaction relationships, In *Proceedings of the 7th International Conference on Spatial Data Handling (SDH'96)*, pp. 337-351.

[Devogele 1997] Devogele T. 1997. *Processus d'intégration et d'appariement de bases de données Géographiques. Application à une base de données routières multi-échelles*, Thèse de doctorat en Informatique, Université de Versailles, 205 p.

[Devogele et al. 1998] Devogele T., Parent C. and Spaccapietra S. 1998. On Spatial Database Integration, *International Journal of Geographical Information Science*, 12(4), pp. 335-352.

[Devogele et al. 2002] Devogele T., Badard T. et Libourel T. 2002. La problématique de la représentation multiple, In Ruas A. (Ed.), *op. cit.*, chapitre 3, pp. 55-74.

[Dietterich et Bakiri 1995] Dietterich T.G. and Bakiri G. 1995. Solving multiclass learning problems via error-correcting output codes, *Journal of Artificial Intelligence Research, 2,* pp. 263-286.

[Doan et al. 2003] Doan A., Domingos P. and Levy A. 2003. Learning to match the schemas of data sources : a multistrategy approach, *Machine Learning Journal*, 50(3), pp. 279-301.

[Doucet et Gançarski 2001] Doucet A. et Gançarski S. 2001. Entrepôts de données et bases de données multidimensionnelles, In Doucet A. et Jomier G. (Eds.) : *Bases de données et internet*. Paris : Hermès – Lavoisier, chapitre 12, pp. 367-394.

[Dougherty et al. 1995] Dougherty J., Kohavi R., Sahami M. 1995. Supervised and unsupervised discretization of continuous features, In *Proceedings of the 12th International Conference on Machine Learning (ICML'95)*, pp. 194-200.

[Duchêne et Regnauld 2002] Duchêne C. et Regnauld N. 2002. Le modèle AGENT, In Ruas A. (Ed.), *op. cit.*, chapitre 21, pp. 369-385.

[**Duchêne 2004**] Duchêne C. 2004. *Généralisation cartographique par agents communicants : le modèle CartACom*, Thèse de doctorat en Informatique, Université Paris 6, 230 p.

[Duckham et al. 2000] Duckham M., Drummond J. and Forrest D. 2000. Spatial data quality capture through inductive learning, *Spatial Cognition and Computation*, 2(4), pp. 261-282.

[Dunkars 2003] Dunkars M. 2003. Matching of Datasets, In *Proceedings of The 9th Scandinavian Research Conference on Geographical Information Science (ScanGIS'03)*, pp. 67-78.

[Dupin de Saint-Cyr et Loiseau 2000] Dupin de Saint-Cyr F. et Loiseau S. 2000. Validation et révision, In *Actes du 12ème Congrès Francophone AFRIF-AFIA de Reconnaissance des Formes et Intelligence Artificielle (RFIA'00)*, vol 1, pages 175-183.

[Egenhofer et Franzosa 1991] Egenhofer M.J. and Franzosa R.D. 1991. Point-set topological spatial relations, *International Journal of Geographical Information Systems*, 5(2), pp. 161-174.

[Egenhofer et Herring 1991] Egenhofer M.J. and Herring J.R. 1991. *A framework for the definition of topological relationships and an algebraic approach to spatial reasoning within this framework*, National Center for Geographic Information and Analysis (NCGIA), Technical Report 91-7, 55 p.

[Egenhofer et Franzosa 1994] Egenhofer M.J. and Franzosa R.D. 1994. On the equivalence of topological relations, *International Journal of Geographical Information Systems*, 8(6), pp. 133-152.

[Egenhofer et al. 1994] Egenhofer M.J., Clementini E. and Di Felice P. 1994. Evaluating inconsistencies among multiple representations, In *Proceedings of the 6th International Symposium on Spatial Data Handling (SDH'94),* pp. 901-920.

[El-Geresy et Abdelmoty 1998] El-Geresy B.A. and Abdelmoty A.I. 1998. A Qualitative Approach to Integration in Spatial Databases. In *Proceedings of the 9th International Conference on Database and Expert Systems Applications* (*DEXA'98*), Lecture Notes in Computer Science 1460, Springer-Verlag, pp. 280-289.

[Elias 2003] Elias B. 2003. Extracting landmarks with data mining methods, In *Proceedings of the International Conference on Spatial Information Theory (COSIT'03)*, Lecture Notes in Computer Science 2858, Springer-Verlag, pp. 375-389.

[Esposito et al. 1997] Esposito F., Lanza A., Malerba D. and Semeraro G. 1997. Machine learning for map interpretation : an intelligent tool for environmental planning, *Applied Artificial Intelligence*, 11(7-8), pp. 673-696.

[Ester et al. 1997] Ester M., Sander J. and Kriegel H.-P. 1997. *Spatial Data Mining: A Database Approach*, In *Proceedings of the 5th International Symposium on Advances in Spatial Databases (SSD'97)*, Lecture Notes in Computer Science 1262, Springer-Verlag, pp. 47-66.

[Estivill-Castro et Lee 2002] Estivill-Castro V. and Lee I. 2002. Multi-level clustering and its visualization for exploratory spatial analysis, *GeoInformatica*, 6(2), pp. 123-152.

[Euzenat 1999a] Euzenat J. 1999. *Représentations de connaissance. De l'approximation à la confrontation*, Mémoire d'habilitation à diriger des recherches, Université Joseph Fourier, 116 p.

[Euzenat 1999b] Euzenat J. 1999. La représentation des connaissances est-elle soluble dans le Web ?, *Document numérique*, 3(3-4), pp. 151-167.

[Fan et al. 2001] Fan W., Lu H., Madnick S.E., Cheung D.W. 2001. Discovering and reconciling data value conflicts for numerical data integration, *Information Systems*, 26(8), 635-656, 2001.

[Fayyad et al. 1996] Fayyad U. M., Piatetsky-Shapiro G., Smyth P. 1996. From Data Mining to Knowledge Discovery: An Overview, In Fayyad U. M., Piatetsky-Shapiro G., Smyth P., Uthurusamy R. (Eds.) : *Advances in Knowledge Discovery and Data Mining*. AAAI Press, pp. 1 - 34.

[Feigenbaum 1981] Feigenbaum E.A. 1981. Expert Systems in the 1980s*,* In Bond A. (Ed.): *State of the art report on machine intelligence*. Maidenhead : Pergamon-Infotech.

[Ferras 1995] Ferras R. 1995. Niveaux géographiques et échelles spatiales, In Bailly A., Ferras R. et Pumain D. (sous la direction de) : *Encyclopédie de géographie*. Paris : Economica, chapitre 22, pp. 403-421.

[Filho et al. 2002] Filho J. L., Iopche C. and Borges K.A.V. 2002. Analysis Patterns for GIS Data Schema Reuse on Urban Management Applications, *CLEI Electronic Journal*, 5(2), 15 p.

[Fonseca et al. 2002] Fonseca F.T., Egenhofer M., Agouris P. and Câmara G. 2002. Using ontologies for integrated Geographic Information Systems, *Transactions in GIS*, 6(3), pp. 231-257.

[Fonseca et al. 2003] Fonseca F.T., Davis C.A. and Câmara G. 2003. Bridging Ontologies and Conceptual Schemas in Geographic Information Integration, *GeoInformatica*, 7(4), pp. 355-378.

[Forgy 1982] Forgy C.L. 1982. RETE: a fast algorithm for the many patterns/many objects match problem, *Artificial Intelligence,* 19(1), pp. 17-37.

[Fougères et Trigano 1999] Fougères A.-J. and Trigano P. 1999. Construction de spécifications formelles à partir des spécifications rédigées en langage naturel, *Document numérique*, 3(3-4), pp. 215-239.

[Freska 1992] Freska C. 1992. Temporal reasoning based on semi-intervals, *Artificial Intelligence*, 54(1), pp. 199-227.

[Friedman-Hill 2003] Friedman-Hill E. 2003. *Jess in Action, Java Rule-based Systems*. Manning Publications, 480 p.

[Friis-Christensen 2003] Friis-Christensen A. 2003*. Issues in the Conceptual Modeling of Geographic Data*, PhD Thesis in Computer Science, University of Aalborg, 155 p.

[Gabay et Doytsher 2000] Gabay Y. and Doytsher Y. 2000. An approach to matching lines in partly similar engineering maps, *Geomatica*, 54(3), pp. 297-310.

[Gahegan 2002] Gahegan M. 2002. On the application of inductive machine learning tools to geographical analysis, *Geographical Analysis*, 32(1), pp. 113-139.

[Garcia-Molina et al. 1997] Garcia-Molina H., Papakonstantinou Y., Quass D., Rajaraman A., Sagiv Y., Ullman J., Vassalos V., Widom J. 1997. The TSIMMIS approach to mediation : Data models and Languages, *International Journal of Intelligent Information Systems*, 8(2), pp. 117-132.

[Gardarin 1999] Gardarin G. 1999. *Bases de données*. Paris : Eyrolles, 788 p.

[Géoroute 1999] Spécifications de contenu de Géoroute®, version 2.5., IGN, Saint-Mandé.

[Géoroute 2000] Spécifications de saisie de Géoroute®, version 2.1., IGN, Saint-Mandé.

[Gesbert et al. 2004] Gesbert N., Libourel T. et Mustière S. 2004. Apport des spécifications pour les modèles de bases de données géographiques, *Revue internationale de Géomatique*, 14(2), pp. 239-257.

[Goder 2003] Goder G. 2003. *Représentation comparée de schémas et spécifications de contenu*, Rapport de stage du « DESS Cartographie Numérique », Université Paris 1, 21 p.

[Goodchild et Jeansoulin 1998] Goodchild M. and Jeansoulin R. (Eds.) 1998*. Data Quality in Geographic Information : from Error to Uncertainty*. Paris : Hermes, 192 p.

[Goyal 2000] Goyal R.K. 2000. *Similarity assessment for cardinal directions between extended spatial objects*, PhD Thesis in Spatial Information Science and Engineering, University of Maine, 167 p.

[Grosso 2004] Grosso E. 2004. *Étude des carrefours d'un réseau routier*, Rapport de stage du « DESS Cartographie Numérique », Université Paris 1, 37 p.

[Gruber 1993] Gruber T.R. 1993. A translation approach to portable ontology specifications, *Knowledge Acquisition*, 5(2), pp. 199-220.

[Guarino 1998] Guarino N. 1998. Formal ontology and information systems, In *Proceedings of the 1st International Conference on Formal Ontology in Information Systems (FOIS'98)*, pp. 3-17.

[Guigo et al. 1995] Guigo M., Davoine P.-A., Dubus N., Guarniéri F., Richard B. et Bailly B. 1995. *Gestion de l'environnement et systèmes-experts*. Paris : Masson, Collection Géographie, 181 p.

[Guillaume 2001] Guillaume S. 2001. *Induction de règles floues interprétables*, Thèse de Doctorat en Informatique, Institut National des Sciences Appliquées de Toulouse, 195 p.

[Guptill 1989] Guptill S.C. 1989. Speculations on seamless, scaleless cartographic databases, In *Proceedings of the 9th International Symposium on Computer-Assisted Cartography (Auto-Carto 9),* pp. 436-443.

[Guptill et Morrison 1995] Guptill S.C. et Morrison J.L. (Eds.) 1995. *Elements of spatial data quality*. Oxford : Pergamon, 202 p.

[Hadzilacos et Tryfona 1998] Hadzilacos Th. and Tryfona N. 1997. Evaluation of Data Modeling Methods for Geographic Applications, *Australian Journal of Information Systems*, 6(1), pp. 15-26.

[Hakimpour et Geppert 2001] Hakimpour F. and Geppert A. 2001. Resolving Semantic Heterogeneity in Schema Integration: an Ontology Based Approach, In *Proceedings of 2nd International Conference on Formal Ontology in Information Systems (FOIS'01)*, pp. 297-308.

[Hakimpour 2003] Hakimpour F. 2003. *Using Ontologies to resolve Semantic Heterogeneity for Integrating Spatial Database Schemata*, PhD Thesis in Computer Science, University of Zurich, 191 p.

[Hammer et McLeod 1993] Hammer J. and McLeod D. 1993. An approach to resolving semantic heterogeneity in a federation of autonomous, heterogeneous database systems, *Journal for Intelligent and Cooperative Information Systems*, 2(1), pp. 51-83.

[Han et al. 1997] Han J., Koperski K. and Stefanovic N. 1997. GeoMiner: A System Prototype for Spatial Data Mining, *SIGMOD Record*, 26(2), pp. 553-556.

[Han et al. 2001] Han J., Kamber M. and Tung A.K.H. 2001. Spatial clustering methods in data mining, In Miller H. J. and Han J. (Eds.), *op. cit.,* chapter 8, pp. 188-217.

[Hass et al. 1997] Hass L.M., Kossman D., Wimmers E.L., Yang J. 1997. Optimizing queries across diverse data sources, In *Proceedings of the 23rd International Conference on Very Large Data Bases (VLDB'97)*, pp. 276-285.

[Hayes-Roth et al. 1983] Hayes-Roth F., Waterman D.A. and Lenat D.B. 1983. An Overview of Expert Systems, In Hayes-Roth F., Waterman D.A. and Lenat D.B. (Eds.) : *Building Expert Systems.* Addison-Wesley.

[Heiler et al. 1996] Heiler S., Miller R.J. and Ventrone V. 1996. Using Metadata to Address Problems of Semantic Interoperability in Large Object Systems, In *Proceedings of the IEEE Metadata Conference*.

[Inmon 1996] Inmon W.H. 1996. *Building the Data Warehouse.* New York : John Wiley & Sons, Second Edition, 401 p.

[Jahard et al. 2003] Jahard Y., Lemarié C. and Lecordix F. 2003. The implementation of new technology to automate map generalisation and incremental updating processes, In *Proceedings of the International Cartographic Conference* (*ICC'2003*), pp. 1149-1458.

[Jakobovits 1997] Jakobovits R. 1997. *Integrating Heterogeneous Autonomous Information Sources*, Technical Report TR-97-12-05, University of Washington, 28 p.

[Jaudoin et al. 2003] Jaudoin H., Rey C., Schneider M. and Vigier F. 2003. Interoperability of the Agricultural Information Systems: a Common ontological Approach for Various Exchange Type, In *Proceedings of the 4th Conference of the European Federation for Information Technology in Agriculture Food and the Environment (EFITA'03)*, pp. 293-299.

[Jeansoulin et Papini 2000] Jeansoulin R. et Papini O. 2000. Revision et information spatiale, In Prade H., Jeansoulin R. et Garbay C. (Eds.) : *Le temps, l'espace et l'évolutif en Sciences du Traitement de l'Information*. Toulouse : CEPADUES-editions, pp. 294-304.

[Kashyap et Sheth 1996] Kashyap V. et Sheth A. 1996. Semantic heterogeneity in Global Information Systems : the role of Metadata, Context and Ontologies, In Papazoglou M. and Schlageter G. (Eds.) : *Cooperative Information Systems : Current Trends and Directions*, pp. 139-178.

[Katsuno et Mendelzon 1991] Katsuno H. and Mendelzon A. 1991. Propositional Knowledge Base Revision and Minimal Change, *Artificial Intelligence*, 52(3) pp. 263-294.

[Kavouras et Kokla 2000] Kavouras M. and Kokla M. 2000. Ontology-Based Fusion of Geographic Databases, In *Spatial Information Management, Experiences and Visions for the*

21st Century Seminar, International Federation of Surveyors (FIG), Commission 3-WG 3.1, Athens, 7 p.

[Kayser 1997] Kayser D. 1997. *La représentation des connaissances*. Paris : Hermès, 308 p.

[Kilpeläinen 2000] Kilpeläinen T. 2000. Knowledge Acquisition for Generalization Rules, *Cartography and Geographic Information Science,* 27(1), pp.41-50.

[Kim et Seo 1991] Kim W. and Seo J. 1991. Classifying schematic and data heterogeneity in multidatabase system, *IEEE Computer*, 24(12), pp. 12-18.

[Kim et al. 1993] Kim W., Choi I., Gala S. and Scheevel M. 1993. On resolving schematic heterogeneity in multidatabase systems, *Distributed and Parallel Databases*, 1(3), pp. 251-279.

[Kirk et al. 1995] Kirk T., Levy A., Sagiv Y and Srivastava D. 1995. The Information Manifold, In *Proceedings of the AAAI Spring Symposium on Information Gathering in Distributed Heterogeneous Environments.*

[Koperski et Han 1995] Koperski K. and Han J. 1995. *Discovery of Spatial Association Rules in Geographic Information Databases*, In *Proceedings of the 4th International Symposium on Advances in Spatial Databases (SSD'95)*, Lecture Notes in Computer Science 951, Springer-Verlag, pp. 47-66.

[Koperski et al. 1998] Koperski K., Han J. and Stefanovic N. 1998. An Efficient Two-Step Method for Classification of Spatial Data*, In Proceedings of the International Symposium on Spatial Data Handling (SDH'98)*, pp. 45-54.

