Intégration et optimisation des grilles régulières de points dans une architecture SOLAP relationnelle

Mehdi Zaamoune

N° d'ordre : D.U : 2538

EDSPIC : 683



Université Blaise Pascal – Clermont II

École Doctorale

Sciences Pour l'Ingénieur de Clermont-Ferrand

THÈSE

Présentée par

Mehdi Zaamoune

Pour obtenir le grade de

DOCTEUR D'UNIVERSITE

Spécialité : INFORMATIQUE

Intégration et optimisation des grilles régulières de points dans une architecture SOLAP relationnelle

---

Soutenue publiquement le *08/01/2015* devant le jury composé de :

Robert LAURINI Professeur (INSA de Lyon – Lyon)	Président
Fadila BENTAYEB Maître de Conférences (Université Lumière Lyon 2 - Lyon)	Rapporteur
Franck RAVAT Professeur des Universités (Université Toulouse 1 Capitole -Toulouse)	Rapporteur
Richard CHBEIR Professeur (IUT de Bayonne - Bayonne)	Examinateur
Sandro BIMONTE Chargé de recherche (Irstea, Clermont-Fd)	Encadrant
Myoung-Ah KANG Maître de Conférences (Université Blaise Pascal, Clermont-Fd)	Encadrant
François PINET Directeur de Recherche (Irstea, Clermont-Fd)	Directeur de thèse

Résumé

Les champs continus sont des types de représentations spatiales utilisées pour modéliser des phénomènes tels que la température, la pollution ou l'altitude. Ils sont définis selon une fonction de mapping f qui affecte une valeur du phénomène étudié à chaque localisation p du domaine d'étude. Par ailleurs, la représentation des champs continus à différentes échelles ou résolutions est souvent essentielle pour une analyse spatiale efficace. L'avantage des champs continus réside dans le niveau de détails généré par la continuité, ainsi que la qualité de l'analyse spatiale fournie par la multi-résolution. L'inconvénient de ce type de représentations dans l'analyse spatio-multidimensionnelle est le coût des performances d'analyse et de stockage.

Par ailleurs, les entrepôts de données spatiaux et les systèmes OLAP spatiaux (EDS et SOLAP) sont des systèmes d'aide à la décision qui permettent l'analyse spatio-multidimensionnelle de grands volumes de données spatiales et non spatiales. L'analyse des champs continus dans l'architecture SOLAP représente un défi de recherche intéressant. Différents travaux se sont intéressés à l'intégration de ce type de représentations dans le système SOLAP. Cependant, celle-ci est toujours au stade embryonnaire.

Cette thèse s'intéresse à l'intégration des champs continus incomplets représentés par une grille régulière de points dans l'analyse spatio-multidimensionnelle. Cette intégration dans le système SOLAP implique que l'analyse des champs continus doit supporter :

(i) les opérateurs OLAP classiques, (ii) la vue continue des données spatiales, (iii) les opérateurs spatiaux (slice spatial) et (iv) l'interrogation des données à différentes résolutions prédéfinies.

Dans cette thèse nous proposons différentes approches pour l'analyse des champs continus dans le SOLAP à différents niveaux de l'architecture relationnelle, de la modélisation conceptuelle à l'optimisation des performances de calcul. Nous proposons un modèle logique FISS qui permet d'optimiser les performances d'analyse à multi-résolution en se basant sur des méthodes d'interpolation. Puis, nous exposons une méthodologie basée sur la méthode d'échantillonnage du Clustering, qui permet d'optimiser les opérations d'agrégation des grilles régulières de points dans l'architecture SOLAP relationnelle en effectuant une estimation des résultats.

Remerciements

Cette section est sans aucun doute, pour moi, la section la plus dure à rédiger, car je crains de ne pas pouvoir remercier, comme il se doit, toutes les personnes qui se sont impliquées dans cette thèse et qui m'ont permis de la mener à son terme.

Je tiens à remercier très sincèrement M. François Pinet, directeur de recherche – HDR à Irstea, Centre de Clermont-Ferrand, pour m'avoir proposé cette thèse et pour l'avoir dirigée avec tant de soutien et d'implication. Sa présence permanente et ses compétences de directeur de recherche m'ont permis de me focaliser sur mes objectifs et de mener à bien les différentes étapes qui ont accompagné la réalisation de cette thèse.

Je tiens aussi à remercier, M. Emmanuel Hugo et M. Jean-Pierre Chanet pour m'avoir accueilli au sein de l'unité TSCF et m'avoir offert un cadre de travail idéal et prospère me permettant, dès le début de ma thèse, d'intégrer facilement l'équipe et de me sentir un membre à part entière d'Irstea.

Je tiens à remercier très sincèrement M. Sandro Bimonte, chargé de recherche à Irstea, Centre de Clermont-Ferrand, pour avoir suivi et encadré cette thèse. J'ai eu l'immense honneur de travailler sous son encadrement au cours de cette thèse. Son implication permanente, sa disponibilité, ses conseils pertinents et nos échanges fréquents, ont été la clé de l'aboutissement de cette thèse.

Je tiens aussi à remercier Mme Myoung-Ah Kang, maitre de conférences à ISIMA, pour ses nombreux conseils et son implication. Je voudrais lui exprimer ma reconnaissance pour les efforts inépuisables qu'elle a fournis, pour sa patience et sa disponibilité malgré ses occupations.

Je remercie sincèrement M. Franck Ravat, professeur à l'IRIT-Université de Toulouse Capitole et Mme Fadila Bentayeb, HDR à l'université Lyon 2, pour leurs valeureux conseils et leurs remarques pertinentes qui ont contribué à l'amélioration de ce mémoire de thèse, ainsi que pour l'honneur qu'ils me font en acceptant de faire partie du jury.

Je tiens à remercier très sincèrement M. Richard Chbeir, professeur à l'IUT de Bayonne et M. Robert Laurini, professeur à l'INSA de Lyon, pour avoir accepté d'être les examinateurs de cette thèse.

Mes remerciements s'adressent également à tous les membres d'Irstea et plus particulièrement aux membres de l'équipe COPAIN, qui ont toujours montré un intérêt pour mes travaux et qui m'ont toujours soutenu avec leurs conseils et leurs encouragements jusqu'à l'aboutissement de cette thèse. Je remercie plus particulièrement Catherine Roussey, Gil De Sousa et Jean-Pierre Chanet pour leurs remarques, conseils et critiques constructives qui m'ont permis de m'améliorer tout au long de cette thèse.

Je remercie très sincèrement l'Irstea, le conseil régional d'auvergne, le FEDER, ainsi que la société Agaetis, pour avoir permis à cette thèse de voir le jour et pour leurs financements. Je tiens à remercier plus particulièrement M. Philippe Beaune pour son aide précieuse et sa disponibilité.

Je voudrais remercier les personnes qui ont eu un impact direct et indirect sur l'aboutissement de cette thèse et qui sont mes chers parents, mon frère FAHD, ma grand-mère et mes oncles. Je remercie aussi mes amis en France comme au Maroc pour leur soutien et leurs encouragements.

Finalement, je voudrais exprimer ma grande reconnaissance envers ma mère qui a fait preuve d'un soutien et d'une confiance inébranlables, ce qui m'a permis de faire de mon rêve une réalité, en lui dédiant cette thèse.

Table des matières

Liste des figures

Liste des Tables

Partie I : Introduction

Chapitre 1. Introduction

1.1 Contexte de thèse

L'information géographique peut être représentée selon deux modèles, en fonction de la nature des données : *les objets discrets* (vecteur) et *les champs continus* (Tomlin 1990). Les objets discrets sont utilisés pour représenter l'information géographique selon le modèle vectoriel. La représentation discrète considère que l'espace est composé d'objets dont les frontières sont bien définies (ex. parcelle, ville, etc.). Les champs continus (également appelés "données spatiales continues" ou "Continuous fields" en anglais) représentent des phénomènes physiques qui changent continuellement dans l'espace (Paolino et al. 2010), par exemple la température, la densité de la couverture végétale, etc. (Mozgeris 2009).

Les Entrepôts de Données Spatiales (EDSs) et le système OLAP Spatial (SOLAP) jouent un rôle important en aidant les décideurs à obtenir le maximum d'avantages des grandes quantités de données spatiales (Bédard, HAN et Merrett 2001). Ces technologies étendent les entrepôts de données (EDs) et les systèmes OLAP pour intégrer les données spatiales en addition aux données classiques entreposées, afin d'effectuer l'analyse en ligne de grands volumes de données géo-référencées.

Les EDSs sont modélisés en accord avec le modèle spatio-multidimensionnel qui étend le modèle multidimensionnel afin de définir le cube spatial (c.-à-d. un cube de données composé au moins d'une dimension spatiale ou d'une mesure spatiale), les dimensions spatiales (c.-à-d. les axes d'analyse) et les mesures spatiales (c.-à-d. les sujets d'analyse) qui intègrent les informations géographiques en utilisant le modèle vectoriel.

Le SOLAP a été défini par Yvan Bédard comme « *Une plateforme visuelle conçue spécialement pour supporter une analyse spatio-temporelle rapide et efficace à travers une approche multidimensionnelle qui comprend des niveaux d'agrégation cartographiques, graphiques et tabulaires* » (Bédard 1997). Les systèmes SOLAP intègrent les capacités du système OLAP et celles des systèmes d'information géographique (SIG) dans un cadre unique généralement basé sur le stockage relationnel des données spatiales selon le modèle vectoriel. Le SOLAP implémente un ensemble d'opérateurs pour l'exploration en ligne des données spatiales. Ces opérateurs permettent de modifier le niveau de granularité des mesures analysées en utilisant des fonctions d'agrégation et de forage (ex. Roll-up spatial, Drill-down

spatial). Ils permettent aussi d'effectuer des opérations de coupe sur les données spatiales (ex. slice spatial) afin de sélectionner un sous-ensemble des données spatiales du cube.