[Krivine et David 1991] Krivine J.-P. et David J.-M. 1992. L'acquisition des connaissances vue comme un processus de modélisation : méthodes et outils, *Intellectica*, 12, pp. 101-137.

[Larson et al. 1989] Larson J.A., Navathe S.B. and Elmasri R. 1989. A Theory of Attributed Equivalence in Databases with Application to Schema Integration, *IEEE Transactions on Software Engineering*, 15(4), pp. 449 - 463.

[Lassoued et al. 2004] Lassoued Y., Manoah S. et Boucelma O. 2004. Correspondances inter-schémas dans les SIG, In *Actes du 14ème Congrès Francophone AFRIF-AFIA de Reconnaissance de Formes et Intelligence Artificielle (RFIA'04)*, pp. 305-315*.*

[Laurini et Millert-Raffort 1993] Laurini R. and Millert-Raffort F. 1993*. Les bases de données en géomatique*. Paris : Hermès, 340 p.

[Laurini 1996] Laurini R. 1996. Raccordement géométrique de bases de données géographiques fédérées, *Revue ISI : Ingénierie des Systèmes d'Information*, 4(3), pp. 361-388.

[Lawrence 2001] Lawrence R. 2001. *Automatic Conflict Resolution to Integrate Relational Schema*, PhD Thesis in Computer Science, University of Manitoba, 162 p.

[Leclercq et al. 1999] Leclercq E., Benslimane D. and Yétongnon K. 1999. ISIS : A Semantic Mediation Model and an Agent Based Architecture for GIS Interoperability, In *Proceedings of the International Database Engineering and Applications Symposium (IDEAS'99)*, pp. 87-91.

[Lecordix et al. 1997] Lecordix F., Plazanet C. et Lagrange J.-P. 1997. A platform for research in generalization : application to caricature, *GeoInformatica*, 1(2), pp. 161-182.

[Lemarié 1996] Lemarié C. 1996. *État de l'art sur l'appariement*, Rapport technique DT/9600022/S-RAP, IGN, Service de la Recherche, Saint-Mandé, Juillet 1996.

[Lemarié et Bucaille 1998] Lemarié C. et Bucaille O. 1998. Spécifications d'un module générique d'appariement de données géographiques, In *Actes du 11ème Congrès Francophone AFRIF-AFIA de Reconnaissance de Formes et Intelligence Artificielle* (*RFIA'98*), pp. 397-406.

[Lépy 1997] Lépy N. 1997. Expertise et acquisition de connaissances en intelligence artificielle, In *Actes des 3ème rencontres doctorales SPI (SPI'97)*.

[Levy 1998] Levy A.Y. 1998. Combining Artificial Intelligence and Databases for Data Integration, In Wooldridge M. and Veloso M.M. (Eds.) : *Artificial Intelligence Today, Recent Trends and Developments,* Lecture Notes in Computer Science 1600, Springer-Verlag, pp. 249-268.

[Li et Clifton 2000] Li W.-S. and Clifton C. 2000. SEMINT: a tool for identifying attribute correspondences in heterogeneous databases using neural networks, *Data & Knowledge Engineering*, 33(1), pp. 49 – 84.

[Lightfoot 2001] Lightfoot D. 2001. *Formal Specification Using Z*. Macmillan Press, 2nd edition.

[Lim et al. 1994] Lim E.-P., Srivastava J. and Shekhar S. 1994. Resolving Attribute Incompatibility in Database Integration: An Evidential Reasoning Approach, In *Proceedings of the 10th International Conference on Data Engineering (ICDE'94)*, pp. 154-163.

[Litwin et al. 1989] Litwin W., Abdelattif A., Zeroual A., Nicoals B & Vigier P. 1989. MSQL : A multidatabase Language, *Information Science*, 48(1-3), pp. 59-101.

[Liu 1996] Liu H. 1996. Efficient rule induction from noisy data*, Expert Systems with Applications*, 10(2), pp. 275-280.

[Liu et al. 2002] Liu H., Hussain F., Tan C.L., Dash M. 2002. Discretization : an enabling technique, *Data Mining and Knowledge Discovery*, 6(4), pp. 393-423.

[Madhavan et al. 2001] Madhavan J., Bernstein P.A., Rahm E. 2001. Generic schema matching with Cupid. In *Proceedings of the 27th International Conference on Very Large Data Bases (VLDB'01)*, pp. 49-58.

[Malerba et al. 2002] Malerba D., Appice A. and Vacca N. 2002. SDMOQL : An OQL-based Data Mining Query Language for Map Interpretation, In *Proceedings of the International Workshop on Database Technologies for Data Mining (DTDM'02).*

[McBrien et Poulovassilis 1998] McBrien P. and Poulovassilis A. 1998. A Formalisation of Semantic Schema Integration, *Information Systems*, 23(5), pp. 307-334.

[Mérenne-Schoumaker 1996] Mérenne-Schoumaker B. 1996. *La localisation des services.* Paris : Nathan, Collection Géographie, 191 p.

[Metais et Sèdes 2002] Metais E. et Sèdes F. 2002. Appariement d'informations dans les entrepôts de données : quelques approches pour le filtrage flexible, *Revue I3 (Information - Interaction - Intelligence)*, 2(2), pp. 63-89.

[Michard 1998] Michard A. 1998. *XML, langage et applications*. Paris : Eyrolles.

[Miller et Han 2001] Miller H. J. and Han J. (Eds.) 2001. *Geographic Data Mining and Knowledge Discovery.* London : Taylor & Francis, 372 p.

[Miquel et al. 2002] Miquel M., Bédard Y. et Brisebois A. 2002. Conception d'entrepôts de données géospatiales à partir de sources hétérogènes. Exemples d'application en foresterie, *Revue ISI : Ingénierie des Systèmes d'Information*, 7(3), pp. 89-111.

[Mitchell 1997] Mitchell T.M. 1997. *Machine Learning*. Singapour : McGraw-Hill International Editions, 414 p.

[Morocho et al. 2003] Morocho V., Saltor F. and Pérez-Vidal L. 2003. Schema Integration on Federated Spatial DB Across Ontologies, In *Proceedings of the 5th Worskop on Engineering Federated Information Systems (EFIS'03)*, pp. 63-72.

[Muller 1997] Muller P.-A. 1997. *Modélisation objet avec UML.* Paris : Eyrolles, 421 p.

[MurMur 2002] MurMur (Collectif) 2002. *MurMur Project, Multi Representations – Multi Resolutions*, Final Report, 27 p.

[Musen 1993] Musen M.-A. 1993. An overview of knowledge acquisition. In David J.-M., Krivine J.-P. and Simons R. (Eds.), *op. cit*.

[Mustière et al. 2000a] Mustière S., Zucker J.-D. and Saitta L. 2000. Abstraction et Changement de Langage pour Automatiser la Généralisation Cartographique, In *Actes du 12ème Congrès Francophone AFRIF-AFIA de Reconnaissance de Formes et Intelligence Artificielle* (*RFIA'00*), vol.1, pp.411-418.

[Mustière et al. 2000b] Mustière S., Zucker J.-D. and Saitta L. 2000. An Abstraction-Based Machine Learning Approach to Cartographic Generalisation, In *Proceedings of the 9th International Symposium on Spatial Data Handling* (*SDH'00*), pp. 50-63.

[Mustière 2001] Mustière S. 2001. *Apprentissage supervisé pour la généralisation cartographique*, Thèse de doctorat en Informatique, Université Paris 6, 241 p.

[Mustière 2002] Mustière S. 2002. *Description des processus d'appariement mis en œuvre au COGIT*, Rapport technique SR/2002.0072, IGN - Service de la Recherche, Saint-Mandé, 18 p.

[Mustière et Zucker 2002] Mustière S. et Zucker J.-D. 2002. Généralisation cartographique et apprentissage automatique à partir d'exemples, In Ruas A. (Ed.), *op. cit.*, chapitre 20, pp. 353-368.

[Mustière 2003] Mustière S. 2003. *Chargement dans Oxygène d'un jeu de données IGN au format shape ou au format structuré à partir de fichiers shape*, Document interne du laboratoire COGIT, IGN - Service de la Recherche, Saint-Mandé, 10 p.

[Mustière et Bonin 2003] Mustière S. et Bonin O. 2003. *La carte topologique dans OXYGENE : schéma de développement des actions de recherche UNIBA et RISQ*, Rapport technique SR/2003.0093, IGN - Service de la Recherche, Saint-Mandé, 6 p.

[Mustière et al. 2003] Mustière S., Gesbert N. et Sheeren D. 2003. A formal Model for the Specifications of Geographic Databases, In *Proceedings of the International Workshop on Semantic Processing of Spatial Data (GeoPro'2003)*, pp. 152-159.

[Nyerges 1989] Nyerges T.H. 1989. Schema integration analysis for the development of GIS databases, *International Journal of Geographical Information Systems*, 3(2), pp. 153-183.

[OpenGIS 1999] The OpenGIS™ Abstract Specification, Topic 5: Features, Version 4, 45 p.

[OpenGIS 2001] The OpenGIS™ Abstract Specification, Topic 1: Feature Geometry (ISO 19107 Geographic Information - Spatial Schema), Version 5, 168 p.

[Openshaw et Openshaw 1997] Openshaw S. and Openshaw C. 1997. *Artificial intelligence in geography*. Chichester : John Wiley & Sons, 329 p.

[Paiva 1998] Paiva J.A. 1998. *Topological equivalence and similarity in multi-representation geographic databases*, PhD Thesis in Spatial Information Science and Engineering, University of Maine, 188 p.

[Pantazis et Donnay 1996] Pantazis D.N. et Donnay J.-P. 1996. *La conception de SIG : méthode et formalisme*. Paris : Hermès, 343 p.

[Pantazis et al. 2002] Pantazis D.N., Cornélis B., Billen R. and Sheeren D. 2001. Establishment of a geographic data dictionary : a case study of UrbIS 2©, the Brussels regional government GIS, *Computers, Environment and Urban Systems*, 26(1), pp. 3-17.

[Parent et Spaccapietra 1996] Parent Ch. et Spaccapietra S. 1996. Intégration de bases de données : panorama des problèmes et des approches, *Revue ISI : Ingénierie des Systèmes d'Information*, 4(3), pp. 333-359.

[Parent et al. 1996] Parent C., Spaccapietra S. et Devogele T. 1996. Conflicts in Spatial Database integration, In *Proceedings pf the 9th Conference on Parallel and Distributed Computing Systems (PDCS'96)*, pp. 772-778.

[Parent et al. 1998] Parent C., Spaccapietra S., Zimanyi E., Donini P. and Plazanet C. 1998. Modeling spatial data in the MADS conceptuel model, In *Proceedings of the 8th International Symposium on Spatial Data Handling* (*SDH'98*), pp. 138-150.

[Parent et Spaccapietra 2001] Parent C. et Spaccapietra S. 2001. Database Integration : the Key to Data Interoperability, In Papazoglou M., Spaccapietra S. and Tari Z. (Eds.) : *Advances in Object-Oriented Data Modeling*. MIT Press.

[Park 2001] Park J. 2001. Schema integration methodology and toolkit for heterogeneous and distributed geographic databases, *Journal of the Korea Industrial Information Systems Society*, 6(3), pp. 51-64.

[Pendyala 2002] Pendyala R.M. 2002. *Development of GIS-based conflation tools for data integration and matching*, Final Report, University of South Florida, 20 p.

[Peuquet 2001] Peuquet D.J. 2001. Making Space for Time: Issues in Space-Time Data Representation, *GeoInformatica*, 5(1), pp. 11-32.

[Plazanet et al. 1998] Plazanet C., Bigolin N.M. and Ruas A. 1998. Experiments with learning techniques for spatial model enrichment and line generalization, *GeoInformatica*, 2(4), pp. 315-333.

[Pumain et Saint-Julien 1997] Pumain D. et Saint-Julien Th. *L'analyse spatiale*. Paris : Armand Colin, Collection Cursus, série « Géographie », 167 p.

[Quinlan 1986] Quinlan J.R. 1986. Induction of decision trees, *Machine Learning*, 1, pp. 81-106.

[Quinlan 1993] Quinlan J.R. 1993. *C4.5 : Programs for machine learning*. San Francisco : Morgan Kaufmann, 302 p.

[Rahm et Bernstein 2001] Rahm E. and Bernstein P.A. 2001. A Survey of Approaches to Automatic Schema Matching, *Very Large Database Journal*, 10(4), pp. 334-350.

[Ram et Ramesh 1999] Ram S. and Ramesh V. 1999. Schema Integration: Past, Current and Future, In Elmagarmid A., Rusinkeiwicz M. and Sheth A.P. (Eds.) : *Management of Heterogeneous and Autonomous Database Systems*. San Francisco : Morgan Kaufmann, pp. 119-155.

[Ram et al. 2001] Ram S., Khatri V., Zhang L. and Zeng D.D. 2002. GeoCosm : A Semantics-Based Approach for Information Integration of Geospatial Data, In *Proceedings of the 20th International Conference on Conceptual Modeling (ER'01)*, Workshop DASWIS, Lecture Notes in Computer Science 2465, Springer-Verlag, pp. 152-165.

[Regnauld 1998] Regnauld N. 1998. *Généralisation du bâti : structure spatiale de type graphe et représentation cartographique*, Thèse de doctorat en Informatique, Université de Provence - Aix-Marseille 1, 191 p.

[Rey-Debove et Rey 1988] Rey-Debove J. et Rey A. (Eds.) 1988. *Le Petit Robert, dictionnaire alphabétique et analogique de la langue française*. Paris : Dictionnaires Le Robert.

[Richardson et Domingos 2003] Richardson M. and Domingos P. 2003. Learning with knowledge from multiple experts, In *Proceedings of the 20th International Conference on Machine Learning (ICML'03)*, pp. 624-631.

[Rivest et al. 2001] Rivest, S., Bédard, Y. et Marchand, P. 2001. Towards better support for spatial decision-making: Defining the characteristics of Spatial On-Line Analytical Processing (SOLAP), *Geomatica*, 55(4), pp. 539-555.

[Rodriguez 2000] Rodriguez A.M. 2000. *Assessing semantic similarities among spatial entity classes*, PhD Thesis in Spatial Information Science and Engineering, University of Maine, 179 p.

[Rousset et al. 2002] Rousset M.-C., Bidault A., Froidevaux C., Gagliardi H., Goasdoué F., Reynaud C. et Safar B. 2002. Construction de Médiateurs pour Intégrer des Sources d'Information Multiples et Hétérogènes : le Projet PICSEL, *Revue I^3: Information - Interaction – Intelligence*, 2(1), pp. 9-58.

[Ruas 1999] Ruas A. 1999. *Modèle de généralisation de données géographiques à base de contraintes et d'autonomie*, Thèse de Doctorat en Sciences de l'Information Géographique, Université de Marne-la-Vallée.

[Ruas 2002] Ruas A. (Ed.) 2002. *Généralisation et représentation multiple*, Traité IGAT – Information Géographique et Aménagement du Territoire. Paris : Hermès Science, 390 p.

[Ruas 2002a] Ruas A. 2002. Échelle et niveau de détail, In Ruas A. (Ed.), *op. cit.*, chapitre 1, pp. 25-44.

[Ruas 2002b] Ruas A. 2002. Les problématiques de l'automatisation de la généralisation, In Ruas A. (Ed.), *op. cit.*, chapitre 4, pp. 75-90.

[Rusinkiewitz et al. 1989] Rusinkiewitz M., Elmasri R, Czedjdo B., Georakopoulous D., Karabatis G., Jamoussi A., Loa K. and Li Y. 1989. Query processing in a heterogeneous multidatabase environment, In *Proceedings of the 1st Annual IEEE Symposium on Parallel and Distributed Processing*, pp. 162-169.

[Russell et Norvig 2003] Russell S. and Norvig P. 2003. *Artificial Intelligence : a modern approach*. Upper Saddle River : Prentice Hall, 2nd Edition, 932 p.

[Saitta et Zucker 2001] Saitta L. and Zucker J.-D. 2001. A Model of Abstraction in Visual Perception, *Applied Artificial Intelligence : Special Issue on Machine Learning in Computer Vision*, 15(8), pp. 761-776.

[Saltor et al. 1991] Saltor F., Castellanos M., García-Solaco M. 1991. Suitability of Data Models as Canonical Models for Federated Databases, *SIGMOD Record*, 20(4), pp. 44-48.

[Sardet 1999] Sardet E. 1999. *Intégration des approches modélisation conceptuelle et structuration documentaire pour la saisie, la représentation, l'échange et l'exploitation d'informations. Application aux catalogues de composants industriels*, Thèse de doctorat en Informatique, Université de Poitiers, 190 p.

[Savasere et al. 1991] Savasere A. , Sheth A., Gala S., Navathe S. and Marcus H. 1991. On applying classification to schema integration, In *Proceedings of 1st International Workshop on Interoperability in Multidatabase Systems (IMS'91)*, pp. 258-261.

[Schaffer 1993] Schaffer C. 1993. Selecting a classification method by cross-validation, *Machine Learning*, 13, pp. 135-143.

[Sester et al. 1998] Sester M., Anders K.-A. and Walter V. 1998. Linking objects of different spatial data sets by integration and aggregation, *GeoInformatica*, 2(4), pp. 335-358.

[Sester 2000] Sester M. 2000. Knowledge Acquisition for the Automatic Interpretation of Spatial Data, *International Journal of Geographical Information Science,* 14(1), pp. 1-24.

[Setra 1998] Setra (collectif) 1998. *The design of interurban intersections of major roads*, Rapport technique du Service d'Études Techniques des Routes et Autoroutes, Centre de la Sécurité et des Techniques Routières, Ministère de l'équipement, des transports, du logement, du tourisme et de la mer, 131 p.

[Sheth et Larson 1990] Sheth A. and Larson J. 1990. Federated database systems for managing distributed, heterogeneous and autonomous databases, *ACM Computing Surveys*, 22(3), pp. 183-236.

[Sheth et Kashyap 1992] Sheth A.P. and Kashyap V. 1992. So Far (Schematically) yet So Near (Semantically), In *Proceedings of the IFIP WG 2.6 Database Semantics Conference on Interoperable Database Systems (DS-5)*, pp. 283-312.

[Shewchuk 1996] Shewchuk J.R. 1996. Triangle: Engineering a 2D Quality Mesh Generator and Delaunay Triangulator, In *Proceedings of the 1st Workshop on Applied Computational Geometry*, pp. 124-133.

[Shortliffe 1976] Shortliffe E. 1976. *Computer Based Medical Consultations : MYCIN*. New York : Elsevier, 264 p.

[Siegel et Madnick 1991] Siegel M. and Madnick S.E. 1991. A Metadata Approach to Resolving Semantic Conflicts, In *Proceedings of the 17th International Conference on Very Large Data Bases (VLDB'91)*, pp. 133-145.

[Solar et Doucet 2002] Solar G.V. et Doucet A. 2002. Médiation de données : solutions et problèmes ouverts, In Actes des *2ème assises du GdR I³ : Information – Interaction – Intelligence*, pp. 217-231.

[Sotnykova 2003] Sotnykova A. 2003. *Design and implementation of federation of spatio-temporal databases : methods and tools*, Final Report Project (ref. BFR99/057), 64 p.

[Spaccapietra et Parent 1991] Spaccapietra S. et Parent C. 1991. Conflicts and Correspondence Assertions in Interoperable Databases, *SIGMOD Record*, 20(4), pp. 49-54.

[Spaccapietra et al. 1992] Spaccapietra S., Parent C. and Dupont Y. 1992. Model independent assertions for integration of heterogeneous schemas, *Very Large DataBase Journal*, 1(1), pp. 81-126.

[Spaccapietra et al. 1999] Spaccapietra S., Vangenot C., Parent C., Zimanyi E. 1999. MurMur: A Research Agenda on Multiple Representations, In *Proceedings of the International Symposium on Database Applications in Non-Traditional Environments (DANTE'99),* pp. 373-384.

[Spiess 1995] Spiess E. 1995. The need for generalization in a GIS environment, In Muller J.-C., Lagrange J.-P. and Weibel R. (Eds.) : *GIS and Generalization : methodology and practise*, GISDATA 1 Series. London : Taylor & Francis, pp. 31-46.

[Stoimenov et Đorđević-Kajan 2002] Stoimenov L. and Đorđević-Kajan S. 2002. Framework for Semantic GIS Interoperability, *FACTA Universitatis* : *Series Mathematics and Informatics*, 17, pp. 107-125.

[Strauch et al. 1998] Strauch J., Souza J. and Mattoso M. 1998. A methodology for GIS database integration, In *Proceedings of the IEEE Workshop on Knowledge and Data Engineering Exchange (KDEX'98)*, pp. 151-159.

[Tejada et al. 2001] Tejada S., Knoblock C.A. and Minton S. 2001. Learning object identification rules for identification integration, *Information Systems*, 26(8), pp. 607-633.

[Thomas 1996] Thomas J. 1996. *Vers l'intégration de l'apprentissage symbolique et l'acquisition de connaissances basée sur les modèles: le système ENIGME,* Thèse de doctorat en Informatique, Université Paris 6.

[Tomasic et al. 1998] Tomasic A., Raschid L. and Valduriez P. 1998. Scaling access to distributed heterogeneous data sources with Disco*, IEEE Transactions on Knowledge and Data Engineering,* 10(5), pp. 808-823.

[Trévisan 2005] Trévisan J. 2005. Dérivation de Bases de Données Cartographiques à partir d'une Base de Données Géographiques : application à la dérivation du bâti pour le 1:25.000 et le 1 :50.000 à partir de la BDTopo, *Bulletin Scientifique et Technique de l'IGN – Journées Recherche 2004*, 75, pp. 101-114.

[Tseng et al. 1992] Tseng F.S.-C., Chen A.L.P. and Yang W.-P. 1992. A probabilistic approach to query processing in heterogeneous database systems, In *Proceedings of the 2nd*

International Workshop on Research Issues on Data Engineering:Transaction and Query Processing (RIDE-TQP'92), pp. 176–183.

[Ubeda 1997] Ubeda T. 1997. Contrôle de la qualité spatiale des bases de données géographiques : cohérence topologique et corrections d'erreurs, Thèse de doctorat en Informatique, INSA-Lyon, 204 p.

[Valduriez et Ozsu 1999] Valduriez P. and Ozsu T. 1999. *Principles of Distributed Database Systems*. Upper Saddle River : Prentice Hall, 2nd edition, 562 p.

[Vangenot 2001] Vangenot C. 2001. *La multi-représentation dans les bases de données géographiques*, Thèse de doctorat en informatique, École Polytechnique Fédérale de Lausanne, 166 p.

[Vangenot et al. 2002] Vangenot C., Parent C. et Spaccapietra S. 2002. Modélisation et manipulation de données spatiales avec multireprésentation dans le modèle MADS, In Ruas A. (Ed.), *op. cit.*, chapitre 5, pp. 93-112.

[Van Lamsweerde 2000] Van Lamsweerde A. 2000. Formal Specification : a Roadmap, In *Proceedings of International Conference on Software Engineering (ICSE'00)*, pp. 147-159.

[Vauglin 1997] Vauglin F. 1997. *Modèles statistiques des imprécisions géométriques des objets géographiques linéaires*, Thèse de doctorat en Informatique, Université de Marne-la-Vallée, 325 p.

[Vauglin et Bel Hadj Ali 1998] Vauglin F. and Bel Hadj Ali A. 1998. Geometric matching of polygonal surfaces in GIS, In *ASPRS-RTI Annual Conference*, pp. 1511-1516.

[Vauglin 2002] Vauglin F. 2002. A practical study on precision and resolution in vector geographical database, In Shi W., Fisher P.F. and Goodchild M.F. (Eds.) : *Spatial Data Quality*. London : Taylor & Francis, pp. 127-139.

[Visser et al. 2002] Visser H., Stuckenschmidt H., Schuster G. and Vögele T. 2002. Ontologies for Geographic Information Processing, *Computers & Geosciences,* 28(1), pp. 103-117.

[Wache et al. 2001] Wache H., Vögele T., Visser U., Stuckenschmidt H., Schuster G, Neumann H and Hübner S. 2001. Ontology-based integration of information - A survey of existing approaches, In *Proceedings of the 17th International Joint Conference on Artificial Intelligence* (*IJCAI'01*), *Workshop: Ontologies and Information Sharing*.

[Walter et Fritsch 1999] Walter V. and Fritsch D. 1999. Matching Spatial Data Sets: a Statistical Approach, *International Journal of Geographical Information Science,* 13(5), pp. 445-473.

[Wang et Zhang 1996] Wang K. and Zhang W. 1996. Detecting data inconsistency for multidatabases, In *Proceedings of the 9th International Conference on Parallel and Distributed Computing Systems (PDCS'96)*, Vol 2, pp. 657-663.

[Weber et al. 2003] Weber C., Hirsch J., Schnell L. et Durrenberg M. 2003. Formes urbaines et transports de polluants, *Revue internationale de Géomatique*, 13(2), pp. 253-272.

[Weibel et al. 1995] Weibel R., Keller S., Reichenbacher T. 1995. Overcoming the knowledge acquisition bottleneck in map generalization : the role of interactive systems and computational intelligence, In *Proceedings of the International Conference on Spatial Information Theory (COSIT'95)*, Lecture Notes in Computer Science 988, Springer-Verlag, pp. 139-156.

[Wiederhold 1992] Wiederhold G. 1992. Mediators in the Architecture of Future Information Systems*, IEEE Computer*, 25(3), pp. 38-49.

[Wielinga et al. 1992] Wielinga B.J., Schreiber A.T. and Breuker A. 1992. KADS : a modelling approach to knowledge acquisition, *Knowledge Acquisition*, 4(1), pp. 5-54.

[Witten et Frank 1999] Witten I.H. and Frank E. 1999. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. San Francisco : Morgan Kaufmann, 416 p.

[Wohed 2000] Wohed P. 2000. Conceptual Patterns for Reuse in Information Systems Analysis, In *Proceedings of the 12th International Conference on Advanced Information Systems (CAiSE'00)*, Lecture Notes in Computer Science 1789, Springer-Verlag, pp. 157-175.

[Zeitouni et Yeh 1999] Zeitouni K. et Yeh L. 2000. Le data mining spatial et les bases de données spatiales, *Revue internationale de géomatique*, 9(4), pp 389-423.

[Zucker 2003] Zucker J.-D. 2003. A grounded theory of abstraction in artificial intelligence, *Philosophical Transactions: Biological Sciences* : *Special issue on Abstraction*, 358(1435), pp. 1293 – 1309.

[Zucker 2001] Zucker J.-D. 2001. *Changements de représentation, abstractions et apprentissages,* Mémoire d'habilitation à diriger des recherches, Université Paris 6, 124 p.

[Zweigenbaum 1999] Zweigenbaum P. 1999. Encoder l'information médicale : des terminologies aux systèmes de représentation des connaissances, *Innovation Stratégique en Information de Santé*, (2-3), pp. 27-47.

Articles publiés :

Sheeren D., Mustière S. and Zucker J.-D. 2004. How to Integrate Spatial Databases in a Consistent Way ?, In A. Benczur, J. Demetrovics and G. Gottlob (Eds.), *Proceedings of the 8th East European Conference on Advances in Databases and Information Systems (ADBIS'04)*, Lecture Notes in Computer Science 3255, Springer-Verlag, pp. 364-378.

Sheeren D., Mustière S. and Zucker J.-D. 2004. Consistency Assessment Between Multiple Representations of Geographical Databases: a Specification-Based Approach, In P. Fisher (Ed.), Developments in Spatial Data Handling, *Proceedings of the 11th International Symposium on Spatial Data Handling (SDH'04)*, Springer-Verlag, pp. 617-628.

Sheeren D. 2004. Apprentissage de concepts pour l'aide à l'interprétation des différences de représentation d'un même phénomène géographique, *Bulletin du Comité Français de Cartographie*, n°179 - Mars 2004, pp. 20-26.

Sheeren D. 2004. Étude de la cohérence inter-représentations : vers une meilleure intégration des bases de données spatiales, *Bulletin d'Information Scientifique et Technique de l'IGN - Journées Recherche 2004*, n°75, pp. 71-80.

Mustière S., Sheeren D. et Gesbert N. 2004. Unification des Bases de Données Géographiques : Recherches au Laboratoire COGIT de l'IGN, *Géomatique Expert*, n°32/33, février-mars 2004, pp. 50-54.

Mustière S., Gesbert N. and Sheeren D. 2003. A formal Model for the Specifications of Geographic Databases, In S. Levachkine, J. Serra and M. Egenhofer (Eds.), *Proceedings of the International Workshop on Semantic Processing of Spatial Data* (GeoPro'2003), pp. 152-159.

Sheeren D. 2003. Spatial Databases Integration : Interpretation of Multiple Representations by using Machine Learning Techniques, In *Proceedings of the 21st International Cartographic Conference* (ICC'2003), Durban, South Africa, pp. 235-245.

Sheeren D. 2002. L'appariement pour la constitution de bases de données géographiques multi-résolutions. Vers une interprétation des différences de représentations, *Revue internationale de Géomatique*, 12(2), pp. 151-168. (publié aussi dans les *Actes des 6ème Journées CASSINI'2002*, École Navale, Brest).

ANNEXES

Annexe 1 : Exemples de spécifications de classes de la BDTopo de l'IGN décrites selon le modèle défini en D.3.2 et instancié en XML.

Annexe 2 : Règles de production introduites dans le système-expert et définies dans le cadre de l'application sur les ronds-points (cf. E.3.5. et E.3.7.).

Annexe 3 : Description des algorithmes d'appariement utilisés dans le cadre de l'application sur les ronds-points (cf. E.3.4.).

Annexe 4 : Liste d'exemples d'apprentissage utilisés dans le cadre de l'application sur les ronds-points (cf. E.3.7.2.), permettant de prédire les conditions que doivent respecter les représentations de la BDCarto.

ANNEXE 1

Exemples de spécifications de classes de la BDTopo de l'IGN décrites selon le modèle défini en D.3.2 et instancié en XML.

DTD – Document Type Definition

```
<!ELEMENT GF_FEATURETYPE (NOM, DEFINITION, ATTRIBUT*, ASSOCIATION*, MODELISE+, CONTRAINTE_EXISTENCE?,
PRECISION_GEOMETRIQUE, PRECISION_SEMANTIQUE?, ACTUALITE?, EXHAUSTIVITE?, COMMENTAIRE*)>

<!ELEMENT NOM (#PCDATA)>

<!ELEMENT DEFINITION (#PCDATA)>

<!ELEMENT ATTRIBUT (NOM, DEFINITION?, TYPE_VALEUR, CONTRAINTE_NON_CHANGEMENT?, CONTRAINTE_VALEUR*,
PRECISION_SEMANTIQUE?)>
    <!ELEMENT TYPE_VALEUR (STRING|INT|FLOAT|BOOL|ENUM)>
        <!ELEMENT ENUM (VALEUR_CODEE+)>
            <!ELEMENT VALEUR_CODEE (NOM, CODE?, DEFINITION?)>
                <!ELEMENT CODE (#PCDATA)>
            <!ELEMENT STRING EMPTY>
            <!ELEMENT FLOAT EMPTY>
            <!ELEMENT BOOL EMPTY>
            <!ELEMENT INT EMPTY>
    <!ELEMENT CONTRAINTE_VALEUR ((VALEUR_CONTRAINTE | VALEUR_INTERDITE), (CONTRAINTE_SIMPLE |
    CONTRAINTE_COMPLEXE))>
        <!ELEMENT VALEUR_CONTRAINTE (#PCDATA)>
        <!ELEMENT VALEUR_INTERDITE (#PCDATA)>
    <!ELEMENT CONTRAINTE_NON_CHANGEMENT (CRITERE_GEOMETRIQUE, OPERATEUR, SEUIL, UNITE, MESURABLE)>

<!ELEMENT ASSOCIATION (NOM, DEFINITION, CLASSE_BD_EN_RELATION)*>
    <!ELEMENT CLASSE_BD_EN_RELATION (OBLIGATOIRE?, NOM, VALEUR_ATTRIBUT_IMPOSEE*)>
        <!ELEMENT OBLIGATOIRE EMPTY>
        <!ELEMENT VALEUR_ATTRIBUT_IMPOSEE (#PCDATA)>

<!ELEMENT MODELISE (MODELISATION, CONTRAINTE_MODELISATION?)>
    <!ELEMENT MODELISATION (DIMENSION+, TYPE_MODELISATION_XY+, TYPE_MODELISATION_Z*, ORIENTATION?)>
        <!ELEMENT DIMENSION (#PCDATA)>
        <!ELEMENT TYPE_MODELISATION_XY (#PCDATA)>
        <!ELEMENT TYPE_MODELISATION_Z (#PCDATA)>
        <!ELEMENT ORIENTATION (#PCDATA)>
    <!ELEMENT CONTRAINTE_MODELISATION (CONTRAINTE_SIMPLE | CONTRAINTE_COMPLEXE)>

<!ELEMENT CONTRAINTE_EXISTENCE (CONTRAINTE_SIMPLE | CONTRAINTE_COMPLEXE)>
    <!ELEMENT CONTRAINTE_SIMPLE (CONTRAINTE_DE_NATURE | CONTRAINTE_GEOMETRIQUE | CONTRAINTE_RELATION)>
        <!ELEMENT CONTRAINTE_DE_NATURE (CRITERE_NATURE, NIER)>
            <!ELEMENT CRITERE_NATURE (#PCDATA)>
            <!ELEMENT NIER (#PCDATA)>
        <!ELEMENT CONTRAINTE_GEOMETRIQUE (CRITERE_GEOMETRIQUE, OPERATEUR, SEUIL, UNITE, MESURABLE)>
            <!ELEMENT CRITERE_GEOMETRIQUE (#PCDATA)>
            <!ELEMENT OPERATEUR (#PCDATA)>
            <!ELEMENT SEUIL (#PCDATA)>
            <!ELEMENT UNITE (#PCDATA)>
            <!ELEMENT MESURABLE (#PCDATA)>
        <!ELEMENT CONTRAINTE_RELATION (ENTITE_EN_RELATION?,(CONTRAINTE_METRIQUE |
        CONTRAINTE_TOPOLOGIQUE | CONTRAINTE_DE_NATURE))>
            <!ELEMENT ENTITE_EN_RELATION (NOM, CONTRAINTE_SUR_ENTITE_REL?, CLASSE_BD_EN_RELATION*)>
                <!ELEMENT CONTRAINTE_SUR_ENTITE_REL (CONTRAINTE_SIMPLE | CONTRAINTE_COMPLEXE)>
            <!ELEMENT CONTRAINTE_METRIQUE (RELATION_METRIQUE, OPERATEUR, SEUIL?, UNITE?, MESURABLE)>
                <!ELEMENT RELATION_METRIQUE (#PCDATA)>
            <!ELEMENT CONTRAINTE_TOPOLOGIQUE (RELATION_TOPOLOGIQUE, MODALITE)*>
                <!ELEMENT RELATION_TOPOLOGIQUE (#PCDATA)>
                <!ELEMENT MODALITE (#PCDATA)>
    <!ELEMENT CONTRAINTE_COMPLEXE (TYPE_DE_LIEN, CONTRAINTE_COMPOSANTE+)>
        <!ELEMENT TYPE_DE_LIEN (#PCDATA)>
        <!ELEMENT CONTRAINTE_COMPOSANTE (CONTRAINTE_SIMPLE | CONTRAINTE_COMPLEXE)>

<!ELEMENT PRECISION_GEOMETRIQUE (EXACTITUDE_PLANIMETRIQUE, EXACTITUDE_ALTIMETRIQUE?, UNITE, TOLERANCE?)>
    <!ELEMENT EXACTITUDE_PLANIMETRIQUE (#PCDATA)>
        <!ELEMENT EXACTITUDE_ALTIMETRIQUE (#PCDATA)>
        <!ELEMENT TOLERANCE (#PCDATA)>

<!ELEMENT PRECISION_SEMANTIQUE (TAUX_DE_CONFUSION_MAX+, CLASSE_CONFONDUE*, ATTRIBUT_CONFONDU*)*>
    <!ELEMENT TAUX_DE_CONFUSION_MAX (#PCDATA)>
        <!ELEMENT CLASSE_CONFONDUE (#PCDATA)>
        <!ELEMENT ATTRIBUT_CONFONDU (#PCDATA)>

<!ELEMENT ACTUALITE (VALEUR)>
    <!ELEMENT VALEUR (#PCDATA)>

<!ELEMENT EXHAUSTIVITE (DEFICIT, EXCEDENT, UNITE, (CONTRAINTE_SIMPLE | CONTRAINTE_COMPLEXE)?)*>
    <!ELEMENT DEFICIT (#PCDATA)>
    <!ELEMENT EXCEDENT (#PCDATA)>

<!ELEMENT COMMENTAIRE (#PCDATA)>
```