L'architecture illustrée dans la figure 1.1, représente l'architecture SOLAP relationnelle basée sur la solution OLAP-SIG hybride utilisée dans cette thèse. Cette architecture est composée de sources de données, d'un ETL (Extract, Transform & Load), d'un entrepôt de données spatiales implémenté dans le SGBDS PostgreSQL étendu avec PostGIS, d'un serveur SOLAP implémenté avec GeoMondrian et d'un client SOLAP qui fournit une visualisation tabulaire (JPivot) et cartographique (OpenLayers) des résultats d'analyse.

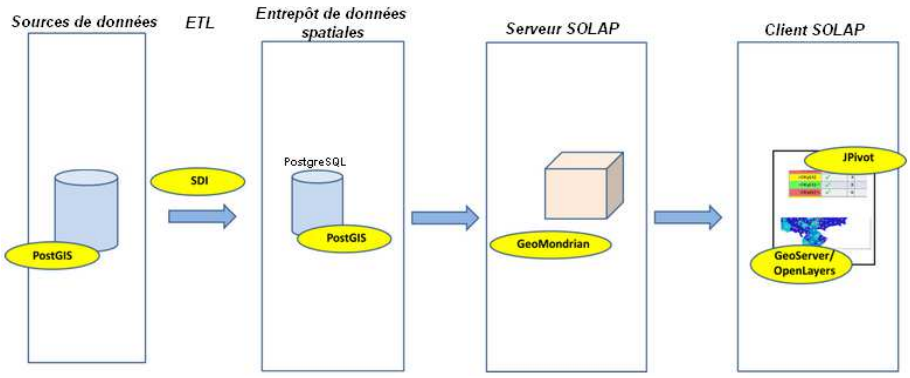


Fig 1. 1 L'architecture SOLAP relationnelle utilisée

Par ailleurs, les représentations continues (ex. température, altitude, etc.) posent des problèmes d'implémentation dans les systèmes informatiques existants, notamment à cause des valeurs infinies qu'elles représentent. Deux catégories de représentations des données spatiales continues ont été proposées : la *représentation incomplète* et la *représentation complète* (Goodchild 1993). Les *représentations incomplètes* associent des valeurs à un échantillon de points et ont besoin de fonctions supplémentaires pour calculer les valeurs dans les zones non échantillonnées (ex. grille de points). Les *représentations complètes* associent des valeurs à des régions et supposent que ces valeurs sont valables pour chaque point dans ces régions (ex. raster).

Selon le type d'analyses à effectuer, la représentation incomplète (ex. grille de points) a l'avantage de réduire le cout du stockage, en permettant d'entreposer un échantillon de points, puis d'utiliser des fonctions supplémentaires, tels que les fonctions d'interpolation, pour améliorer le niveau de détails selon le type d'analyses à effectuer.

Des opérateurs d'analyse ad-hoc (*map algebra*), différents des opérateurs topologiques utilisés dans le modèle vectoriel, ont été définis pour les champs continus. Ils permettent une

analyse (ex. agrégation) point par point des champs continus complets (Tomlin 1990) et incomplets (Ledoux 2008).

Aussi, la représentation des champs continus à différentes échelles ou résolutions est souvent souhaitable pour une analyse efficace des phénomènes spatiaux complexes. Ces résolutions ou échelles représentent les besoins d'analyse des décideurs et doivent être explicitement représentés dans les données et dans le modèle de requêtes.

Plusieurs travaux ce sont intéressés à l'intégration des champs continus dans le SOLAP (Vaisman et Zimányi 2009) (Ahmed et Buras 2009) (L. Gómez, Vaisman et Zimányi 2010). En effet, l'intégration des champs continus dans l'analyse en ligne SOLAP représente un atout majeur pour l'analyse des phénomènes spatio-temporels continus. Leur modélisation dans les entrepôts de données spatiales, ainsi que l'adaptation des fonctionnalités SOLAP pour gérer leur analyse continue et à multi-résolution sont des problématiques ouvertes.

1.2 Problématique

Les modèles SOLAP existants se focalisent généralement sur la représentation complète du champ continu (ex. raster). Ce type de représentation modélise la continuité en se basant sur un découpage régulier ou irrégulier de l'espace d'étude et en considérant que toutes les localisations dans chaque sous-espace ont la même valeur.

L'application sur laquelle se basent nos recherches et nos expérimentations a été définie en collaboration avec la société Agaetis. Agaetis est une société spécialisée dans l'analyse de données et le développement logiciel (« Agaetis » 2014). Cette application concerne l'analyse des valeurs de la pollution (odeur) à différentes localisations (capteurs) dans le territoire français, selon différentes dimensions (cf. section 4.2.1). L'ensemble de ces localisations constitue un nuage de points qui est transformé grâce au modèle de simulation *ADMS5* en une grille régulière de points. Cette grille de points représente l'axe spatial de l'analyse de la pollution.

Ainsi, *dans cette thèse, nous nous focalisons sur ce type de représentations, qui sont les champs continus incomplets représentés par des grilles régulières de points.*

L'intégration des champs continus à multi-résolution dans le SOLAP présente encore des défis, tant par sa modélisation particulière que par le grand volume de données que les champs continus comportent. En effet, la modélisation d'un champ continu à multi-résolution dans le SOLAP doit supporter : *(i) les opérateurs OLAP classiques, (ii) la vue continue des*

données spatiales, (iii) les opérateurs spatiaux (slice spatial) et (iv) l'interrogation des données à différentes résolutions prédéfinies.

Toutefois, au-delà de la modélisation des champs continus et de leur exploitation dans le SOLAP, les performances de calculs des requêtes représentent une contrainte essentielle pour une analyse spatio-multidimensionnelle en ligne. En effet, l'analyse à multi-résolution des champs continus tel qu'effectuée classiquement implique le stockage complet des valeurs au niveau de résolution le plus détaillé prévu pour l'analyse (McHugh 2008). Ceci peut engendrer de nombreuses jointures dans l'ED pour retrouver les faits analysés, ce qui affecte les performances de calculs et de stockage.

Si nous prenons pour exemple un ED de données spatiales, composé d'une dimension spatiale (ex. localisation) qui représente les points d'une grille régulière de résolution 500*500 (250.000 points), et d'une dimension temporelle (ex. temps), avec une granularité journalière, qui est constituée de 365 jours. L'extraction des mesures de chaque couple (point/jour) sur toute la grille et toute l'année, implique 91.250.000 jointures dans la table de faits du cube.

Plusieurs travaux se focalisent sur l'utilisation des méthodes d'approximation et notamment des techniques d'échantillonnage, dans le calcul des requêtes multidimensionnelles (Han 1997) (Jin et al. 2006) (Cao 2013). Ces travaux visent à améliorer les performances des requêtes en utilisant un échantillon de données dans les calculs. Ces méthodes sont souvent utilisées dans l'OLAP pour compresser les données et créer un cube de données compact (Das 2009) (Feng et Wang 2002). Ceci réduit le nombre de données à un échantillon de celles-ci ce qui minimise le nombre de jointures requises lors des opérations d'agrégation par exemple. Cependant, la perte des données originales dans le processus de compression peut limiter l'analyse si l'utilisateur a besoin de comparer les résultats estimés et les résultats réels, ou bien si l'utilisateur a besoin d'effectuer certaines analyses qui ne supportent pas une estimation. De plus, la modélisation d'un cube compressé dans une architecture relationnelle SOLAP présente des problématiques liées à la modélisation des échantillons dans le cube, l'adaptation des opérateurs d'agrégation (ex. Roll-up) pour l'estimation des résultats, et l'accès aux données originales.

1.3 Contributions

Dans cette thèse, notre principale contribution est **l'extension de l'architecture SOLAP relationnelle afin de traiter les *champs continus incomplets représentés par des grilles régulières de points à multi-résolution*** (cf. chapitre 4)**.**

Nous avons traité cette problématique du niveau conceptuel au niveau physique (calcul des performances et optimisation), en passant par le niveau logique.

Pour permettre une analyse des phénomènes représentés par des grilles régulières de points dans l'architecture représentée dans la figure 1.1, nous avons proposé les contributions suivantes en nous appuyant sur la démarche présentée dans la table 1.1:

(A) Proposition d'un modèle UML conceptuel pour la définition d'une dimension spatiale de type grille régulière à plusieurs niveaux de résolutions, ainsi que ses opérateurs de continuité et de multi-résolution, grâce à l'extension du profil UML « CI SOLAP ».

(B) Proposition d'un modèle logique basé sur la modélisation en étoile, pour la gestion des grilles régulières à multi-résolution dans l'EDS. Le modèle logique que nous proposons (*FISS*), adapte le schéma en étoile classique pour que la table de faits soit reliée au niveau le moins détaillé de la dimension spatiale. Ceci améliore considérablement les performances de calcul puisque la table de faits contient moins de données (seulement celles qui correspondent au niveau de résolution le plus grossier). Les valeurs qui correspondent aux niveaux de résolutions élevées sont calculées dynamiquement selon chaque requête.

(C) Proposition d'opérateurs de continuité et de multi-résolution (FieldMDX) basés sur des fonctions d'interpolation pour l'analyse des grille régulières continues et à multi-résolution dans le SOLAP.

(D) Nous proposons aussi des expérimentations sur l'utilisation de nos modèles et de nos opérateurs dans une architecture SOLAP fonctionnelle qui utilise PostreSQL-PostGIS comme SGBDS et GeoMondrian comme serveur SOLAP.