CLASSE TRONCON COURS D'EAU

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<!-- Nom du fichier : Troncon_Cours_Eau.xml -->
  <!DOCTYPE GF_FEATURETYPE (View Source for full doctype...)>

   <GF_FEATURETYPE>
        <NOM>"Troncon cours d'eau"</NOM>
        <DEFINITION>"Tronçon du réseau hydrographique (fleuve, rivière, torrent,...) permettant
      un écoulement de l'eau naturel et permanent" </DEFINITION>
        <ASSOCIATION>
            <NOM>"Passe sous"</NOM>
            <DEFINITION>"Passe sous"</DEFINITION>
            <CLASSE_BD_EN_RELATION>
                <NOM>"Pont"</NOM>
            </CLASSE_BD_EN_RELATION>

            <NOM>"Passe sous"</NOM>
            <DEFINITION>"Passe sous"</DEFINITION>
            <CLASSE_BD_EN_RELATION>
                <NOM>"Pont surfacique"</NOM>
            </CLASSE_BD_EN_RELATION>

            <NOM>"Passe sous"</NOM>
            <DEFINITION>"Passe sous"</DEFINITION>
            <CLASSE_BD_EN_RELATION>
                <NOM>"Passerelle"</NOM>
            </CLASSE_BD_EN_RELATION>
        </ASSOCIATION>

        <MODELISE>
            <MODELISATION>
                <DIMENSION>"1"</DIMENSION>
                <TYPE_MODELISATION_XY>"Axe"</TYPE_MODELISATION_XY>
            </MODELISATION>
            <CONTRAINTE_MODELISATION>
              <CONTRAINTE_COMPLEXE>
                  <TYPE_DE_LIEN>"OU"</TYPE_DE_LIEN>

                <CONTRAINTE_COMPOSANTE>
                    <CONTRAINTE_SIMPLE>
                      <CONTRAINTE_RELATION>
                        <ENTITE_EN_RELATION>
                            <NOM>"Pont"</NOM>
                            <CLASSE_BD_EN_RELATION>
                                <NOM>"Pont"</NOM>
                            </CLASSE_BD_EN_RELATION>
                        </ENTITE_EN_RELATION>
                        <CONTRAINTE_TOPOLOGIQUE>
                            <RELATION_TOPOLOGIQUE>"SP"</RELATION_TOPOGIQUE>
                            <MODALITE>"Permise"</MODALITE>
                        </CONTRAINTE_TOPOLOGIQUE>
                      </CONTRAINTE_RELATION>
                    </CONTRAINTE_SIMPLE>
                </CONTRAINTE_COMPOSANTE>

                <CONTRAINTE_COMPOSANTE>
                    <CONTRAINTE_SIMPLE>
                      <CONTRAINTE_RELATION>
                        <ENTITE_EN_RELATION>
                            <NOM>"Pont"</NOM>
                            <CLASSE_BD_EN_RELATION>
                                <NOM>"Pont surfacique"</NOM>
                            </CLASSE_BD_EN_RELATION>
                        </ENTITE_EN_RELATION>
                        <CONTRAINTE_TOPOLOGIQUE>
                            <RELATION_TOPOLOGIQUE>"SP"</RELATION_TOPOLOGIQUE>
                            <MODALITE>"Permise"</MODALITE>
                        </CONTRAINTE_TOPOLOGIQUE>
                      </CONTRAINTE_RELATION>
                    </CONTRAINTE_SIMPLE>
                </CONTRAINTE_COMPOSANTE>

                <CONTRAINTE_COMPOSANTE>
                    <CONTRAINTE_SIMPLE>
                      <CONTRAINTE_RELATION>
                        <ENTITE_EN_RELATION>
                            <NOM>"Pont"</NOM>
                            <CLASSE_BD_EN_RELATION>
```

```
                                <NOM>"Passerelle"</NOM>
                            </CLASSE_BD_EN_RELATION>
                        </ENTITE_EN_RELATION>
                        <CONTRAINTE_TOPOLOGIQUE>
                            <RELATION_TOPOLOGIQUE>"SP"</RELATION_TOPOLOGIQUE>
                            <MODALITE>"Permise"</MODALITE>
                        </CONTRAINTE_TOPOLOGIQUE>
                    </CONTRAINTE_RELATION>
                </CONTRAINTE_SIMPLE>
            </CONTRAINTE_COMPOSANTE>
        </CONTRAINTE_COMPLEXE>
    </CONTRAINTE_MODELISATION>
</MODELISE>

<CONTRAINTE_EXISTENCE>
    <CONTRAINTE_SIMPLE>
        <CONTRAINTE_GEOMETRIQUE>
            <CRITERE_GEOMETRIQUE>"largeur"</CRITERE_GEOMETRIQUE>
            <OPERATEUR>"inférieure"</OPERATEUR>
            <SEUIL>"7.5"</SEUIL>
            <UNITE>"mètre"</UNITE>
            <MESURABLE>"Non"</MESURABLE>
        </CONTRAINTE_GEOMETRIQUE>
    </CONTRAINTE_SIMPLE>
</CONTRAINTE_EXISTENCE>

<PRECISION_GEOMETRIQUE>
    <EXACTITUDE_PLANIMETRIQUE>"2,5"</EXACTITUDE_PLANIMETRIQUE>
    <EXACTITUDE_ALTIMETRIQUE>"1"</EXACTITUDE_ALTIMETRIQUE>
    <UNITE>"mètre"</UNITE>
</PRECISION_GEOMETRIQUE>

<PRECISION_SEMANTIQUE>
    <TAUX_DE_CONFUSION_MAX>"2"</TAUX_DE_CONFUSION_MAX>
    <CLASSE_CONFONDUE>"Troncon cours d'eau temporaire"</CLASSE_CONFONDUE>
</PRECISION_SEMANTIQUE>

<EXHAUSTIVITE>
    <DEFICIT>"98"</DEFICIT>
    <EXCEDENT>"102"</EXCEDENT>
    <UNITE>"Pourcent"</UNITE>
</EXHAUSTIVITE>

</GF_FEATURETYPE>
```

CLASSE RESERVOIR D'EAU

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<!--  Nom du fichier : Reservoir_Eau.xml -->
<!DOCTYPE GF_FEATURETYPE (View Source for full doctype...)>

    <GF_FEATURETYPE>
        <NOM>"Réservoir d'eau"</NOM>
        <DEFINITION>"réservoir, souvent partiellement enterré, destiné à l'alimentation en eau
      d'une collectivité"</DEFINITION>
        <ATTRIBUT>
            <NOM>"Toponyme"</NOM>
            <TYPE_VALEUR>
                <STRING />
            </TYPE_VALEUR>
        </ATTRIBUT>
        <MODELISE>
         <MODELISATION>
            <DIMENSION>"2"</DIMENSION>
            <TYPE_MODELISATION_XY>"pourtour"</TYPE_MODELISATION_XY>
         </MODELISATION>
        </MODELISE>
        <CONTRAINTE_EXISTENCE>
         <CONTRAINTE_SIMPLE>
            <CONTRAINTE_DE_NATURE>
                <CRITERE_NATURE>"destiné à l'alimentation en eau d'une collectivité"
                </CRITERE_NATURE>
                <NIER>"Non"</NIER>
           </CONTRAINTE_DE_NATURE>
         </CONTRAINTE_SIMPLE>
        </CONTRAINTE_EXISTENCE>
        <PRECISION_GEOMETRIQUE>
            <EXACTITUDE_PLANIMETRIQUE>"2,5"</EXACTITUDE_PLANIMETRIQUE>
            <UNITE>"mètre"</UNITE>
        </PRECISION_GEOMETRIQUE>
        <PRECISION_SEMANTIQUE>
            <TAUX_DE_CONFUSION_MAX>"10"</TAUX_DE_CONFUSION_MAX>
            <CLASSE_CONFONDUE>"Point d'eau"</CLASSE_CONFONDUE>
            <TAUX_DE_CONFUSION_MAX>"2"</TAUX_DE_CONFUSION_MAX>
            <CLASSE_CONFONDUE>"Château d'eau"</CLASSE_CONFONDUE>
            <TAUX_DE_CONFUSION_MAX>"1"</TAUX_DE_CONFUSION_MAX>
            <CLASSE_CONFONDUE>"Bâtiment quelconque"</CLASSE_CONFONDUE>
       </PRECISION_SEMANTIQUE>
       <EXHAUSTIVITE>
            <DEFICIT>"95"</DEFICIT>
            <EXCEDENT>"105"</EXCEDENT>
            <UNITE>"Pourcent"</UNITE>
      </EXHAUSTIVITE>
    </GF_FEATURETYPE>
```

ANNEXE 2

Règles de production introduites dans le système-expert fondé sur JESS, définies dans le cadre de l'application sur les ronds-points (cf. E.3.5. et E.3.7.).

Règles du contrôle intra-base de Géoroute

```
;; ------------------------------------------------------------
;; BLOCKS RULES
;; ------------------------------------------------------------

(defclass rpxatraite interpretation.georoute_enrichie.routierInterpretation.RondPointComplexe)
(defclass controleintra
interpretation.georoute_enrichie.routierInterpretation.ControleIntraRondPointComplexe)


;;;;;;;;;;;;;;;;;; ########## Règles déduites des Spécifications ########## ;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;
;;     Règles pour l'étape du contrôle intra-base de Géoroute :
;;     Contrôle du diamètre, du sens du cycle, du nombre de nœuds, de la correspondance avec le carrefour
::     complexe.
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(defrule controle_diametre_non
   ( rpxatraite (longueurGrandAxe ?x&:(< ?x 25)) )
   ( controleintra (conformiteDiametre ?o&:(eq ?o nil)) (OBJECT ?controleintra) )
    =>
   (set ?controleintra conformiteDiametre "non conforme")
)


(defrule controle_diametre_oui
   ( rpxatraite (longueurGrandAxe ?x&:(> ?x 25)) )
   ( controleintra (conformiteDiametre ?o&:(eq ?o nil)) (OBJECT ?controleintra) )
   =>
   (set ?controleintra conformiteDiametre "conforme")
)


(defrule controle_sensCycle_non
   ( rpxatraite (sensCycle ?x&:(eq ?x "sens_non_giratoire")) )
   ( controleintra (conformiteSensCycle ?o&:(eq ?o nil)) (OBJECT ?controleintra) )
    =>
   (set ?controleintra conformiteSensCycle "non conforme")
)


(defrule controle_sensCycle_oui
   ( rpxatraite (sensCycle ?x&:(eq ?x "sens_giratoire")) )
   ( controleintra (conformiteSensCycle ?o&:(eq ?o nil)) (OBJECT ?controleintra) )
    =>
   (set ?controleintra conformiteSensCycle "conforme")
)


(defrule controle_nombreNoeuds_non
   ( rpxatraite (nombreNoeuds ?x&:(< ?x 2)) )
   ( controleintra (conformiteNombreNoeuds ?o&:(eq ?o nil)) (OBJECT ?controleintra) )
    =>
   (set ?controleintra conformiteNombreNoeuds "non conforme")
)


(defrule controle_nombreNoeuds_oui
   ( rpxatraite (nombreNoeuds ?x&:(> ?x 1)) )
   ( controleintra (conformiteNombreNoeuds ?o&:(eq ?o nil)) (OBJECT ?controleintra) )
    =>
   (set ?controleintra conformiteNombreNoeuds "conforme")
)
```

```
(defrule controle_correspondance_carrefour_absence_oui
   ( controleintra (conformiteNombreNoeuds ?x&:(eq ?x "non conforme"))
          (conformiteCorrespondanceCarrefour ?o&:(eq ?o nil))
          (OBJECT ?controleintra) )
   ( rpxatraite (source ?h&:(eq ?h "carrefour correspondant absent")) )
    =>
   (set ?controleintra conformiteCorrespondanceCarrefour "non correspondance conforme")
)


(defrule controle_correspondance_carrefour_absence_non
   ( controleintra (conformiteNombreNoeuds ?x&:(eq ?x "conforme"))
     (conformiteCorrespondanceCarrefour ?o&:(eq ?o nil))  (OBJECT ?controleintra) )
   ( rpxatraite (source ?h&:(eq ?h "carrefour correspondant absent")) )
    =>
   (set ?controleintra conformiteCorrespondanceCarrefour "non correspondance conforme (absence terre-plein
  ) ou non conforme (deficit)")
)


(defrule controle_correspondance_ carrefour_importe
   ( controleintra (conformiteCorrespondanceCarrefour ?o&:(eq ?o nil)) (OBJECT ?controleintra) )
   ( rpxatraite (source ?x&:(eq ?x "carrefour correspondant importe")) )
    =>
   (set ?controleintra conformiteCorrespondanceCarrefour "correspondance conforme")
)


(defrule controle_correspondance_carrefour_existant
   ( controleintra (conformiteCorrespondanceCarrefour ?o&:(eq ?o nil)) (OBJECT ?controleintra) )
   ( rpxatraite (source ?x&:(eq ?x "carrefour correspondant existant")) )
    =>
   (set ?controleintra conformiteCorrespondanceCarrefour "correspondance conforme")
)
```