Notre seconde contribution concerne **l'optimisation des requêtes d'agrégation dans le SOLAP grâce aux méthodes d'échantillonnage** (cf. chapitre 5)**.** Cette contribution consiste à extraire des échantillons du cube de données original, puis à les intégrer à un cube de données plus compact, qui est utilisé pour répondre aux requêtes d'agrégation jugées

couteuses pour le cube original. La méthodologie que nous proposons est aussi basée sur la démarche présentée dans la table 1.1. Elle consiste en :

(A) La proposition d'un méta-modèle pour la définition des concepts du cube compact « ClusterCube ».

(B) Un mapping entre les entités (membres de dimensions, faits, …) du cube original et celles du « ClusterCube ».

(C) La proposition d'un modèle logique et la mise en œuvre du « ClusterCube » dans une architecture SOLAP relationnelle.

(D) Proposition d'une approche (*ClusterCube&Field*) qui combine les capacités d'analyse des grilles régulières à multi-résolution (*FISS*) et l'optimisation fournie par le « ClusterCube » dans le même modèle multidimensionnel.

Ces différentes propositions ont été réalisées et appliquées pour le cas d'étude fournit par la société Agaetis.

1.4 Motivations et Méthodologie de recherche

Nos travaux de recherche se focalisent sur l'intégration des champs continus incomplets représentés par des grilles régulières de points à multi-résolution dans une architecture SOLAP relationnelle en vue d'une analyse spatio-multidimensionnelle.

Cette intégration doit satisfaire les exigences suivantes :

(1) Permettre d'utiliser les opérateurs OLAP classiques (ex. Roll-up, slice),

(2) Permettre d'utiliser les opérateurs SOLAP (ex. opérateurs topologiques, Map Algebra),

(3) Fournir une vue continue des données pendant l'analyse spatiale,

(4) Permettre d'explorer les données spatiales continues à différents niveaux de résolution,

(5) Conserver des performances adéquates pour une analyse SOLAP pertinente.

Comme le montre la table 1.1, nous nous sommes focalisés sur la représentation par grille régulière de points dans le SOLAP et nous avons étudié sa modélisation et son intégration à différents niveaux : (i) *modélisation conceptuelle,* (ii) *modélisation logique,* (iii) *performances de calcul.*

De la même manière que pour les données spatiales vectorielles, nous considérons que l'intégration des grilles régulières de points dans le SOLAP doit être modélisée puis réalisée

aux différents tiers qui constituent l'architecture OLAP-SIG hybride (EDS et serveur SOLAP). Ceci permet d'exploiter ce type de représentations spatiales dans tous le processus d'analyse SOLAP, de l'entreposage au requêtage.

	Objets discrets	**Champs continus**	
	Vectoriel	**Grille de points**	Raster, TIN, Diagrammes de Voronoi, …)
Modélisation conceptuelle	X	**X**	…
Modélisation logique	X	**X**	…
Performances de calcul	X	**X**	…
Architecture OLAP-SIG Hybride	X	**X**	…

Table 1. 1 Intégration de la représentation par grille régulière des champs continus dans l'architecture OLAP-SIG

Cette démarche nous permet d'identifier les besoins d'analyse des grilles régulières de points à chaque étape de l'analyse spatio-multidimensionnelle et de proposer un ensemble d'approches et de modèles qui permettent d'exploiter ce type de données dans une architecture OLAP-SIG hybride utilisée dans la majorité des solutions SOLAP existantes.

La méthodologie de recherche appliquée est constituée des étapes suivantes :

(1) Revue de littérature **des concepts OLAP-SOLAP fondamentaux** :

Cette étape a consisté en une revue des concepts de l'analyse multidimensionnelle et spatio-multidimensionnelle dans une architecture OLAP-SIG hybride relationnelle. Le choix de cette architecture est justifié dans la section 2.2.1.4. Cette étape nous a permis de nous familiariser avec les concepts de base nécessaires pour répondre à la problématique de recherche présentée. Elle vise à comprendre les concepts de dimensions, hiérarchies, mesures, faits, ainsi que l'architecture et les outils d'analyse OLAP et SOLAP.

(2) Revue de littérature **des champs continus dans les SIG** :

Pour permettre une intégration des champs continus dans l'analyse SOLAP, nous nous sommes intéressés à la définition des champs continus, ainsi que les opérateurs utilisés dans leur analyse dans les SIG (map algebra, interpolation). Cette étape nous a permis d'identifier les caractéristiques des grilles régulières de points et leur potentiel dans l'analyse continue des phénomènes spatiaux.

(3) Etat de l'art sur **les modèles conceptuels et logiques** pour l'intégration des champs continus dans le SOLAP.

Pour pouvoir exprimer le besoin d'analyse d'un champ continu dans le SOLAP conceptuellement, puis logiquement dans l'EDS, nous avons effectué un état de l'art qui regroupe les principaux travaux de la littérature afin de les comparer selon la démarche définie dans la table 1.1. Cette comparaison a montré qu'aucun modèle ne satisfait l'ensemble des critères définis pour une analyse spatio-multidimensionnelle efficace des grilles régulières de points.

(4) Etat de l'art sur l'utilisation des techniques d'approximation dans le SOLAP pour améliorer **les performances de calcul** :

Dans cette étape, nous nous sommes intéressés aux techniques d'approximation basées sur l'échantillonnage (ex. clustering), afin de proposer une optimisation non-spatiale des requêtes SOLAP. Nous avons classifié les travaux sur l'utilisation et l'intégration des techniques d'approximation dans l'analyse multidimensionnelle selon des critères que nous avons définis dans la section 2.5.

(5) **Modélisation conceptuelle et logique** d'une grille régulière de points dans une architecture SOLAP relationnelle :

Dans cette étape, nous avons étendu le profil UML « CI SOLAP » proposé dans (Boulil 2012) afin d'y définir le concept de champ continu incomplet à plusieurs niveaux de résolution. CI SOLAP définit un ensemble de stéréotypes et de contraintes (OCL), qui regroupent les composants et les règles d'agrégation pour l'analyse des données spatiales et non-spatiales dans le SOLAP. Nous avons étendu ce profil afin de définir le champ continu, ainsi que l'opération d'interpolation à utiliser pour gérer sa continuité et sa résolution. Le choix d'utilisation du profil UML « CI SOLAP » est motivé par le fait que celui-ci dispose de tous les éléments et des contraintes nécessaires pour effectuer une modélisation spatio-multidimensionnelle complexe. Ce profil a été réalisé dans notre équipe (Boulil 2012).

Cette extension nous a aussi permis de proposer un modèle logique (*FISS*) pour l'analyse des grilles régulières à multi-résolution, basée sur des méthodes d'interpolation, dans le SOLAP.

(6) **Expérimentations et performances** de l'analyse des grilles régulières dans le SOLAP :

Cette étape a consisté en l'implémentation et le calcul des performances des requêtes sur les grilles régulières à différents niveaux de résolution. L'objectif de ces expérimentations est de

comparer l'approche classique utilisée pour changer la résolution des données matricielles (*FASS*) avec l'approche que nous proposons (*FISS*). Ainsi, nous avons appliqué l'approche proposée au cas d'étude définit par la société Agaetis.

(7) **Modélisation conceptuelle et logique** d'une méthode d'approximation (Clustering) dans le SOLAP adaptée à l'analyse des grilles régulières de points à multi-résolution :

Nous avons proposé un méta-modèle spatio-multidimensionnel qui définit les concepts du « ClusterCube ». Le ClusterCube est un cube réduit constitué d'un échantillon de données récupéré du cube original, qui permet d'utiliser des méthodes d'approximation pour estimer les requêtes d'agrégation coûteuses.

Cette étape avait pour objectif de combiner les capacités de l'approche *FISS* de l'étape 5 et les capacités d'approximation du « ClusterCube », dans un même modèle spatio-multidimensionnel « ClusterCube&Field ».

(8) **Expérimentations et performances** de l'analyse approximée des grilles régulières :

Dans cette dernière étape, nous avons implémenté les modèles proposés pour le « ClusterCube » dans une architecture SOLAP relationnelle, afin de fournir une preuve d'utilisabilité et comparer les performances obtenues entre le « ClusterCube » et le cube de données original. Ces expérimentations ont été appliquées au cas d'étude définit par la société Agaetis.

1.5 Organisation du mémoire

Les travaux présentés dans cette thèse sont regroupées en 4 parties :

Partie Etat de L'art. Cette partie présente les principaux concepts de l'analyse multidimensionnelle et spatio-multidimensionnelle des champs continus dans le SOLAP. Celle-ci est organisée en 2 chapitres :

Chapitre 2. Ce chapitre introduit les concepts des systèmes OLAP et SOLAP, ainsi que leurs architectures et leurs modélisations multidimensionnelles. Il définit aussi les concepts liés aux champs continus et leur analyse dans les SIG, puis présente un ensemble de travaux qui traitent de l'intégration de l'analyse spatiale continue dans le SOLAP (Sandro Bimonte et al. 2014).

Chapitre 3. Ce chapitre introduit les techniques d'approximation et les travaux qui se sont focalisés sur leur intégration dans le système SOLAP. Ce chapitre fournit aussi une comparaison et un bilan des différents travaux effectués.

Partie Contributions. Cette partie regroupe les contributions présentées précédemment. Elle s'articule autour de 2 chapitres.

Chapitre 4. Ce chapitre présente l'intégration du champ continu représenté par une grille régulière de points dans le modèle conceptuel SOLAP, ainsi que sa modélisation logique dans l'EDS selon deux approches *FASS* et *FISS*. La première approche (*FASS*) consiste à pré-calculer puis stocker dans la table de faits toutes les valeurs au niveau de résolution le plus détaillé, puis utiliser des opérateurs d'agrégation pour explorer la hiérarchie spatiale. L'approche que nous proposons (*FISS*) consiste à ne stocker dans la table de faits que le niveau de résolution le moins détaillé, puis utiliser des fonctions d'interpolation pour estimer les valeurs des niveaux de résolution les plus détaillés. Ceci, permet de réduire le coût du stockage et améliorer les performances de requêtage.