Règles du contrôle intra-base de BDCarto

```
;; ----------------------------------------------------------
;; BLOCKS RULES
;; ------------------------------------------------------------

(defclass rpxatraite interpretation.bdcarto_enrichie.routierInterpretation.RondPointComplexe)
(defclass controleintra
interpretation.bdcarto_enrichie.routierInterpretation.ControleIntraRondPointComplexe)


;;;;;;;;;;;;;;;;;; ########## Règles déduites des Spécifications ########## ;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;
;;     Règles pour l'étape du contrôle intra-base de BDCarto
;;     Contrôle du diamètre, du nombre de nœuds, de la vocation de la liaison, de la correspondance avec le
;;     nœud GRP.
;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;


(defrule controleD_diametre_non
   ( rpxatraite (longueurGrandAxe ?x&:(< ?x 100)) )
   ( controleintra (conformiteDiametreSpec ?o&:(eq ?o nil)) (OBJECT ?controleintra) )
    =>
   (set ?controleintra conformiteDiametreSpec "non conforme")
)


 (defrule controleD_diametre_oui
   ( rpxatraite (longueurGrandAxe ?x&:(> ?x 100)) )
   ( controleintra (conformiteDiametreSpec ?o&:(eq ?o nil)) (OBJECT ?controleintra) )
   =>
   (set ?controleintra conformiteDiametreSpec "conforme")
)


(defrule controle_nombreNoeuds_non
   ( rpxatraite (nombreNoeuds ?x&:(< ?x 2)) )
   ( controleintra (conformiteNombreNoeuds ?o&:(eq ?o nil)) (OBJECT ?controleintra) )
    =>
   (set ?controleintra conformiteNombreNoeuds "non conforme")
)
```

```
(defrule controle_nombreNoeuds_oui
   ( rpxatraite (nombreNoeuds ?x&:(> ?x 1)) )
   ( controleintra (conformiteNombreNoeuds ?o&:(eq ?o nil)) (OBJECT ?controleintra) )
    =>
   (set ?controleintra conformiteNombreNoeuds "conforme")
)


(defrule controle_VocationTroncon_non
   ( rpxatraite (vocationTroncon ?x&:(eq ?x "non_bretelle")) )
   ( controleintra (conformiteVocationTroncons ?o&:(eq ?o nil)) (OBJECT ?controleintra) )
    =>
   (set ?controleintra conformiteVocationTroncons "non conforme")
)


(defrule controle_VocationTroncon_oui
   ( rpxatraite (vocationTroncon ?x&:(eq ?x "bretelle")) )
   ( controleintra (conformiteVocationTroncons ?o&:(eq ?o nil)) (OBJECT ?controleintra) )
    =>
   (set ?controleintra conformiteVocationTroncons "conforme")
)


(defrule controle_correspondance_absence_oui
   ( controleintra (conformiteNombreNoeuds ?x&:(eq ?x "non conforme"))
     (conformiteCorrespondanceNoeud ?o&:(eq ?o nil))  (OBJECT ?controleintra) )
   ( rpxatraite (source ?h&:(eq ?h "noeud (grand rond-point) correspondant absent")) )
    =>
   (set ?controleintra conformiteCorrespondanceNoeud "non correspondance conforme")
)


(defrule controle_correspondance_absence_non
   ( controleintra (conformiteNombreNoeuds ?x&:(eq ?x "conforme"))
     (conformiteCorrespondanceNoeud ?o&:(eq ?o nil)) (OBJECT ?controleintra) )
   ( rpxatraite (source ?h&:(eq ?h "noeud (grand rond-point) correspondant absent")) )
    =>
   (set ?controleintra conformiteCorrespondanceNoeud "non correspondance conforme (erreur de construction
  du rond-point) ou  non conforme (deficit)")
)


(defrule controle_correspondance_importe
   ( controleintra (conformiteCorrespondanceNoeud ?o&:(eq ?o nil)) (OBJECT ?controleintra) )
   ( rpxatraite (source ?x&:(eq ?x "noeud (grand rond-point) correspondant importe")) )
    =>
   (set ?controleintra conformiteCorrespondanceNoeud "correspondance conforme")
)


(defrule controle_correspondance_existant
   ( controleintra (conformiteCorrespondanceNoeud ?o&:(eq ?o nil)) (OBJECT ?controleintra) )
   ( rpxatraite (source ?x&:(eq ?x "noeud (grand rond-point) correspondant existant")) )
    =>
   (set ?controleintra conformiteCorrespondanceNoeud "correspondance conforme")
)
```

<u>Règles du contrôle inter-bases : approche par *prédiction, comparaison, classification*</u>

```
;; ------------------------------------------------------------
;; BLOCKS RULES
;; ------------------------------------------------------------


(defclass coupleatraite interpretation.appariement.CoupleAppariement)
(defclass rpGeo interpretation.georoute_enrichie.routierInterpretation.RondPointComplexe)
(defclass rpBdc interpretation.bdcarto_enrichie.routierInterpretation.RondPointSimple)


;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;
;;      Règles pour l'étape du contrôle inter-bases : prédiction des conditions sur les représentations
;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

```
;;;;;;;;;;;;;;;;;;; ########## Règles déduites des Spécifications ########## ;;;;;;;;;;;;;;;;;;;

;;;;;;;;;;;;;;;; PREDICTION BDCARTO ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(defrule rep_spec_carrefour_point
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "point"))
     (confianceAppariement ?c&:(eq ?c "Appariement certain")) (cardinaliteLien ?y&:(eq ?y "1 - 1"))
     (OBJECT ?coupleatraite)  )
    =>
   (set ?coupleatraite predictionSpecRepBDCarto "Carrefour_simple")
)


(defrule rep_spec_carrefour_surface
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "surface"))
     (confianceAppariement ?c&:(eq ?c "Appariement certain")) (cardinaliteLien ?y&:(eq ?y "1 - 1"))
     (OBJECT ?coupleatraite) )
   ( rpGeo (longueurGrandAxe ?j&:(< ?j 50)) )
    =>
   (set ?coupleatraite predictionSpecRepBDCarto "Carrefour_simple")
)


(defrule rep_spec_petit_rondpoint
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "surface"))
     (confianceAppariement ?c&:(eq ?c "Appariement certain")) (cardinaliteLien ?y&:(eq ?y "1 - 1"))
     (OBJECT ?coupleatraite) )
   ( rpGeo (longueurGrandAxe ?j&:(and (> ?j 50) (< ?j 100)) ) )
    =>
   (set ?coupleatraite predictionSpecRepBDCarto "Petit_RondPoint")
)


(defrule rep_spec_grand_rondpoint
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "surface"))
      (confianceAppariement ?c&:(eq ?c "Appariement certain")) (cardinaliteLien ?y&:(eq ?y "1 - 1"))
      (OBJECT ?coupleatraite))
   ( rpGeo (longueurGrandAxe ?j&:(> ?j 100) ) )
    =>
   (set ?coupleatraite predictionSpecRepBDCarto "Grand_RondPoint")
)

;;;;;;;;;;;;;;;; PREDICTION GEOROUTE ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(defrule rep_spec_rondPoint_sup100
   ( coupleatraite (natureObjBDCarto ?x&:(eq ?x "surface"))
      (confianceAppariement ?c&:(eq ?c "Appariement certain"))
      (cardinaliteLien ?y&:(eq ?y "1 - 1")) (OBJECT ?coupleatraite) )
    =>
   (set ?coupleatraite predictionSpecRepGeoroute "surface_Diam_>100")
)


(defrule rep_spec_rondPoint_50_100
   ( coupleatraite (natureObjBDCarto ?x&:(eq ?x "point"))
      (confianceAppariement ?c&:(eq ?c "Appariement certain"))
      (cardinaliteLien ?y&:(eq ?y "1 - 1")) (OBJECT ?coupleatraite))
   ( rpBdc (nature ?n&:(eq ?n "Petit rond-point")) )
    =>
   (set ?coupleatraite predictionSpecRepGeoroute "surface_50<_Diam_<100")
)


(defrule rep_spec_rondPoint_25_50
   ( coupleatraite (natureObjBDCarto ?x&:(eq ?x "point"))
      (confianceAppariement ?c&:(eq ?c "Appariement certain"))
      (cardinaliteLien ?y&:(eq ?y "1 - 1")) (OBJECT ?coupleatraite))
   ( rpBdc (nature ?n&:(eq ?n "Carrefour simple")) )
    =>
   (set ?coupleatraite predictionSpecRepGeoroute "surface_25<_Diam_<50 ou point_rondPointSimple")
)

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

```
;;;;;;;;;;;;;;;;;;; ########## Règles induites des données par apprentissage automatique  ##########

;;;;;;;;;;;;;;;; PREDICTION BDCARTO ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(defrule rep_carrefour_point
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "point"))
     (confianceAppariement ?c&:(eq ?c "Appariement certain")) (cardinaliteLien ?y&:(eq ?y "1 - 1"))
     (OBJECT ?coupleatraite))
    =>
   (set ?coupleatraite predictionRepBDCarto "Carrefour_simple")
)


(defrule rep_carrefour_surface
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "surface"))
     (confianceAppariement ?c&:(eq ?c "Appariement certain")) (cardinaliteLien ?y&:(eq ?y "1 - 1"))
     (OBJECT ?coupleatraite) )
   ( rpGeo (longueurGrandAxe ?j&:(< ?j 44)) )
    =>
   (set ?coupleatraite predictionRepBDCarto "Carrefour_simple")
)


(defrule rep_petit_rondpoint
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "surface"))
     (confianceAppariement ?c&:(eq ?c "Appariement certain")) (cardinaliteLien ?y&:(eq ?y "1 - 1"))
     (OBJECT ?coupleatraite) )
   ( rpGeo (longueurGrandAxe ?j&:(and (> ?j 44) (< ?j 84)) ) )
    =>
   (set ?coupleatraite predictionRepBDCarto "Petit_RondPoint")
)


(defrule rep_grand_rondpoint
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "surface"))
     (confianceAppariement ?c&:(eq ?c "Appariement certain")) (cardinaliteLien ?y&:(eq ?y "1 - 1"))
     (OBJECT ?coupleatraite))
   ( rpGeo (longueurGrandAxe ?j&:(> ?j 84) ) )
    =>
   (set ?coupleatraite predictionRepBDCarto "Grand_RondPoint")
)


;;;;;;;;;;;;;;;; PREDICTION GEOROUTE ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(defrule rep_rondPoint_surface
   ( coupleatraite (natureObjBDCarto ?x&:(eq ?x "surface"))
     (confianceAppariement ?c&:(eq ?c "Appariement certain")) (cardinaliteLien ?y&:(eq ?y "1 - 1"))
     (OBJECT ?coupleatraite) )
    =>
   (set ?coupleatraite predictionRepGeoroute "surface")
)


(defrule rep_rondPoint_ petitRP
   ( coupleatraite (natureObjBDCarto ?x&:(eq ?x "point"))
     (confianceAppariement ?c&:(eq ?c "Appariement certain")) (cardinaliteLien ?y&:(eq ?y "1 - 1"))
     (OBJECT ?coupleatraite))
   ( rpBdc (nature ?n&:(eq ?n "Petit rond-point")) )
    =>
   (set ?coupleatraite predictionRepGeoroute "surface")
)


(defrule rep_rondPoint_carrefourSimple
   ( coupleatraite (natureObjBDCarto ?x&:(eq ?x "point"))
     (confianceAppariement ?c&:(eq ?c "Appariement certain")) (cardinaliteLien ?y&:(eq ?y "1 - 1"))
     (OBJECT ?coupleatraite))
   ( rpBdc (nature ?n&:(eq ?n "Carrefour simple")) )
    =>
   (set ?coupleatraite predictionRepGeoroute "point_ou_surface")
)
```

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;
;;      Règles pour l'étape du contrôle inter-bases : comparaison des représentations stockées avec les
;;      conditions prédites sur les représentations
;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;; ########## Comparaison pour les règles déduites des Spécifications ##########
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;


;;;;;;;;;;;;;;; COMPARAISON Prédiction conditions BDCarto et représentations stockées BDCarto


;; Cas d'un surface BDCarto stockée
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(defrule comparaison_spec_surface_GrandRondPoint
   ( coupleatraite (natureObjBDCarto ?x&:(eq ?x "surface"))
     (predictionSpecRepBDCarto ?y&:(eq ?y "Grand_RondPoint")) (OBJECT ?coupleatraite) )
    =>
   (set ?coupleatraite comparaisonSpecPredictionBDC "Accord")
)


(defrule comparaison_spec_surface_CarrefourSimple
   ( coupleatraite (natureObjBDCarto ?x&:(eq ?x "surface"))
     (predictionSpecRepBDCarto ?y&:(eq ?y "Carrefour_simple")) (OBJECT ?coupleatraite) )
    =>
   (set ?coupleatraite comparaisonSpecPredictionBDC "Désaccord")
   (set ?coupleatraite conformiteSpecRepObjBDCarto "erreur de modélisation possible: surface au lieu de
   point")
)


(defrule comparaison_spec_surface_PetitRondPoint
   ( coupleatraite (natureObjBDCarto ?x&:(eq ?x "surface"))
     (predictionSpecRepBDCarto ?y&:(eq ?y "Petit_RondPoint")) (OBJECT ?coupleatraite) )
    =>
   (set ?coupleatraite comparaisonSpecPredictionBDC "Désaccord")
   (set ?coupleatraite conformiteSpecRepObjBDCarto "erreur de modélisation possible: surface au lieu de
   point")
)


;; Cas d'un point BDCarto stocké
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;::::::::

;; Carrefour Simple

(defrule comparaison_spec_point_CarrefourSimple_CarrefourSimple
   ( coupleatraite (natureObjBDCarto ?x&:(eq ?x "point"))
     (predictionSpecRepBDCarto ?y&:(eq ?y "Carrefour_simple")) (OBJECT ?coupleatraite) )
   ( rpBdc (nature ?n&:(eq ?n "Carrefour simple")) )
    =>
   (set ?coupleatraite comparaisonSpecPredictionBDC "Accord")
)


(defrule comparaison_spec_point_CarrefourSimple_PetitRondPoint
  ( coupleatraite (natureObjBDCarto ?x&:(eq ?x "point"))
    (predictionSpecRepBDCarto ?y&:(eq ?y "Petit_RondPoint")) (OBJECT ?coupleatraite) )
  ( rpBdc (nature ?n&:(eq ?n "Carrefour simple")) )
    =>
   (set ?coupleatraite comparaisonSpecPredictionBDC "Désaccord")
   (set ?coupleatraite conformiteSpecRepObjBDCarto "erreur sémantique possible: carrefour simple au lieu de
   petit rondPpoint")
)


(defrule comparaison_spec_point_CarrefourSimple_GrandRondPoint
   ( coupleatraite (natureObjBDCarto ?x&:(eq ?x "point"))
   (predictionSpecRepBDCarto ?y&:(eq ?y "Grand_RondPoint")) (OBJECT ?coupleatraite) )
   ( rpBdc (nature ?n&:(eq ?n "Carrefour simple")) )
   =>
   (set ?coupleatraite comparaisonSpecPredictionBDC "Désaccord")
   (set ?coupleatraite conformiteSpecRepObjBDCarto "erreur de modélisation possible: point au lieu d'une
   surface")
)
```

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;; Petit Rond-Point

(defrule comparaison_spec_point_PetitRondPoint_PetitRondPoint
   ( coupleatraite (natureObjBDCarto ?x&:(eq ?x "point"))
     (predictionSpecRepBDCarto ?y&:(eq ?y "Petit_RondPoint")) (OBJECT ?coupleatraite) )
   ( rpBdc (nature ?n&:(eq ?n "Petit rond-point")) )
    =>
   (set ?coupleatraite comparaisonSpecPredictionBDC "Accord")
)


(defrule comparaison_spec_point_PetitRondPoint_CarrefourSimple
   ( coupleatraite (natureObjBDCarto ?x&:(eq ?x "point"))
     (predictionSpecRepBDCarto ?y&:(eq ?y "Carrefour_simple")) (OBJECT ?coupleatraite) )
   ( rpBdc (nature ?n&:(eq ?n "Petit rond-point")) )
    =>
   (set ?coupleatraite comparaisonSpecPredictionBDC "Désaccord")
   (set ?coupleatraite conformiteSpecRepObjBDCarto "erreur sémantique possible: petit rondPpoint au lieu de
   carrefour simple ")
)


(defrule comparaison_spec_point_PetitRondPoint_GrandRondPoint
   ( coupleatraite (natureObjBDCarto ?x&:(eq ?x "point"))
     (predictionSpecRepBDCarto ?y&:(eq ?y "Grand_RondPoint")) (OBJECT ?coupleatraite) )
   ( rpBdc (nature ?n&:(eq ?n "Petit rond-point")) )
    =>
   (set ?coupleatraite comparaisonSpecPredictionBDC "Désaccord")
   (set ?coupleatraite conformiteSpecRepObjBDCarto "erreur de modélisation possible: point au lieu d'une
   surface")
)

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;


;;;;;;;;;;;;;;;;; COMPARAISON Prédiction conditions GEOROUTE et représentations stockées GEOROUTE


; Cas d'un point Géoroute stocké
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(defrule comparaison_spec_point_pointOuSurface25
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "point"))
     (predictionSpecRepGeoroute ?y&:(eq ?y "surface_25<_Diam_<50 ou point_rondPointSimple"))
     (OBJECT ?coupleatraite) )
    =>
   (set ?coupleatraite comparaisonSpecPredictionGeo "Accord")
)


(defrule comparaison_spec_point_Surface50_100
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "point"))
     (predictionSpecRepGeoroute ?y&:(eq ?y "surface_50<_Diam_<100")) (OBJECT ?coupleatraite) )
    =>
   (set ?coupleatraite comparaisonSpecPredictionGeo "Désaccord")
   (set ?coupleatraite conformiteSpecRepObjGeoroute "erreur de modélisation possible: point au lieu d'une
   surface")
)


(defrule comparaison_spec_point_SurfaceSup100
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "point"))
     (predictionSpecRepGeoroute ?y&:(eq ?y "surface_Diam_>100")) (OBJECT ?coupleatraite) )
    =>
   (set ?coupleatraite comparaisonSpecPredictionGeo "Désaccord")
   (set ?coupleatraite conformiteSpecRepObjGeoroute "erreur de modélisation possible: point au lieu d'une
   surface")
)

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

```
; Cas d'une surface Géoroute stockée
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(defrule comparaison_spec_surfaceInf25_pointOuSurface25
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "surface"))
     (predictionSpecRepGeoroute ?y&:(eq ?y "surface_25<_Diam_<50 ou point_rondPointSimple"))
     (OBJECT ?coupleatraite) )
   ( rpGeo (longueurGrandAxe ?j&:(< ?j 25) ) )
    =>
   (set ?coupleatraite comparaisonSpecPredictionGeo "Désaccord")
   (set ?coupleatraite conformiteSpecRepObjGeoroute "erreur géométrique possible: diamètre trop petit OU
    erreur de   modélisation: surface au lieu de point")
)


(defrule comparaison_spec_surfaceInf25_Surface50_100
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "surface"))
     (predictionSpecRepGeoroute ?y&:(eq ?y "surface_50<_Diam_<100")) (OBJECT ?coupleatraite) )
   ( rpGeo (longueurGrandAxe ?j&:(< ?j 25) ) )
    =>
   (set ?coupleatraite comparaisonSpecPredictionGeo "Désaccord")
   (set ?coupleatraite conformiteSpecRepObjGeoroute "erreur géométrique possible: diamètre trop petit")
)


(defrule comparaison_spec_surfaceInf25_SurfaceSup100
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "surface"))
     (predictionSpecRepGeoroute ?y&:(eq ?y "surface_Diam_>100")) (OBJECT ?coupleatraite) )
   ( rpGeo (longueurGrandAxe ?j&:(< ?j 25) ) )
    =>
   (set ?coupleatraite comparaisonSpecPredictionGeo "Désaccord")
   (set ?coupleatraite conformiteSpecRepObjGeoroute "erreur géométrique possible: diamètre trop petit")
)


;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;


(defrule comparaison_spec_surface25_50_pointOuSurface25
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "surface"))
     (predictionSpecRepGeoroute ?y&:(eq ?y "surface_25<_Diam_<50 ou point_rondPointSimple"))
     (OBJECT ?coupleatraite) )
   ( rpGeo (longueurGrandAxe ?j&:(and (> ?j 25) (< ?j 50)) ) )
    =>
   (set ?coupleatraite comparaisonSpecPredictionGeo "Accord")
)


(defrule comparaison_spec_surface25_50_Surface50_100
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "surface"))
     (predictionSpecRepGeoroute ?y&:(eq ?y "surface_50<_Diam_<100")) (OBJECT ?coupleatraite) )
   ( rpGeo (longueurGrandAxe ?j&:(and (> ?j 25) (< ?j 50)) ) )
    =>
   (set ?coupleatraite comparaisonSpecPredictionGeo "Désaccord")
   (set ?coupleatraite conformiteSpecRepObjGeoroute "erreur géométrique possible: diamètre trop petit")
)


(defrule comparaison_spec_surface25_50_SurfaceSup100
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "surface"))
     (predictionSpecRepGeoroute ?y&:(eq ?y "surface_Diam_>100")) (OBJECT ?coupleatraite) )
   ( rpGeo (longueurGrandAxe ?j&:(and (> ?j 25) (< ?j 50)) ) )
    =>
   (set ?coupleatraite comparaisonSpecPredictionGeo "Désaccord")
   (set ?coupleatraite conformiteSpecRepObjGeoroute "erreur géométrique possible: diamètre trop petit")
)


;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(defrule comparaison_spec_surface50_100_pointOuSurface25
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "surface"))
    (predictionSpecRepGeoroute ?y&:(eq ?y "surface_25<_Diam_<50 ou point_rondPointSimple"))
    (OBJECT ?coupleatraite) )
   ( rpGeo (longueurGrandAxe ?j&:(and (> ?j 50) (< ?j 100)) ) )
    =>
   (set ?coupleatraite comparaisonSpecPredictionGeo "Désaccord")
   (set ?coupleatraite conformiteSpecRepObjGeoroute "erreur géométrique possible: diamètre trop grand OU
   erreur de  modélisation: surface au lieu de point")
)
```

```
(defrule comparaison_spec_surface50_100_Surface50_100
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "surface"))
     (predictionSpecRepGeoroute ?y&:(eq ?y "surface_50<_Diam_<100")) (OBJECT ?coupleatraite) )
   ( rpGeo (longueurGrandAxe ?j&:(and (> ?j 50) (< ?j 100)) ) )
    =>
   (set ?coupleatraite comparaisonSpecPredictionGeo "Accord")
)


(defrule comparaison_spec_surface50_100_SurfaceSup100
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "surface"))
     (predictionSpecRepGeoroute ?y&:(eq ?y "surface_Diam_>100")) (OBJECT ?coupleatraite) )
   ( rpGeo (longueurGrandAxe ?j&:(and (> ?j 50) (< ?j 100)) ) )
    =>
   (set ?coupleatraite comparaisonSpecPredictionGeo "Désaccord")
   (set ?coupleatraite conformiteSpecRepObjGeoroute "erreur géométrique possible: diamètre trop petit")
)


;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(defrule comparaison_spec_surface100_pointOuSurface25
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "surface"))
     (predictionSpecRepGeoroute ?y&:(eq ?y "surface_25<_Diam_<50 ou point_rondPointSimple"))
     (OBJECT ?coupleatraite) )
   ( rpGeo (longueurGrandAxe ?j&:(> ?j 100) ) )
    =>
   (set ?coupleatraite comparaisonSpecPredictionGeo "Désaccord")
   (set ?coupleatraite conformiteSpecRepObjGeoroute "erreur géométrique possible: diamètre trop grand OU
    erreur de modélisation: surface au lieu de point")
)

(defrule comparaison_spec_surface100_Surface50_100
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "surface"))
     (predictionSpecRepGeoroute ?y&:(eq ?y "surface_50<_Diam_<100")) (OBJECT ?coupleatraite) )
   ( rpGeo (longueurGrandAxe ?j&:(> ?j 100) ) )
    =>
   (set ?coupleatraite comparaisonSpecPredictionGeo "Désaccord")
   (set ?coupleatraite conformiteSpecRepObjGeoroute "erreur géométrique possible: diamètre trop grand")
)

(defrule comparaison_spec_surface100_SurfaceSup100
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "surface")) (predictionSpecRepGeoroute ?y&:(eq ?y
   "surface_Diam_>100"))
     (OBJECT ?coupleatraite) )
   ( rpGeo (longueurGrandAxe ?j&:(> ?j 100) ) )
    =>
   (set ?coupleatraite comparaisonSpecPredictionGeo "Accord")
)



;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;; ########## Comparaison pour les règles induites des données ########## ;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;;;;;;;;;;;;;;; COMPARAISON Prédiction conditions BDCarto et représentations stockées BDCarto


;; Cas d'un surface BDCarto stockée
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(defrule comparaison_surface_GrandRondPoint
   ( coupleatraite (natureObjBDCarto ?x&:(eq ?x "surface"))
     (predictionRepBDCarto ?y&:(eq ?y "Grand_RondPoint")) (OBJECT ?coupleatraite) )
    =>
   (set ?coupleatraite comparaisonPredictionBDC "Accord")
)


(defrule comparaison_surface_CarrefourSimple
   ( coupleatraite (natureObjBDCarto ?x&:(eq ?x "surface"))
    (predictionRepBDCarto ?y&:(eq ?y "Carrefour_simple")) (OBJECT ?coupleatraite) )
    =>
   (set ?coupleatraite comparaisonPredictionBDC "Désaccord")
   (set ?coupleatraite conformiteRepObjBDCarto "erreur de modélisation possible: surface au lieu de point")
)
```

```
(defrule comparaison_surface_PetitRondPoint
   ( coupleatraite (natureObjBDCarto ?x&:(eq ?x "surface"))
     (predictionRepBDCarto ?y&:(eq ?y "Petit_RondPoint")) (OBJECT ?coupleatraite) )
    =>
   (set ?coupleatraite comparaisonPredictionBDC "Désaccord")
   (set ?coupleatraite conformiteRepObjBDCarto "erreur de modélisation possible: surface au lieu de point")
)


;; Cas d'un point BDCarto stocké
;;;;;;;;;;;;;;;;;;;;;;;;

(defrule comparaison_point_CarrefourSimple_CarrefourSimple
   ( coupleatraite (natureObjBDCarto ?x&:(eq ?x "point"))
     (predictionRepBDCarto ?y&:(eq ?y "Carrefour_simple")) (OBJECT ?coupleatraite) )
   ( rpBdc (nature ?n&:(eq ?n "Carrefour simple")) )
    =>
   (set ?coupleatraite comparaisonPredictionBDC "Accord")
)


(defrule comparaison_point_CarrefourSimple_PetitRondPoint
   ( coupleatraite (natureObjBDCarto ?x&:(eq ?x "point"))
     (predictionRepBDCarto ?y&:(eq ?y "Petit_RondPoint")) (OBJECT ?coupleatraite) )
   ( rpBdc (nature ?n&:(eq ?n "Carrefour simple")) )
    =>
   (set ?coupleatraite comparaisonPredictionBDC "Désaccord")
   (set ?coupleatraite conformiteRepObjBDCarto "erreur sémantique possible: carrefour simple au lieu de petit
    rondPpoint")
)


(defrule comparaison_point_CarrefourSimple_GrandRondPoint
   ( coupleatraite (natureObjBDCarto ?x&:(eq ?x "point"))
     (predictionRepBDCarto ?y&:(eq ?y "Grand_RondPoint")) (OBJECT ?coupleatraite) )
   ( rpBdc (nature ?n&:(eq ?n "Carrefour simple")) )
    =>
   (set ?coupleatraite comparaisonPredictionBDC "Désaccord")
   (set ?coupleatraite conformiteRepObjBDCarto "erreur de modélisation possible: point au lieu d'une
   surface")
)

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(defrule comparaison_point_PetitRondPoint_PetitRondPoint
   ( coupleatraite (natureObjBDCarto ?x&:(eq ?x "point"))
     (predictionRepBDCarto ?y&:(eq ?y "Petit_RondPoint")) (OBJECT ?coupleatraite) )
   ( rpBdc (nature ?n&:(eq ?n "Petit rond-point")) )
    =>
   (set ?coupleatraite comparaisonPredictionBDC "Accord")
)


(defrule comparaison_point_PetitRondPoint_CarrefourSimple
   ( coupleatraite (natureObjBDCarto ?x&:(eq ?x "point"))
     (predictionRepBDCarto ?y&:(eq ?y "Carrefour_simple")) (OBJECT ?coupleatraite) )
   ( rpBdc (nature ?n&:(eq ?n "Petit rond-point")) )
    =>
   (set ?coupleatraite comparaisonPredictionBDC "Désaccord")
   (set ?coupleatraite conformiteRepObjBDCarto "erreur sémantique possible: petit rondPpoint au lieu de
   carrefour simple ")
)


(defrule comparaison_point_PetitRondPoint_GrandRondPoint
   ( coupleatraite (natureObjBDCarto ?x&:(eq ?x "point"))
     (predictionRepBDCarto ?y&:(eq ?y "Grand_RondPoint")) (OBJECT ?coupleatraite) )
   ( rpBdc (nature ?n&:(eq ?n "Petit rond-point")) )
    =>
   (set ?coupleatraite comparaisonPredictionBDC "Désaccord")
   (set ?coupleatraite conformiteRepObjBDCarto "erreur de modélisation possible: point au lieu d'une
   surface")
)

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

```
;;;;;;;;;;;;;;; COMPARAISON Prédiction conditions GEOROUTE et représentations stockées GEOROUTE

; Cas d'un point Géoroute stocké
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(defrule comparaison_point_point
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "point")) (predictionRepGeoroute ?y&:(eq ?y "point"))
     (OBJECT ?coupleatraite) )
    =>
   (set ?coupleatraite comparaisonPredictionGeo "Accord")
)


(defrule comparaison_point_point_ou_surface
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "point"))
     (predictionRepGeoroute ?y&:(eq ?y "point_ou_surface")) (OBJECT ?coupleatraite) )
    =>
   (set ?coupleatraite comparaisonPredictionGeo "Accord")
)


(defrule comparaison_point_surface
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "point"))
    (predictionRepGeoroute ?y&:(eq ?y "point_ou_surface")) (OBJECT ?coupleatraite) )
    =>
   (set ?coupleatraite comparaisonPredictionGeo "Désaccord")
   (set ?coupleatraite conformiteRepObjGeoroute "erreur de modélisation possible: point au lieu d'une
   surface")
)


; Cas d'une surface Géoroute stockée
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(defrule comparaison_surface_surface
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "surface")) (predictionRepGeoroute ?y&:(eq ?y "surface"))
     (OBJECT ?coupleatraite) )
    =>
   (set ?coupleatraite comparaisonPredictionGeo "Accord")
)


(defrule comparaison_surface_point_ou_surface
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "surface"))
    (predictionRepGeoroute ?y&:(eq ?y "point_ou_surface")) (OBJECT ?coupleatraite) )
    =>
   (set ?coupleatraite comparaisonPredictionGeo "Accord")
)


(defrule comparaison_surface_point
   ( coupleatraite (natureObjGeoroute ?x&:(eq ?x "surface")) (predictionRepGeoroute ?y&:(eq ?y "point"))
     (OBJECT ?coupleatraite) )
    =>
   (set ?coupleatraite comparaisonPredictionGeo "Désaccord")
   (set ?coupleatraite conformiteRepObjGeoroute "erreur de modélisation possible : surface au lieu de point")
)

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;
;;
;;      Règles pour l'étape du contrôle inter-bases : classification  des couples en équivalence et incohérence
;;
;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;
;;;;;;;;;;;;;;;;;; ########## Classification des couples  ########## ;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;

;;;;;;;;;;;;;;;;;;;; Règles pour l'interprétation menée avec les spécifications


(defrule interpretation_accord_accord
   ( coupleatraite (comparaisonPredictionBDC ?x&:(eq ?x "Accord"))
    (comparaisonPredictionGeo ?y&:(eq ?y "Accord")) (OBJECT ?coupleatraite) )
    =>
   (set ?coupleatraite interpretationRepSpatiales "Equivalence")
)
```

```
(defrule interpretation_accord_desaccord
   ( coupleatraite (comparaisonPredictionBDC ?x&:(eq ?x "Accord"))
     (comparaisonPredictionGeo ?y&:(eq ?y "Désaccord")) (OBJECT ?coupleatraite) )
    =>
   (set ?coupleatraite interpretationRepSpatiales "Incohérence")
)


(defrule interpretation_desaccord_accord
   ( coupleatraite (comparaisonPredictionBDC ?x&:(eq ?x "Désaccord"))
     (comparaisonPredictionGeo ?y&:(eq ?y "Accord")) (OBJECT ?coupleatraite) )
    =>
   (set ?coupleatraite interpretationRepSpatiales "Incohérence")
)


(defrule interpretation_desaccord_desaccord
   ( coupleatraite (comparaisonPredictionBDC ?x&:(eq ?x "Désaccord"))
     (comparaisonPredictionGeo ?y&:(eq ?y "Désaccord")) (OBJECT ?coupleatraite) )
    =>
   (set ?coupleatraite interpretationRepSpatiales "Incohérence")
)


;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;Règles pour l'interprétation menée avec les connaissances induites des données

;;  (les règles sont similaires aux précédentes, seuls les attributs dans lesquels on affecte les valeurs
;;  d'interprétation sont différents.)

defrule interpretation_spec_accord_accord
   ( coupleatraite (comparaisonSpecPredictionBDC ?x&:(eq ?x "Accord"))
     (comparaisonSpecPredictionGeo ?y&:(eq ?y "Accord")) (OBJECT ?coupleatraite) )
    =>
   (set ?coupleatraite interpretationRepSpatialesSpec "Equivalence")
)


(defrule interpretation_spec_accord_desaccord
   ( coupleatraite (comparaisonSpecPredictionBDC ?x&:(eq ?x "Accord"))
     (comparaisonSpecPredictionGeo ?y&:(eq ?y "Désaccord")) (OBJECT ?coupleatraite) )
    =>
   (set ?coupleatraite interpretationRepSpatialesSpec "Incohérence")
)


(defrule interpretation_spec_desaccord_accord
   ( coupleatraite (comparaisonSpecPredictionBDC ?x&:(eq ?x "Désaccord"))
      (comparaisonSpecPredictionGeo ?y&:(eq ?y "Accord")) (OBJECT ?coupleatraite) )
    =>
   (set ?coupleatraite interpretationRepSpatialesSpec "Incohérence")
)


(defrule interpretation_spec_desaccord_desaccord
   ( coupleatraite (comparaisonSpecPredictionBDC ?x&:(eq ?x "Désaccord"))
     (comparaisonSpecPredictionGeo ?y&:(eq ?y "Désaccord")) (OBJECT ?coupleatraite) )
    =>
  (set ?coupleatraite interpretationRepSpatialesSpec "Incohérence")
)
```

ANNEXE 3

Description des algorithmes d'appariement utilisés dans le cadre de l'application sur les ronds-points (cf. E.3.4.).