Ce chapitre présente aussi les opérateurs FieldMDX de continuité et de multi-résolution pour une analyse continue à plusieurs niveaux de détails spatiaux, ainsi qu'une implémentation dans une architecture SOLAP relationnelle basée sur la solution OLAP-SIG hybride. Nous présentons à la fin de ce chapitre nos expérimentations et une comparaison entre les performances obtenues avec l'approche *FASS* et l'approche *FISS*.

Ce chapitre est issu d'une version améliorée des articles :

- *Mehdi Zaamoune, Sandro Bimonte, François Pinet, Philippe Beaune,* ***A New Relational Spatial OLAP Approach for Multi-resolution and Spatio-multidimensional Analysis of Incomplete Field Data.*** Publié dans *ICEIS (2013) INSTICC International Conference on Enterprise Information Systems, July 3-7, 2013, Anger, pages 145-152.*(Zaamoune et al. 2013a)
- *Mehdi Zaamoune, Sandro Bimonte, François Pinet, Philippe Beaune,* ***Intégration des données champs continus incomplets dans l'OLAP: de la modélisation conceptuelle à l'implémentation.*** Publié dans *EDA (2013), Revue des Nouvelles Technologies de l'Information vol.RNTI-B-9, pages 33-4.*(Zaamoune et al. 2013b)
- *Mehdi Zaamoune, Sandro Bimonte, Philippe Beaune,* ***Integration of incomplete field data represented with regular grids of points in Spatial OLAP: from conceptual modeling to implementation in Relational OLAP architectures.*** Soumi à *Computers & Geosciences journal.*

Chapitre 5. L'intérêt de ce chapitre est de proposer une modélisation SOLAP qui intègre l'utilisation des méthodes d'échantillonnage pour l'optimisation de l'analyse des grilles régulières de points à résolutions élevées.

Dans ce chapitre, nous nous basons sur la technique d'échantillonnage du *cluster sampling* (clustering), afin de proposer un méta-modèle SOLAP qui permet d'exploiter les avantages de l'échantillonnage dans l'analyse.

Nous définissons un mapping entre le cube des données originales et le cube des données échantillonnées, puis nous proposons un modèle conceptuel et un modèle logique du ClusterCube, ainsi qu'une méthodologie de calcul des opérations d'agrégation en utilisant les échantillons de données.

Nous proposons aussi une implémentation du ClusterCube dans une architecture SOLAP relationnelle basée sur la solution OLAP-GIS hybride.

La dernière partie de ce chapitre se focalise sur l'utilisation du ClusterCube dans l'analyse spatiale des grilles régulières de points à multi-résolution. Cette partie traite de la compatibilité du ClusterCube avec les opérateurs spatiaux (ex. slice spatiale) et les opérateurs de continuité et de multi-résolution FieldMDX présentés dans le chapitre précédent. Ainsi, nous proposons le modèle multidimensionnel du ClusterCube&Field qui permet de combiner l'approche *FISS* et le ClusterCube dans un même cube de données tout en permettant d'avoir un accès au cube de données originales.

Partie Conclusions et Perspectives.

Chapitre 6. Cette partie présente un bilan des travaux effectués et leurs limites, ainsi que les perspectives de recherches.

Partie Annexes. Cette partie comporte deux annexes :

Annexe A. Fournit le modèle XML de mapping SOLAP qui permet d'analyser un champ continu à multi-résolution représenté par une grille régulière selon plusieurs dimensions.

Annexe B. Fournit le modèle XML SOLAP qui permet de modéliser le ClusterCube pour une analyse estimée et optimisée des grilles régulières de points.

Partie II : Etat de l’art

Chapitre 2. SOLAP et l'intégration des champs continus dans l'analyse spatio-multidimensionnelle

2.1 Introduction

Pour faire face au besoin grandissant que représentent les nombreuses sources d'informations qui proviennent de divers environnements (économiques, météorologiques, …), des outils d'analyse capables de stocker, de charger et d'analyser de grands volumes de données ont été proposés. En effet, les systèmes d'information classiques non adaptés à l'analyse décisionnelle, ont été remplacés par les ED et les systèmes OLAP qui permettent d'historiser et d'analyser des masses importantes de données, tout en mettant aux mains des décideurs des outils qui leurs permettent d'avoir une vision globale ou détaillée de leurs activités.

Suite au besoin d'analyse spatiale grandissant auprès de différents organismes et provenant de sources spatiales différentes (ex. capteurs), les systèmes OLAP et les ED ont montré des limites d'analyse qui ont abouti à la naissance des EDS et de SOLAP.

Pour l'analyse des données spatiales, les EDS sont modélisés selon le modèle spatio-multidimensionnel qui étend le modèle multidimensionnel des ED pour définir les dimensions spatiales, les mesures spatiales, etc.

Les systèmes SOLAP existants permettent d'analyser efficacement les données spatiales représentées vectoriellement. Cependant, les données géographiques peuvent être représentées en utilisant un autre type de représentations appelé « champs continus », qui fournit une représentation spatialement continues et à multi-résolution des phénomènes et qui requiert une modélisation particulière adaptée au modèle spatio-multidimensionnel.

Ainsi, ce chapitre regroupe les principaux travaux sur l'analyse spatio-multidimensionnelle des champs continus et est organisé de la manière suivante :

La section 2.2 présente les concepts principaux liés à l'analyse OLAP. Dans cette section nous exposons les concepts et les définitions liés à la modélisation multidimensionnelle.

La section 2.3, introduit les principaux concepts et définitions liés à la représentation des données spatiales avec des champs continus à multi-résolution.

La section 2.4, présente les concepts principaux de SOLAP. Dans cette section nous exposons les concepts et les définitions liés à la modélisation spatio-multidimensionnelle (dimensions spatiales, mesures spatiales, …), ainsi que l'architecture typique d'un système SOLAP.

Puis, nous présentons dans la section 2.5, les principaux travaux existants dans la littérature, qui traitent de l'intégration des champs continus dans le système SOLAP, que nous comparons selon les besoins de base d'une analyse spatio-multidimensionnelle pertinente. Puis nous concluons avec une discussion dans la section 2.6.

2.2 OLAP – les concepts principaux

L'analyse de grands volumes de données représente un besoin que les outils transactionnels n'arrivent plus à satisfaire. En effet, le besoin d'analyse rapide et intuitive de données historisées a donné naissance à un type d'outils qui permettent de résumer et d'agréger ces données tout en facilitant leur interrogation avec des temps de réponses minimisés. OLAP (« Online Analytical Processing ») regroupe ces outils dédiés à l'analyse multidimensionnelle et à l'exploitation rapide des données sous plusieurs niveaux d'agrégation (Codd et Salley 1993) . Sa structure multidimensionnelle permet de conserver une historisation des données organisée thématiquement, ce qui permet aux décideurs d'effectuer leurs analyses sous différentes perspectives, en se basant sur différents critères. Les systèmes OLAP permettent aux décideurs de manipuler les données contenues dans un entrepôt de données (ED) en spécifiant les dimensions d'analyses, leurs hiérarchies et les faits qui sont le sujet de l'analyse dans le cube de données. Chaque cellule du cube de données contient un fait analysable selon les hiérarchies des dimensions qui stipulent les niveaux de granularité auxquels l'analyse sera effectuée. Des opérations, tel que le *roll-up* et le *drill-down*, permettent, respectivement, d'obtenir une vue générale ou détaillée des données

2.2.1 Dimensions et hiérarchies

Les dimensions représentent les axes sur lesquels se basent les décideurs dans leur analyse d'un phénomène donné (ex. les ventes). Elles permettent d'observer le même fait sous différentes perspectives (ex. lieu, temps, produit, …). Une dimension est un concept abstrait qui permet de regrouper les données qui partagent une sémantique commune dans le domaine d'analyse. Par exemple, la dimension temporelle est un type de dimensions qui organise le temps sur différents niveaux (jour, mois, année).

Une dimension doit avoir au moins un attribut qui la représente au niveau de granularité le plus détaillé, puis d'autres attributs qui représentent des niveaux hiérarchiques plus élevés (Pedersen et Tryfona 2001). Ces attributs constituent les membres de la dimension. Un lien hiérarchique lie les membres de chaque dimension pour permettre une exploration multi-granulaire de celle-ci. Dans le monde réel, l'organisation hiérarchique des données est souvent compliquée. Différents auteurs ont proposé des définitions pour la hiérarchisation des données.

Une hiérarchie est définie comme un ensemble de relations binaires entre les niveaux d'une dimension. Chaque niveau d'une hiérarchie est appelé un *niveau hiérarchique* et l'ensemble des niveaux hiérarchiques forment un *chemin hiérarchique*. Le niveau supérieur d'une hiérarchie est appelé un *niveau parent* et le niveau le plus bas est appelé un *niveau fils*. Les cardinalités montrent le nombre minimal et maximal des membres d'un niveau qui sont liés aux membres d'un autre niveau (Elzbieta Malinowski 2004).

A supposer que chaque dimension possède un membre « ALL » qui représente une agrégation globale de tous les membres de sa hiérarchie, la figure 2.1 montre un exemple d'une instance de la dimension temporelle "Temps" constituée de 4 niveaux de granularité (ALLTemps, Année, Mois et Jour).

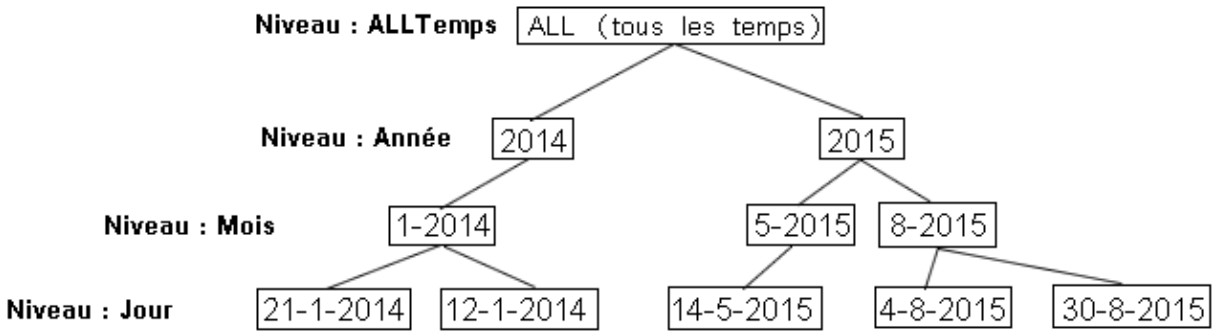


Fig 2. 1 Instance d'une hiérarchie de la dimension "Temps"

Plusieurs travaux ont aussi abordé le sujet des hiérarchies complexes. Une hiérarchie est considérée complexe si elle est non-stricte, non-couvrante, ou non-onto. Une hiérarchie est dite non-stricte si la relation entre les membres de ses niveaux est de type plusieurs-à-plusieurs (Rafanelli 2003). Par exemple, un produit qui appartient à plusieurs catégories et une catégorie qui regroupe plusieurs produits. La hiérarchie non-couvrante permet de relier un membre d'un niveau au membre d'un autre niveau en sautant un niveau intermédiaire (Torlone 2003). Par exemple, dans une hiérarchie composée des niveaux « magasin, ville, région », un magasin peut être directement lié à une région sans qu'il ne soit lié à une ville en

particulier. Une hiérarchie non-onto n'impose pas la contrainte d'agrégation (roll-up) à ses niveaux. Dans la hiérarchie non-onto un membre *m* du niveau hiérarchique *v* n'est pas essentiellement relié à un membre *m'* du niveau *v+1,* de la même hiérarchie, avec une relation d'agrégation (Rafanelli 2003).

2.2.2 Faits et mesures

Le fait est un concept qui désigne l'objet que nous voulons analyser. Par exemple, pour l'analyser des ventes d'une entreprise, les ventes sont alors considérées comme des faits. Un fait est une cellule du cube multidimensionnel qui est associée à une combinaison des membres des dimensions. Pour observer un fait nous utilisons des mesures. Un fait peut être utilisé pour calculer plusieurs mesures.

Les mesures sont le plus souvent numériques. Chaque mesure est associée à une fonction d'agrégation (somme, moyenne, …) qui permet de calculer la valeur de la mesure à un niveau hiérarchique élevé des dimensions. La mesure agrégée est ainsi appelée un agrégat (Pedersen et Tryfona 2001).

Certaines mesures peuvent être calculées à partir d'autres mesures existantes. Des formules de calcul sont appliquées sur une mesure déjà calculée pour obtenir une nouvelle mesure appelée « mesure dérivée » (Blaschka et al. 1998). Par exemple, une mesure dérivée peut être utilisée pour calculer le total des ventes toutes taxes comprises à part d'une mesure qui calcule le total des ventes hors taxes.

2.2.3 Les opérateurs OLAP

L'algèbre définie par (Codd et Salley 1993) permet la navigation dans les cellules et les niveaux hiérarchiques du cube OLAP. Les opérateurs les plus utilisés lors de l'exploitation des données dans l'OLAP sont :

Slice (sélection) : Au moyen d'un prédicat sur les attributs des dimensions, cet opérateur permet à l'utilisateur de choisir un sous-ensemble des membres sur l'ensemble des dimensions (figure 2.2 - a). Par exemple, une opération slice peut consister à ne récupérer que le nombre de ventes effectuées auprès des clients dont l'âge est inférieur à 25 ans (prédicat : âge < 25).

Roll-up : Aussi appelé «Drill-up», il regroupe les cellules du cube en se basant sur une hiérarchie d'agrégation. Cette opération modifie la granularité des données des plus détaillées

aux moins détaillées (figure 2.2 - b). Par exemple, un Roll-up peut consister à passer des ventes mensuelles aux ventes annuelles au niveau de la dimension temporelle.

Drill-down : Cet opérateur présente l'effet contraire du « Roll-up » en descendant dans la hiérarchie d'agrégation et en affichant les données détaillées (figure 2.2 - b). Le Drill-down permet, par exemple, de passer de l'analyse des ventes par départements à l'analyse des ventes par magasins.

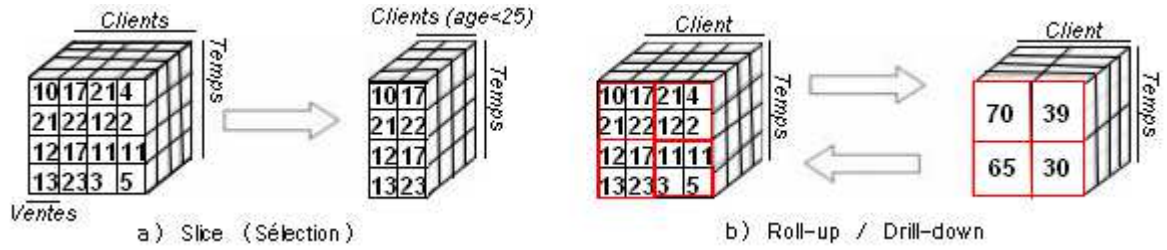


Fig 2. 2 Les opérateurs OLAP

2.2.4 L'architecture du système OLAP

L'architecture typique d'un système OLAP est généralement composée des tiers : sources de données, ETL, entrepôt de données, serveur OLAP et client OLAP. La figure 2.3 montre l'architecture classique d'un système OLAP.

Les sources de données alimentent le système OLAP en données décisionnelles. Celles-ci peuvent être de diverses hétérogénéités (encodage, unités statistiques, variables, signes conventionnels, …) et de différents types (bases de données OLTP, fichiers XML, fichiers Excel, …).

La couche d'intégration ETL (Extract Transform Load) effectue une transformation des données source pour les adapter aux schémas conventionnels de l'entrepôt de données (schéma en étoile ou schéma en flocon). Les ETL représentent une catégorie d'outils spécialisés dans le traitement de l'homogénéité, du nettoyage et des problèmes de chargement des entrepôts de données. L'ETL est souvent une combinaison complexe de processus et de technologies qui consomment une partie importante des efforts de développement des entrepôts de données et qui requiert les compétences d'analystes, de concepteurs de bases de données et de développeurs d'applications.

En général, les outils ETL permettent la transformation d'une ou plusieurs bases de données sources vers une ou plusieurs bases de données cibles. Les tâches essentielles d'un ETL sont les suivantes :

- ***Extraction des données* :** Cette tâche est effectuée périodiquement (ex. mise à jour) ou bien selon un nouvel événement (ex. modification des données sources), et consiste à extraire les nouvelles données sources pour alimenter l'entrepôt de données cible.
- ***Transformation* :** Cette tâche consiste en la vérification et l'adaptation des données sources pour qu'elles puissent être utilisées aux fins prévues.
- ***Chargement* :** Le module de chargement de données se compose de toutes les étapes nécessaires à l'administration de l'entrepôt de données cible. Il consiste en l'écriture physique des nouvelles tables dans l'entrepôt de données, y compris leurs clés primaires et secondaires et leurs attributs descriptifs.

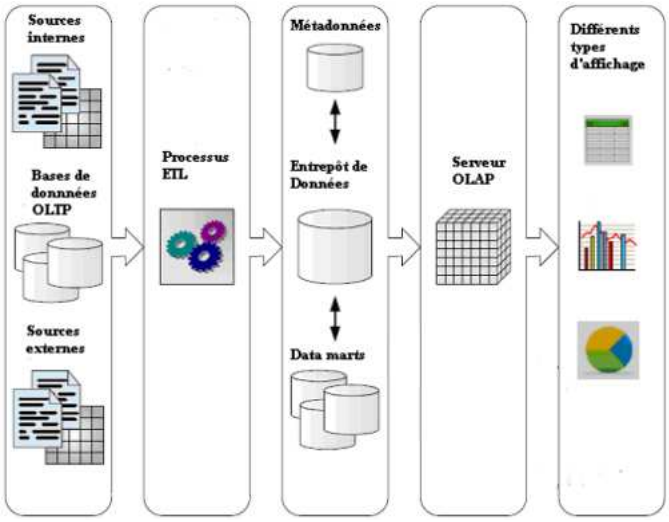


Fig 2. 3 Architecture classique d'un système OLAP. Issue de (Boulil 2012)

L'entrepôt de données (ED): Ce tiers comprend l'entrepôt de données, les métadonnées, ainsi que d'éventuels magasins de données (Datamarts). L'ED est alimenté avec un ensemble de données ciblées et organisées qui correspondent à un métier ou à un domaine précis. *Un entrepôt de données (data warehouse) est une collection de données thématiques, intégrées, non volatiles et historisées pour la prise de décisions* (Inmon 2005). Les données y sont organisées par thèmes et permettent de suivre l'évolution des différentes valeurs des indicateurs dans le temps, tout en gardant une traçabilité grâce à la non-suppression des données. L'objectif de cette synthèse des données est d'offrir une extraction d'informations pertinente et essentielle à la prise de décision.

Le serveur OLAP : Le serveur OLAP permet d'analyser les données conformément au modèle multidimensionnel, tout en procurant des temps de calcul optimisés.