Algorithmes développés pour calculer les correspondances entre ronds-points de la BDCarto et Géoroute

Nous proposons une méthode d'appariement qui calcule les liens entre les objets dans les deux sens (BDCarto → Géoroute et Géoroute → BDCarto). Elle se compose des étapes suivantes :

1. Appariement des ronds-points complexes BDCarto et Georoute ;
2. Appariement des ronds-points complexes BDCarto et des ronds-points simples Georoute ;
3. Appariement des ronds-points complexes Georoute et des ronds-points simples BDCarto ;
4. Appariement des ronds-points simples Georoute et des ronds-points simples BDCarto ;
5. Recherche des appariements 1-0 ('Petit Rond-point' BDCarto - Georoute) ;

La première étape vise à établir les liens entre les ronds-points détaillés surfaciques des deux bases. La base choisie comme référence est la BDCarto, c'est-à-dire la base de plus faible résolution. Toute la liste des ronds-points détaillés de cette base est ainsi parcourue et on vérifie qu'il existe une intersection avec un rond-point détaillé de l'autre base. Si un seul objet est candidat (intersection non vide), un couple d'objets appariés est créé et le lien est qualifié. On indique que les objets ont été appariés par intersection et que le lien est probablement juste. Si plusieurs objets sont candidats, le couple est créé mais le lien est considéré comme anormal. Il se peut que l'objet dans Géoroute soit représenté par un point (ce qui signifie que les représentations sont incohérentes). Dans ce cas, aucun candidat n'est trouvé à ce stade. Le lien sera détecté à l'étape suivante. Le pseudo-code du premier algorithme est le suivant :

```
Recherche Appariement SurfaceRPX_BDCarto – SurfaceRPX_Géoroute
    Tant que liste SurfaceRPX_BDCarto a un suivant
       Récupère SurfaceRPX_BDCarto suivante
       Crée Couple_Appariement
       Affecte SurfaceRPX_BDCarto au Couple_Appariement
       Tant que liste SurfaceRPX_Géoroute a un suivant
          Si SurfaceRPX_BDCarto intersecte SurfaceRPX_Géoroute
                Affecte SurfaceRPX_Géoroute au Couple_Appariement
          FinSi
       FinTant que
       Test la cardinalité de Géoroute pour le Couple_Appariement
        Si == 1,   critèreAppariement = « intersection »
                   confianceAppariement = « lien potentiellement juste »
        Si > 1,    critèreAppariement = « intersection »
                   confianceAppariement = « cardinalitéGeo > 1 anormal»
       FinTest
    FinTant que
FinRecherche Appariement SurfaceRPX_BDCarto – SurfaceRPX_Géoroute
```

Lors de la deuxième étape, les liens entre les objets surfaciques de la BDCarto et les objets ronds-points simples de Géoroute sont recherchés (on teste l'intersection également). En principe, ces liens ne devraient pas exister mais en raison d'erreurs de saisie dans les bases, quelques liens de ce type apparaissent. L'algorithme est le suivant :

```
Recherche Appariement SurfaceRPX_BDCarto – NoeudRPS_Géoroute
    Tant que liste SurfaceRPX_BDCarto a un suivant
       Récupère SurfaceRPX_BDCarto suivante
       Récupère Couple_Appariement de SurfaceRPX_BDCarto
       Si SurfaceRPX_BDCarto du Couple_Appariement n'est pas apparié
           Tant que liste NoeudRPS_Géoroute a un suivant
                Si NoeudRPS_Géoroute intersecte SurfaceRPX_BDCarto
                      Affecte NoeudRPS_Géoroute au Couple_Appariement
                FinSi
           FinTant que
        Test la cardinalité de Géoroute pour le Couple_Appariement
         Si == 0, critèreAppariement = « intersection »
                  confianceAppariement = « cardinalitéGeo anormal et
                                         lien non conforme»
         Si == 1,     critèreAppariement = « intersection »
                      confianceAppariement = « lien non conforme »
         Si > 1,      critèreAppariement = « intersection »
                      confianceAppariement = « cardinalitéGeo anormal et
                                         lien non conforme»
        FinTest
       FinSi
    FinTant que
FinRecherche Appariement SurfaceRPX_BDCarto – NoeudRPS_Géoroute
```

A l'issue de ce deuxième appariement, tous les ronds-points détaillés de la BDCarto sont censés être appariés car il n'est pas possible qu'un rond-point détaillé existant dans cette base ne soit pas représenté dans Géoroute, à moins d'une erreur (oubli ou représentation erronée dans Géoroute, erreur dans la BDCarto, extraction d'un objet rond-point qui n'en est pas un).

Le troisième algorithme développé a pour objectif de compléter l'appariement des ronds-points complexes de Géoroute. Certains de ces objets ont déjà été appariés lors de la première étape mais ils s'agit maintenant de rechercher les liens pour les objets qui n'ont pas de correspondants. Puisque les relations avec les ronds-points complexes de la BDCarto ont déjà été calculées, les seuls candidats à l'appariement sont les ronds-points simples de la BDCarto. Le pseudo-code est le suivant :

```
Recherche Appariement SurfaceRPX_Géoroute – NoeudRPS_BDCarto (intersec)
    Tant que liste SurfaceRPX_Géoroute a un suivant
       Récupère SurfaceRPX_Géoroute suivante
       Si SurfaceRPX_Géoroute n'est pas associé à un Couple_Appariement
          Crée Couple_Appariement
          Affecte SurfaceRPX_Géoroute au Couple_Appariement
          Tant que liste NoeudRPS_BDCarto a un suivant
              Si NoeudRPS_BDCarto intersecte SurfaceRPX_Géoroute
                 Affecte NoeudRPS_BDCarto au Couple_Appariement
              FinSi
          FinTant que
          Test la cardinalité de BDCarto pour le Couple_Appariement
           Si == 1,    critèreAppariement = « intersection »
                       confianceAppariement = « lien potentiellement juste »
```

```
          Si > 1,     critèreAppariement = « intersection »
                      confianceAppariement = « cardinalitéBDC > 1 anormal »
        FinTest
      FinSi
    FinTant que
FinRecherche Appariement SurfaceRPX_Géoroute – NoeudRPS_BDCarto (intersec)
```

Pour chaque rond-point complexe dans Géoroute, on a donc maintenant recherché son homologue dans la BDCarto. Toutefois, cette recherche n'a été effectuée que sur la base d'un critère d'intersection. Un tel critère n'est pas suffisant car suite aux différences de qualité des bases, il est possible que les ronds-points n'aient pas de correspondant qui les intersectent. Nous devons donc également effectuer une recherche des liens d'appariement sur base d'un critère de proximité. Ce critère n'est pris en compte que pour les correspondances entre les ronds-points complexes et les ronds-points simples et entre les ronds-points simples. Nous considérons que le calcul des correspondances entre les ronds-points complexes des deux bases peut se faire uniquement à partir du critère d'intersection. L'algorithme est le suivant :

```
Recherche Appariement SurfaceRPX_Géoroute – NoeudRPS_BDCarto (distance)
    Tant que liste SurfaceRPX_Géoroute a un suivant
      Récupère SurfaceRPX_Géoroute suivante
      Récupère Couple_Appariement de SurfaceRPX_Géoroute
      Si SurfaceRPX_Géoroute du Couple_Appariement n'est pas apparié
        Calcul barycentre(SurfaceRPX_Géoroute)
        Tant que liste NoeudRPS_BDCarto a un suivant
          Si NoeudRPS_BDCarto n'est pas apparié
              Calcul distance(Bary_SurfaceRPX_Géoroute,NoeudRPS_BDCarto)
              Si distance < 50
                    Affecte NoeudRPS_BDCarto au Couple_Appariement
              FinSi
          FinSi
        FinTant que
        Test la cardinalité de BDCarto pour le Couple_Appariement
         Si == 0,    critèreAppariement = « distance »
                    confianceAppariement = « cardinalité à étudier »
         Si == 1,    critèreAppariement = « distance »
                     confianceAppariement = « lien potentiellement juste »
         Si > 1,     critèreAppariement = « distance »
                  confianceAppariement = « cardinalitéBDC > 1 anormal »
        FinTest
      FinSi
    FinTant que
FinRecherche Appariement SurfaceRPX_Géoroute – NoeudRPS_BDCarto (distance)
```

Le seuil de 50 m qui correspond à la distance maximale de recherche admise a été fixée de manière empirique.

Il reste à présent à apparier la majorité des ronds-points simples des deux bases. Pour cette tâche, nous n'avons pas calculé les liens de la BDCarto vers Géoroute pour les nœuds de type « carrefour simple » car la classe des ronds-points simples de la BDCarto de cette nature contient beaucoup plus d'objets qu'elle ne devrait. Les liens ont donc été définis dans le sens inverse sur base d'un critère de distance euclidienne. L'algorithme est le suivant :

```
Recherche Appariement NoeudRPS_Géoroute – NoeudRPS_BDCarto (distance)
    Tant que liste NoeudRPS_Géoroute a un suivant
       Récupère NoeudRPS_Géoroute suivant
       Si NoeudRPS_Géoroute n'est pas associé à un Couple_Appariement
          Crée Couple_Appariement
          Affecte au NoeudRPS_Géoroute au Couple_Appariement
          Tant que liste NoeudRPS_BDCarto a un suivant
               Si NoeudRPS_BDCarto n'est pas apparié
                   Calcul distance(NoeudRPS_Géoroute,NoeudRPS_BDCarto)
                   Si distance < 50
                       Affecte NoeudRPS_BDCarto au Couple_Appariement
                   FinSi
               FinSi
          FinTant que
          Test la cardinalité de BDCarto pour le Couple_Appariement
           Si == 0,    critèreAppariement = « distance »
                      confianceAppariement = « cardinalité à étudier »
           Si == 1,    critèreAppariement = « distance »
                       confianceAppariement = « lien potentiellement juste »
           Si > 1,     critèreAppariement = « distance »
                     confianceAppariement = « cardinalitéBDC > 1 anormal »
          FinTest
       FinSi
    FinTant que
FinRecherche NoeudRPS_Géoroute – NoeudRPS_BDCarto (distance)
```

Après cette étape, nous avons cherché à détecter les ronds-points simples de la BDCarto de type 'petit rond-point' non appariés. Ceux-ci devraient l'être au regard des spécifications de chaque base. L'algorithme développé à cette fin est le suivant :

```
Recherche NoeudRPS_BDCarto de type 'petit rond-point' non apparié
    Tant que liste NoeudRPS_BDCarto a un suivant
       Récupère NoeudRPS_BDCarto suivant
       Si NoeudRPS_BDCarto n'est pas associé à un Couple_Appariement
          Si NoeudRPS_BDCarto = 'petit rond-point'
              Crée Couple_Appariement
              Affecte NoeudRPS_BDCarto au Couple_Appariement
              Qualifie confianceAppariement = « cardinalitéGeo anormal et lien
              non conforme »
          FinSi
       FinSi
    FinTant que
FinRecherche NoeudRPS_BDCarto de type 'petit rond-point' non apparié
```

Chaque correspondance de cardinalité 1-0 est donc non conforme dans ce cas puisqu'il n'est pas possible qu'un 'petit rond-point' de la BDCarto n'ait pas d'objet homologue dans Georoute.

Cet algorithme clôture ainsi la phase d'appariement proprement dite. Il faut noter que les couples ont été caractérisés à l'aide de plusieurs attributs (cardinalité du lien, nature de l'objet BDCarto, nature de l'objet Géoroute…) en plus des attributs *critèreAppariement* et *confianceAppariement* instanciés au fur et à mesure du calcul des liens.

Algorithme défini par [Devogele 1997] utilisé pour valider les liens d'appariement calculés avec les méthode précédente

Le processus d'appariement de [Devogele 1997] a été développé dans une optique d'intégration du routier des bases Géoroute et BDCarto. Géoroute est la base de comparaison, la BDCarto étant la base de référence (du point de vue de l'appariement). L'algorithme s'applique sur les objets des classes *Route*, *Tronçon Routier* et *Nœud Routier*. Il n'exploite que des éléments linéaires et ponctuels. La notion de rond-point n'existe qu'à travers la géométrie. Quatre étapes principales sont réalisées successivement, à la suite de l'appariement des objets *Route* (agrégat des tronçons) [Devogele 1997, Mustière 2002] :

- *Pré-appariement des nœuds ;*
- *Pré-appariement des tronçons ;*
- *Validation de l'appariement des nœuds ;*
- *Validation de l'appariement des tronçons ;*

Le *pré-appariement des nœuds* consiste à rechercher pour chaque nœud de la BDCarto, tous les candidats potentiels à l'appariement dans Géoroute. Cette recherche est effectuée sur la base d'un critère de proximité (distance euclidienne), qui peut varier en fonction de la sémantique du nœud de la BDCarto ('carrefour simple', 'changement d'attribut',…). S'il n'y a pas de candidat, le nœud n'est pas apparié. Dans le cas contraire, le processus se poursuit.

Le *pré-appariement des tronçons* se fait de manière similaire, sur la base d'un critère de distance mais cette fois, il s'agit de la distance de Hausdorff (cf. chapitre C.4.3.). Pour chaque tronçon communicant d'un nœud de la BDCarto, on recherche les tronçons homologues dans Géoroute.

Le pré-appariement des nœuds et des tronçons a donc pour objectif d'associer un ensemble de candidats aux éléments de la base de référence (on réduit l'espace de recherche). Les deux étapes de validation qui suivent s'attachent à filtrer les candidats pour ne retenir que le meilleur (ou les meilleurs dans le cas de regroupements connexes).

Ainsi, lors de la *validation de l'appariement des nœuds*, chaque nœud candidat de la BD Géoroute est qualifié. On dit que le nœud est *complet* si chaque tronçon communicant du nœud de la BDCarto correspondant s'apparie au moins avec un tronçon communicant de Géoroute. Dans le cas contraire, le nœud est dit *incomplet*. On considère comme normal qu'un tronçon de Géoroute n'ait pas nécessairement de correspondant dans la BDCarto car le contenu de Géoroute est plus détaillé. Les correspondances entre les tronçons tiennent compte des sens de circulation (pour la BDCarto, en pratique, il s'agit quasiment toujours d'un double sens). Plusieurs situations sont possibles à l'issue de cette qualification :

- Soit un seul nœud est complet parmi les candidats de Géoroute. Celui-ci est dès lors sélectionné et l'appariement du nœud est validé automatiquement ;
- Soit il existe un seul nœud incomplet. Dans ce cas, l'appariement doit être validé interactivement ;
- Soit plusieurs nœuds sont incomplets et dans ce cas, l'algorithme va rechercher d'éventuels carrefours complexes (appariements 1-n).

Les carrefours complexes désignent ici un ensemble d'éléments connexes composés de nœuds et de tronçons (un rond-point par exemple). Il ne s'agit pas

d'objets de la classe *Carrefour Complexe* de Géoroute. Ceux-ci ne sont pas pris en compte dans le processus.

La recherche des carrefours complexes se déroule de la manière suivante (figure 137). Pour chaque nœud de la BDCarto non encore apparié, tous les nœuds candidats incomplets correspondant de Géoroute sont sélectionnés. On associe également les tronçons de route pré-appariés qui relient ces nœuds candidats. De cette manière, on obtient un ensemble de groupes connexes. Les relations topologiques sont donc exploitées. Ces groupes sont ensuite considérés comme des « hyper-nœuds » (un groupe est vu comme un nœud) et on applique le même raisonnement que celui réalisé auparavant sur les nœuds pour qualifier les groupes. Un groupe peut ainsi être *complet*. Cela signifie que chaque tronçon communicant du nœud BDCarto s'apparie avec au moins un tronçon constitutif du groupe ou un tronçon reliant un nœud du groupe. Dans ce cas, l'appariement du nœud BDCarto avec le groupe est validé (lien 1-n). Un nœud peut également être *incomplet* voire *impossible* (aucun tronçon ne s'apparie). L'appariement doit alors être effectué ou validé interactivement. Finalement, une étape de nettoyage du groupe retenu est réalisée de manière à ne retenir que les éléments du groupe constituant un carrefour complexe. Les impasses dans le groupe sont ainsi filtrées de même que les chemins parallèles. Ce filtrage clôture la phase de validation de l'appariement des nœuds.