Mondrian, Oracle OLAP, Microsoft Analysis Service et DB2 Olap Server sont des types de serveurs OLAP fonctionnelles. Le serveur OLAP fournit l'ensemble des opérateurs OLAP définis précédemment pour permettre aux décideurs d'avoir une vue multidimensionnelle sur leurs données. Pour faire face aux grandes quantités de données qui peuvent être entreposées dans les ED, le serveur OLAP dispose d'une politique de gestion des données particulière qui lui permet de garder les anciens résultats des requêtes en mémoire pour les réutiliser si besoin. Le serveur OLAP dispose d'un puissant moteur de calcul qui lui assure des performances maximales malgré l'utilisation fréquente de fonctions mathématiques et de procédures multidimensionnelles.

Le client OLAP : Le client OLAP met à disposition des décideurs une interface intuitive et interactive qui permet l'exploration multidimensionnelle des données et l'utilisation des opérateurs OLAP d'une manière simple et transparente. JPivot (figure 2.4-a), Cognos BI (figure 2.4-b), Microstrategy (figure 2.4-c), Polaris et Advizor, sont des exemples de clients OLAP. L'outil de visualisation le plus utilisé dans l'OLAP est la table de pivot. Il s'agit d'une table multidimensionnelle qui offre une vue imbriquée des données sur plusieurs niveaux de chaque dimension d'analyse. Son affichage tabulaire est souvent couplé à un affichage graphique (histogrammes, graphes, …).

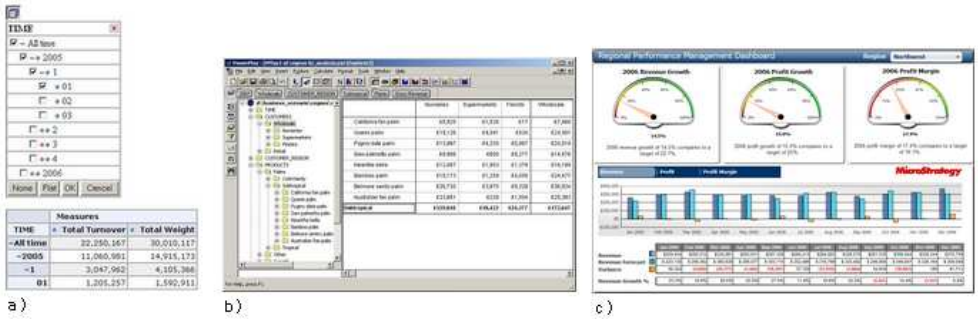


Fig 2. 4 Clients OLAP, a) JPivot, b) Cognos BI, c) Microstrategy

2.2.5 Implémentation physique du serveur OLAP (ROLAP, MOLAP et HOLAP)

Les principales approches pour l'implémentation des serveurs OLAP sont MOLAP (*Multidimensionnal OLAP*), ROLAP (*Relational OLAP*) et HOLAP (*Hybrid OLAP*).

2.2.5.1 MOLAP

Avec une approche MOLAP, les données sont extraites de l'entrepôt de données, indexées, puis stockées dans le cube physique. Étant donné que les données sont pré-agrégées, les

résultats sont rapidement renvoyés à l'utilisateur final (Wilkie et al. 2009). SAS CFO vision, Crystal Holos et Hyperion Essbase sont des exemples de serveurs MOLAP.

Cependant, bien qu'étant une technique populaire qui offre un accès rapide aux données, MOLAP présente un ensemble de défis. Le fait que les données soient stockées dans l'entrepôt de données et dans le cube MOLAP rend le processus de mise à jour complexe. En effet les modifications doivent être effectuées dans les deux endroits. Lors de la construction du cube MOLAP la majorité des efforts sont focalisés dans le transfert des données et le traitement des agrégations. Puisque les décideurs élargissent leur domaine d'analyse en y ajoutant de nouvelles dimensions et de nouveaux faits, le coût de la maintenance impose un challenge considérable.

2.2.5.2 ROLAP

Les serveurs ROLAP se basent sur l'entreposage relationnel des données dans un SGBDR (Système de Gestion de Base de Données Relationnelles), tel que PostgreSQL, IBM DB2, SQL Server, Teradata et Oracle. Le ROLAP répond aux défis imposés par MOLAP par une optimisation des tables relationnelles pour les requêtes multidimensionnelles à bas niveau, et avec des techniques d'indexation optimisées pour les requêtes d'agrégation, comme les vues matérialisées. Une vue matérialisée représente une table de données pré-agrégées et pré-stockées dans l'entrepôt de données (Blakeley, Larson et Tompa 1986).

L'implémentation ROLAP du serveur OLAP est basée sur un schéma relationnel dans l'entrepôt de données. Le schéma relationnel définit la structure relationnelle utilisée dans le stockage des tables et leurs relations. Le modèle de données le plus utilisé est le schéma en étoile (Kimball 1996). Le « schéma en étoile » (figure 2.5-a) est un modèle dé-normalisé qui est constitué, au centre, d'une table de faits et aux extrémités, des tables de dimensions. La dé-normalisation améliore les performances des requêtes en réduisant le nombre de jointures.

Le modèle de données appelé « schéma en flocon » (figure 2.5-b) est une variante du modèle en étoile. Ce modèle est normalisé pour éviter la redondance, ce qui procure un gain en stockage. Les niveaux d'une même dimension (ex. *dimension 1* de la figure 2.5-a), sont répartis chacun dans une table (ex. *dimension1 à dimension n'* de la figure 2.5-b) et chaque table dispose des attributs du niveau qu'elle représente (ex. *attribut 1.1 à attribut 1.k*) et la clé étrangère qui la lie à la table qui la précède dans la hiérarchie (ex. *attribut (n'-1).1*).

Quand plusieurs tables de faits, se partagent certaines tables de dimensions, ce modèle est alors appelé « schéma en constellation ».

Le choix du type de schémas à utiliser dépend de la nature des données et de leurs relations hiérarchiques.

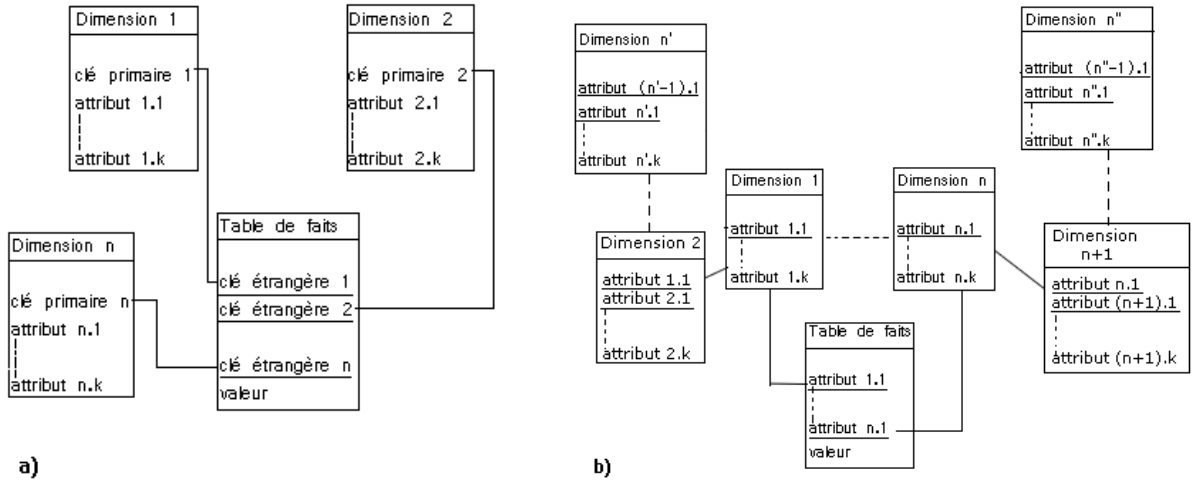


Fig 2. 5 a) modèle en étoile générique b) modèle en flocon générique

2.2.5.3 HOLAP

L'implémentation HOLAP représente une combinaison entre l'implémentation MOLAP et ROLAP. L'approche HOLAP permet d'entreposer les données dont l'accès est le plus fréquent par les utilisateurs dans une structure multidimensionnelle et le reste dans une structure relationnelle. En combinant les structure ROLAP et MOLAP, l'approche HOLAP donne accès aux données détaillées ou agrégées selon le besoin d'analyse du décideur.

2.3 Les champs continus

Dans cette section nous introduisons les principaux concepts et définitions liés à la représentation des données spatiales avec des champs continus à multi-résolution. Nous présentons les types de champs continus définis dans les SIG, ainsi que leur manipulation et leur analyse dans les systèmes d'informations géographiques.

2.3.1 Les champs continus dans les Systèmes d'Informations Géographiques (SIG)

Les données spatiales discrètes (vecteur) et les champs continus sont deux modèles différents utilisés pour représenter les données spatiales (Tomlin 1990). L'espace géographique peut être représenté avec des objets discrets (vecteurs) pour les données spatiales ayant des «

limites claires », ou avec des champs continus (continuous fields) pour les données spatiales avec des bordures floues (Couclelis 1992). La coexistence des deux approches est essentielle pour une analyse spatiale efficace. Plusieurs auteurs se sont intéressés à la définition des champs continus dans les SIG (Laurini et Pariente 1996) (Dumais et al. 2009).

Dans cette section nous commençons par présenter une panoplie de définitions qui correspondent à la représentation discrète puis continue des objets spatiaux dans les systèmes d'informations géographiques. Puis, nous décrivons les principaux opérateurs utilisés pour l'analyse des champs continus dans les SIG.

2.3.1.1 Les objets discrets

La représentation discrète considère que l'espace d'étude est vide sauf dans les endroits où se trouvent des objets dont les frontières sont bien définies. Ces objets peuvent être de différentes dimensionnalités : objets 1-D (point, ligne, …), objets 2-D (polygone, surface, …) ou objets 3-D (problématiques dans les SIG). L'objet géographique discret est utilisé pour représenter l'information géographique selon le modèle de vecteur.

Il est clair qu'aucune représentation discrète ne peut représenter toutes les caractéristiques du monde réel. Par conséquent, cette représentation impose des compromis. Les objets du monde réel et leurs relations spatiales sont représentés grâce à des géométries. Ainsi, la représentation discrète présente une limite concernant la fidélité dans la représentation de la donnée spatiale originale (Latecki 1998). En outre, les structures et les modèles multidimensionnelles existants traitants des données discrètes, ne sont pas adéquats pour l'analyse de phénomènes continus (Goodchild 1993).

Les données discrètes ou les données catégoriques ou discontinues, représentent principalement les objets discernables dans l'espace. Un objet discret présente des limites définissables. En utilisant une représentation discrète nous pouvons précisément définir là où l'objet commence et là où il se termine. Un lac au milieu d'un paysage peut être considéré comme un objet discret. Le contour qui représente l'intersection entre le lac et la terre est clair et bien défini (figure 2.6). D'autres exemples d'objets discrets comprennent les bâtiments, les routes et les parcelles. Chaque objet discret dispose d'attributs qui permettent de le définir (nom, code, …). En effet, un objet géographique est représenté avec une géométrie et un ensemble facultatif d'attributs alphanumériques ([a1 ..., an]).

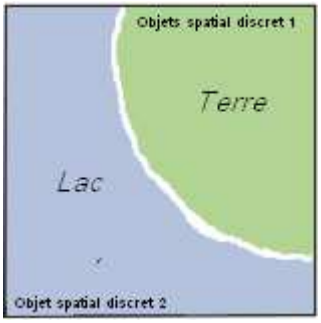


Fig 2. 6 Exemple d'objets spatiaux discrets

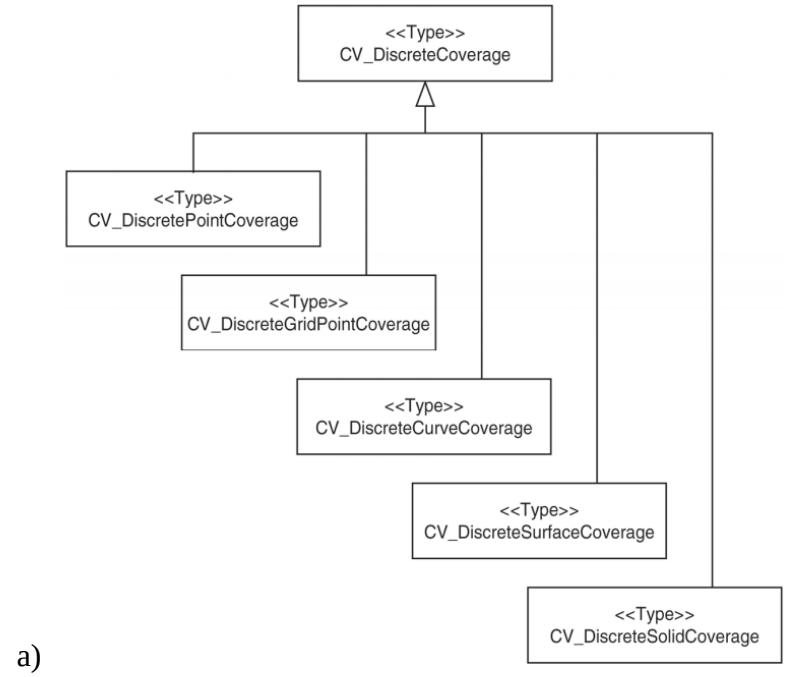
Les SGBDS, tel que PostGIS ou Oracle Spatial, proposent un type de stockage de géométries discrètes (vecteurs) permettant la création et l'accès aux données vectorielles en langage SQL. Ce stockage étend les capacités du SGBD en permettant le stockage d'objets qui représentent des entités géographiques discrètes (points, lignes et polygones) et en offrant des fonctions spatiales pour l'analyse de celles-ci, tel que les fonctions d'intersection, d'union ou de distance.

2.3.1.2 Les champs continus

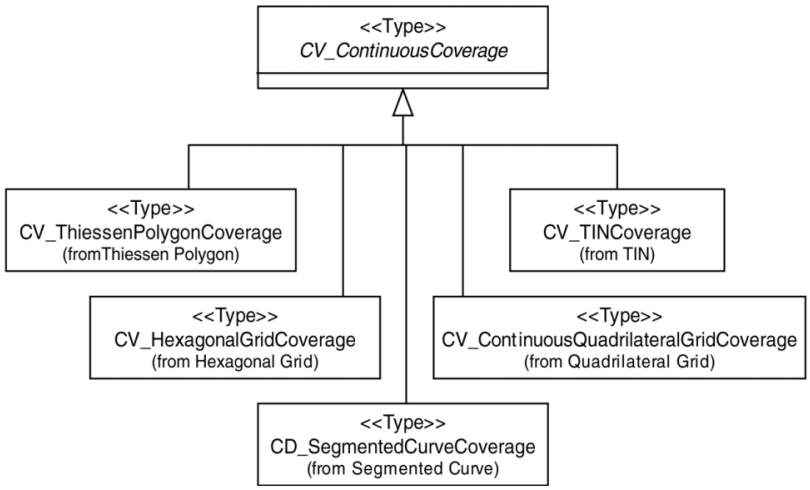
Le nombre de positions géographiques interrogeables dans un champ continu est théoriquement infini (Pariente 1994). Ainsi, ses valeurs sont considérées « continues ». Les champs continus sont pour la plupart basés sur des phénomènes physiques et la nature de leur construction dépend souvent des variables du temps et de l'espace. Le champ continu suppose que le monde réel est une série de tuiles ou de couches continues. Chacune représente la variabilité du phénomène sur une parcelle du terrain étudié. Il n'existe pas de zones vides dans un champ continu (Goodchild 1993).

Les champs continus sont généralement représentés grâce à un ensemble fini de points ou de régions dans l'espace. Cependant, l'utilisation ultérieure de ce type de données, nécessite souvent des valeurs du champ continu aux endroits non échantillonnés (Vckovski 1995).

Ainsi, deux catégories de représentations continues des données spatiales ont été proposées dans la littérature pour remédier au fait que l'on ne peut pas être fidèle à une représentation strictement continue: la représentation continue complète « Discrete coverage » (figure 2.7-a) et la représentation continue incomplète « Continuous coverage » (figure 2.7-b) (ISO 2005).



a)



b)

Fig 2. 7 La représentation des champs continus d'après ISO 19123:2005(E) a) Les représentations continues complètes b) Les représentations continues incomplètes

2.3.1.2.1 La représentation continue complète (CV_ContinuousCoverage)

La représentation continue complète consiste à diviser l'espace en sous-espaces ou en régions, en assignant une valeur à chaque sous-espace et en considérant que tous les points qui se trouvent dans la même région ont la même valeur (ex. ContinuousQuadrilateralGridCoverage ou rasters).

Selon (Mozgeris 2009), la représentation continue complète forme un espace représenté par un ensemble d'objets géométriques accompagnés d'une fonction de couverture f(x) qui assigne à chaque position (vecteur) une valeur du phénomène étudié. Un exemple de champs continus complets peut être un ensemble de polygones qui représentent une zone forestière, auxquels nous avons attribué, chacun, une valeur mesurée du phénomène de l'humidité. Voici quelques exemples de représentations continues complètes :

(1) **Triangular Irregular Network (TIN)** : défini comme le *CV_TINCoverage* dans (ISO 19123:2005(E)), le TIN est composé de maillages triangulaires qui forment des structures largement utilisées dans les systèmes d'information géographique (SIG) pour représenter des surfaces et des terrains (figure 2.8-a). (Bartholdi et Goldsman 2004) définissent le TIN comme une surface plane dont les faces sont des triangles. Chaque composant du TIN (triangle) est adjacent aux autres composants avec au moins un coté. Par exemple, les sommets dans un TIN peuvent décrire des caractéristiques du terrain comme les pics, les sommets, les fossés ou les crevasses, tandis que les bords peuvent représenter les caractéristiques linéaires du terrain comme les crêtes ou les canaux. Les TIN peuvent être organisés selon un modèle hiérarchique afin qu'ils puissent représenter un terrain à différents niveaux de résolution (Pedrini 2008). Les TIN sont devenus de plus en plus utilisés en raison de leur efficacité dans le stockage de données relatives à l'élévation du terrain.

(2) **Raster data :** défini comme le *CV_ContinuousQuadrilateralGridCoverage* dans (ISO 19123:2005(E)), ce type de représentations est aussi appelé « structure matricielle ». La représentation par rasters consiste à diviser l'espace en une grille de polygones carrés (cellules) et d'associer à chaque cellule une valeur du phénomène étudié et des attributs additionnels (figure 2.8-b). Chaque cellule est un membre à part entière de la structure matricielle. La résolution des données raster est directement liée à la taille de chaque cellule. Ainsi, le choix de la taille de la cellule doit être fidèlement adapté aux besoins d'analyse pour éviter de générer un surplus d'informations. Les photographies aériennes, les images numériques à partir des satellites, les photos numériques ou les cartes scannées sont souvent représentés avec des rasters.