En ce qui concerne la validation des liens entre les tronçons, elle est principalement caractérisée par le filtrage de certains candidats à l'appariement. Ce filtrage est notamment fondé sur des critères de plus court chemin qui permettent d'éliminer des routes parallèles ou des impasses à ne pas apparier. Ces plus courts chemins sont calculés en tenant compte des sens de circulation ce qui permet de conserver toutes les chaussées lorsque celles-ci sont dédoublées.

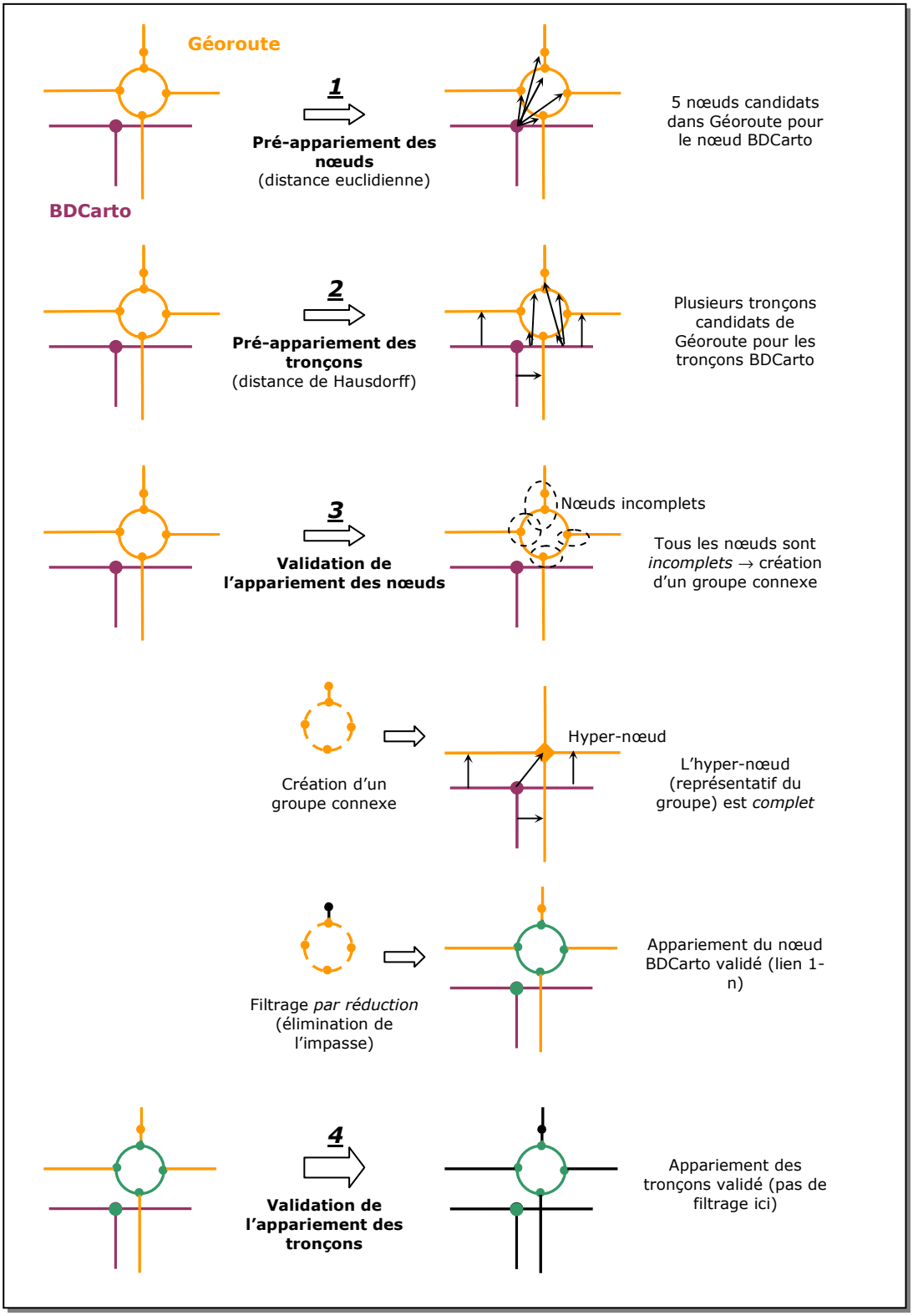


***Figure 137**. Méthode d'appariement d'un nœud BDCarto et d'un rond-point Géoroute dans le processus de [Devogele 1997].*

ANNEXE 4

Extrait d'exemples d'apprentissage utilisés dans le cadre de l'application sur les ronds-points (cf. E.3.7.2.). Ils sont destinés à prédire les conditions que doivent respecter les représentations de la BDCarto.

Exemples d'apprentissage décrits dans le format ARFF du logiciel WEKA.

```
@relation GtoC

@attribute Nature_Obj_Georoute {point, surface}
@attribute Diametre_Georoute real
@attribute Nature_Obj_carto {Carrefour_simple, Petit_rondpoint, Grand_rondpoint}

@data
point,0.000000000000000,Carrefour_simple
surface,62.769419305900900,Carrefour_simple
point,0.000000000000000,Carrefour_simple
point,0.000000000000000,Carrefour_simple
point,0.000000000000000,Carrefour_simple
surface,49.396356140913900,Carrefour_simple
point,0.000000000000000,Carrefour_simple
point,0.000000000000000,Carrefour_simple
surface,93.171884171138200,Carrefour_simple
point,0.000000000000000,Carrefour_simple
point,0.000000000000000,Carrefour_simple
point,0.000000000000000,Carrefour_simple
point,0.000000000000000,Carrefour_simple
surface,32.449961479175900,Carrefour_simple
point,0.000000000000000,Carrefour_simple
point,0.000000000000000,Carrefour_simple
point,0.000000000000000,Carrefour_simple
surface,51.855568649856700,Carrefour_simple
point,0.000000000000000,Carrefour_simple
surface,59.000000000000000,Carrefour_simple
point,0.000000000000000,Carrefour_simple
surface,62.681735776859300,Carrefour_simple
point,0.000000000000000,Carrefour_simple
point,0.000000000000000,Carrefour_simple
point,0.000000000000000,Carrefour_simple
point,0.000000000000000,Carrefour_simple
surface,53.009433122794300,Carrefour_simple
point,0.000000000000000,Carrefour_simple
point,0.000000000000000,Carrefour_simple
surface,31.064449134018100,Carrefour_simple
point,0.000000000000000,Carrefour_simple
point,0.000000000000000,Carrefour_simple
surface,40.249223594996200,Carrefour_simple
point,0.000000000000000,Carrefour_simple
point,0.000000000000000,Carrefour_simple
surface,52.430906915673300,Carrefour_simple
point,0.000000000000000,Carrefour_simple
point,0.000000000000000,Carrefour_simple
point,0.000000000000000,Carrefour_simple
surface,55.946402922797500,Carrefour_simple
point,0.000000000000000,Carrefour_simple
point,0.000000000000000,Carrefour_simple
surface,47.169905660283000,Carrefour_simple
surface,67.357256476195600,Carrefour_simple
point,0.000000000000000,Carrefour_simple
point,0.000000000000000,Carrefour_simple
point,0.000000000000000,Carrefour_simple
point,0.000000000000000,Carrefour_simple
point,0.000000000000000,Carrefour_simple
surface,14.866068747318500,Carrefour_simple
surface,16.401219466856700,Carrefour_simple
surface,25.079872407968900,Carrefour_simple
surface,31.144823004794900,Carrefour_simple
surface,33.241540277189300,Carrefour_simple
surface,34.205262752974100,Carrefour_simple
surface,35.468295701936400,Carrefour_simple
surface,38.209946349085600,Carrefour_simple
surface,40.447496832313400,Carrefour_simple
surface,40.459856648287800,Carrefour_simple
surface,40.804411526206300,Carrefour_simple
surface,41.231056256176600,Carrefour_simple
surface,42.520583250938600,Carrefour_simple
surface,43.863424398922600,Carrefour_simple
surface,44.011362169330800,Carrefour_simple
surface,44.598206241955500,Carrefour_simple
surface,49.203658400570200,Carrefour_simple
surface,50.931326312987400,Carrefour_simple
surface,51.107729356722600,Carrefour_simple
surface,52.924474489597000,Carrefour_simple
surface,56.293871780150300,Carrefour_simple
surface,57.428216061444900,Carrefour_simple
```

surface,60.440052945046300,Carrefour_simple
surface,64.404968752418500,Carrefour_simple
surface,69.462219947249000,Carrefour_simple
surface,70.880180586677400,Carrefour_simple
surface,72.339477465627300,Carrefour_simple
surface,137.058381721075000,Carrefour_simple
surface,88.102213366067000,Grand_rondpoint
surface,88.526832090615300,Grand_rondpoint
surface,195.133287780430000,Grand_rondpoint
surface,122.560189294893000,Grand_rondpoint
point,0.000000000000000,Grand_rondpoint
surface,96.166522241370500,Grand_rondpoint
surface,65.946948375190200,Grand_rondpoint
surface,96.208107766445600,Grand_rondpoint
surface,101.533245786787000,Grand_rondpoint
surface,53.907327887774200,Grand_rondpoint
surface,77.006493232713800,Grand_rondpoint
surface,87.097646351666700,Grand_rondpoint
surface,91.547801721286600,Grand_rondpoint
surface,97.948966303887000,Grand_rondpoint
surface,99.035347225119600,Grand_rondpoint
surface,108.226614102078000,Grand_rondpoint
point,0.000000000000000,Petit_rondpoint
point,0.000000000000000,Petit_rondpoint
point,0.000000000000000,Petit_rondpoint
surface,64.443773942872100,Petit_rondpoint
surface,84.899941107164500,Petit_rondpoint
surface,75.960516059331800,Petit_rondpoint
surface,64.899922958351800,Petit_rondpoint
surface,68.600291544570000,Petit_rondpoint
surface,56.612719418872600,Petit_rondpoint
surface,71.063352017759500,Petit_rondpoint
point,0.000000000000000,Petit_rondpoint
surface,56.080299571239800,Petit_rondpoint
point,0.000000000000000,Petit_rondpoint
point,0.000000000000000,Petit_rondpoint
point,0.000000000000000,Petit_rondpoint
point,0.000000000000000,Petit_rondpoint
surface,42.059481689626200,Petit_rondpoint
surface,81.271151093115400,Petit_rondpoint
surface,53.413481444294600,Petit_rondpoint
surface,64.381674411279500,Petit_rondpoint
point,0.000000000000000,Petit_rondpoint
point,0.000000000000000,Petit_rondpoint
surface,48.414873747640800,Petit_rondpoint
surface,56.044625076808200,Petit_rondpoint
point,0.000000000000000,Petit_rondpoint
surface,35.341194094144600,Petit_rondpoint
surface,62.128898268036300,Petit_rondpoint
surface,65.000000000000000,Petit_rondpoint
surface,81.271151093115400,Petit_rondpoint
surface,59.481089431852200,Petit_rondpoint
surface,58.309518948453000,Petit_rondpoint
surface,95.189285111298100,Petit_rondpoint
surface,77.524189773257200,Petit_rondpoint
surface,74.726166769077600,Petit_rondpoint
point,0.000000000000000,Petit_rondpoint
point,0.000000000000000,Petit_rondpoint
point,0.000000000000000,Petit_rondpoint
point,0.000000000000000,Petit_rondpoint
surface,35.510561809129400,Petit_rondpoint
surface,38.587562763149500,Petit_rondpoint
surface,39.051248379533300,Petit_rondpoint
surface,45.276925690687100,Petit_rondpoint
surface,45.967379738244800,Petit_rondpoint
surface,46.043457732885400,Petit_rondpoint
surface,48.507731342539600,Petit_rondpoint
surface,51.739733281106100,Petit_rondpoint
surface,56.400354608814300,Petit_rondpoint
surface,57.801384066473700,Petit_rondpoint
surface,58.523499553598100,Petit_rondpoint
surface,59.464274989274000,Petit_rondpoint
surface,60.835844697020500,Petit_rondpoint
surface,62.225396744416200,Petit_rondpoint
surface,65.069193939989800,Petit_rondpoint
surface,65.215028942721500,Petit_rondpoint
surface,68.680419334771100,Petit_rondpoint
surface,68.876701430890300,Petit_rondpoint
surface,70.576199954375600,Petit_rondpoint
surface,73.573092907665600,Petit_rondpoint
surface,76.059187479225700,Petit_rondpoint
surface,79.511005527536900,Petit_rondpoint
surface,83.725742755737900,Petit_rondpoint
surface,91.547801721286600,Petit_rondpoint
surface,95.441081301502500,Petit_rondpoint
surface,96.254870006665100,Petit_rondpoint
surface,97.185389848474700,Petit_rondpoint
surface,113.951744172698000,Petit_rondpoint

Methodology for assessing consistency between multiple representations for spatial databases integration

An approach combining the use of metadata and machine learning

Nowadays most databases are run independently. An independence that leads to a series of problems: repeated efforts of maintenance and updating, difficulty in proceeding with an analysis at various levels and no guarantee of coherence between sources.

Joint management of these sources requires them to be integrated in order to define the explicit links between the various bases and to provide a unified vision. Our thesis deals with this issue. It concentrates in particular on the means of relating data and of assessing coherence between multiple representations. We have sought to systematically analyse each difference in representation between matching data so as to determine whether it results from different criteria used for data capture or from errors in the capture itself, the aim being to ensure coherent data integration.

In order to study the conformity of representations, we suggest exploiting existing database specifications. These documents describe specific selection and modelling rules for objects. They are reference metadata used to determine whether representations are equivalent or incoherent. But their use is insufficient since specifications described in a natural language can be imprecise or incomplete. So the data contained in the bases is a second interesting source of knowledge. If one uses machine learning techniques to analyse how they tally, it becomes possible to establish evaluation rules that enable a justification of the conformity of representations.

The methodology we put forward is based upon these elements. It consists in a coherence evaluation process and a knowledge acquisition proceeding. The process comprises several steps: data enrichment, intra-base control, matching, inter-bases control, and the final assessment. Each of these steps exploits knowledge inferred from the specifications or induced from the data through learning. The benefit of using machine learning techniques is twofold: not only does it enable to acquire evaluation rules, it also reveals the discrepancy tolerated in the data when compared to the written specifications.

This approach has been carried out on NGI databases that showed different levels of detail.

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

A l'heure actuelle, la plupart des bases de données spatiales sont gérées de manière indépendante. Cette indépendance pose différents problèmes : elle multiplie les efforts de maintenance et de mise à jour, elle rend difficile la mise en œuvre d'analyses multi-niveaux et ne garantit pas une cohérence entre les sources.

Une gestion conjointe de ces sources nécessite leur intégration qui permet de définir des liens explicites entre les bases et d'en fournir une vision unifiée. Notre thèse s'inscrit dans ce cadre. Le sujet que nous traitons porte en particulier sur la mise en correspondance des données et l'évaluation de la cohérence inter-représentations. Nous cherchons à analyser automatiquement chaque différence de représentation entre les données appariées afin d'en déduire si celle-ci résulte des critères de saisie différents des bases ou d'erreurs de saisie. Cette évaluation vise à garantir une intégration cohérente des données.

Pour étudier la conformité des représentations nous proposons d'exploiter les spécifications des bases. Ces documents décrivent les règles de sélection et de modélisation des objets. Ils constituent des métadonnées de référence pour juger si les représentations sont équivalentes ou incohérentes. L'utilisation de ces documents est toutefois insuffisante. Les spécifications décrites en langue naturelle peuvent être imprécises ou incomplètes. Dans ce contexte, les données des bases constituent une seconde source de connaissances intéressante. L'analyse des correspondances à l'aide de techniques d'apprentissage automatique permet d'induire des règles rendant possible la justification de la conformité des représentations.

La méthodologie que nous proposons repose sur ces éléments. Elle se compose de deux méthodes : *MECO* et *MACO*. La première est la Méthode d'Evaluation de la COhérence. Elle comprend plusieurs étapes : l'enrichissement des données, le contrôle intra-base, l'appariement, le contrôle inter-bases et l'évaluation finale. Chacune de ces étapes exploite des connaissances déduites des spécifications ou induites des données par apprentissage automatique, en appliquant *MACO* (Méthode d'Acquisition de connaissances pour l'évaluation de la COhérence). L'intérêt d'utiliser l'apprentissage est double. Outre le fait qu'il permet d'acquérir des règles pour l'évaluation, il met en évidence l'écart toléré sur les données par rapport aux spécifications papiers.

Notre approche a été mise en œuvre sur des bases de données de l'IGN présentant différents niveaux de détail.