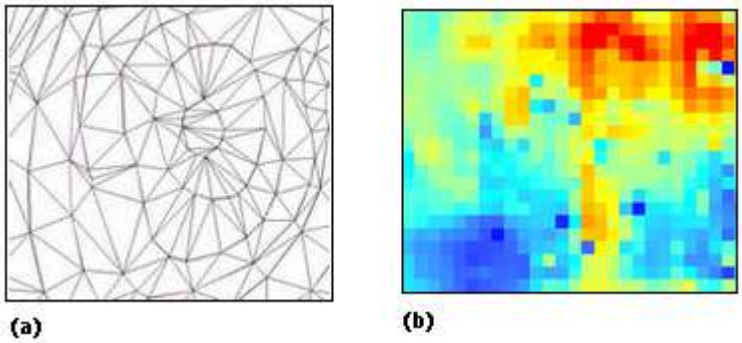


Fig 2. 8 Les représentations continues complètes ; (a) Triangular Irregular Network (TIN), (b) Données Raster

2.3.1.2.2 La représentation continue incomplète (CV_DiscreteCoverage)

La représentation continue incomplète est caractérisée par un domaine fini constitué de points. En règle générale, la surface est représentée avec un ensemble de points répartis de manière régulière ou irrégulière dans l'espace d'étude. En effet, dans une représentation continue incomplète, les valeurs sont affectées à des emplacements précis dans l'espace, puis des fonctions d'interpolation spatiale sont utilisées pour calculer les valeurs dans les emplacements non-échantillonnés (ex. DiscreteGridPointCoverage ou Grilles régulières de points).

L'interpolation spatiale permet de prédire les valeurs dans les localisations non-échantillonnées à l'aide des mesures recueillies sur un échantillon dans le même espace géographique (O'Sullivan et Unwin 2010) (Pariente 1994). La méthode d'interpolation représente un algorithme qui permet d'interpoler des valeurs en se basant sur les valeurs aux voisins les plus proches. La plupart des algorithmes d'interpolation dépendent du modèle structuré de relations spatiales entre les points. Ceux-ci sont utilisés soit pour réarranger les membres de la représentation continue de manière régulière (ex. grille régulière de points) soit pour partitionner la surface continue d'une manière régulière (ISO 19123:2005(E)).

Plusieurs types de champs continus incomplets peuvent servir à représenter des phénomènes (ex. pollution, température, propagation d'une épidémie, …) sur un espace géographique (ex. la pollution dans Clermont-Ferrand). Voici quelques exemples de champs continus incomplets :

(1) Grille régulière de points : définie comme le *CV_ DiscreteGridPointCoverage* dans (ISO 19123:2005(E)), ce type de représentations est lié à une matrice de valeurs (figure 2.9-a). Les points qui constituent la grille sont organisés d'une manière

régulière et chaque point dispose d'une valeur du phénomène étudié. Chaque point dans la grille dispose d'attributs qui décrivent son contenu et des facteurs d'échelle pertinents qui décrivent son niveau de détails (Bedient et Howcroft 1978). Nous nous sommes intéressés tout au long de cette thèse à ce type de représentations

(2) Les lignes de contours : définies comme le *CV_DiscretCurveCoverage* dans (ISO 19123:2005(E)). Les lignes de contour sont la source de données d'altitude et de types de terrains, la plus accessible et donc la plus fréquemment utilisée dans des projets environnementaux. La ligne de contour (aussi appelée isoline, isopleth, ou isarithm), d'une fonction à deux variables est une courbe le long de laquelle la fonction a une valeur constante. En d'autres termes, tous les points se trouvant sur la même ligne de contour ont la même valeur. Une carte de contours (figure 2.9-b) est une carte illustrée avec des lignes de contour, par exemple une carte topographique, qui montre des vallées et des collines. La différence entre deux contours successifs dans une carte de contours représente la différence d'altitude entre les deux (Robinson 1971).

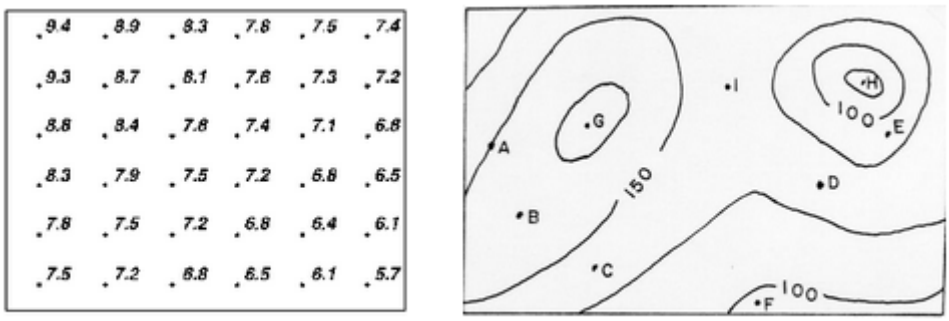


Fig 2. 9 Les représentations continues incomplètes ; (a) Grille régulière de points (b) Carte de contours

2.3.2 La hiérarchie à multi-résolution des champs continus

Les hiérarchies à multiples résolutions sont surtout proposées pour les champs continus ayant une relation de subdivision (ex. raster ou TIN). Ce type de relations résulte de l'application de manière itérative d'un opérateur de division sur le champ continu dont la résolution de base est grossière. A partir d'un champ continu, nous pouvons appliquer la subdivision pour calculer des mailles de plus en plus fines. Le résultat obtenu est une surface ayant la même topologie que le champ continu initial (Kobbelt, Vorsatz et Seidel 1999). Pour chaque type de représentations continues (raster, TIN, grille régulière de points, …) existe un processus de subdivision différent. Nous présentons dans cette section les méthodes les plus fréquemment

utilisées pour créer une hiérarchie de résolution pour les données raster et les grilles régulières de points.

2.3.2.1 Hiérarchie à multi-résolution pour les données raster (représentation complète)

Les hiérarchies de données matricielles (rasters) sont construites à partir d'une agrégation de plusieurs ensembles de cellules pour calculer les valeurs de cellules plus grossières. Les opérateurs d'agrégation classiques, tel que la moyenne, la somme, le minimum ou le maximum, peuvent être utilisés pour effectuer une analyse à multi-résolution des données raster. La figure 2.10 montre une hiérarchie matricielle à multi-résolution. La relation entre les niveaux de résolution de cette hiérarchie, est une relation d'inclusion. Les valeurs des cellules à petites échelles (50 m) sont regroupées en une seule cellule à plus grande échelle (100 m), ce qui permet de généraliser l'analyse spatiale. Ainsi, plus petites sont les cellules et meilleure est la résolution spatiale et vise versa.

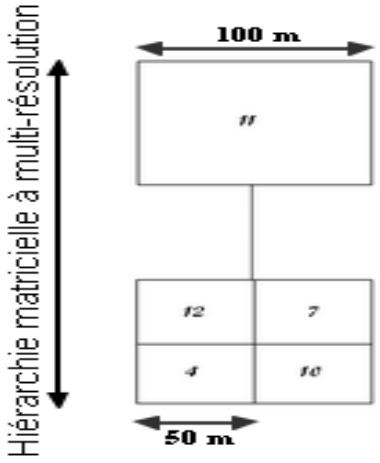


Fig 2. 10 Hiérarchie matricielle à multi-résolution

Parmi les auteurs qui se sont intéressés à la hiérarchisation des résolutions d'un raster en vue d'une analyse à multi-granularité, (Patil et Taillie 2001) décrivent le modèle HMTM (Hierarchical Markov Transition Matrix) pour la génération de résolutions spatiales hiérarchisées pour les cartes rasters. Ce modèle consiste à représenter chaque niveau de la hiérarchie de résolutions avec une matrice de transition de Markov. Le modèle HTMT génère une série de rasters. Tous les rasters de la série couvrent la même étendue spatiale, mais les rasters successifs ont une résolution de plus en plus fine. La subdivision du raster parent est effectuée grâce au partitionnement complet du Quadtree.

L’approche des pyramides de résolutions offre un puissant moyen de gérer la multi-résolution ou la multi-échelle des données raster. Celle-ci permet d’avoir une vision locale et globale des données (Jolion et Rosenfeld 1994). L’agrégation des cellules d’un raster selon une pyramide de résolutions, est une technique très utilisée dans les SIG (De Cola et Montagne 1993). Elle permet d’effectuer une compression des données qui réduit la taille des fichiers rasters ce qui contribue à faciliter leur stockage et leur manipulation. Le système d’information géographique ArcGIS propose des pyramides de données raster pour améliorer les performances (ArcGis 2012). Les pyramides permettent une hiérarchisation des jeux de données raster, tel que la taille des cellules est réduite à une échelle donnée entre deux niveaux consécutifs, comme le montre la figure 2.11.

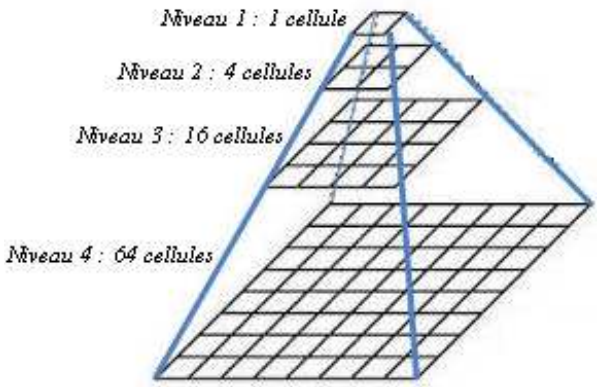


Fig 2. 11 Pyramide de résolutions raster

2.3.2.2 Hiérarchie à multi-résolution pour les grilles de points (représentation incomplète)

Le ré-échantillonnage est utilisé pour différentes fins dans les SIG. Une grille de points peut être ré-échantillonnée en une grille plus fine afin d'améliorer son apparence pour l'affichage et réduire l’espace entre ses points. Le processus de ré-échantillonnage peut être effectué récursivement pour obtenir plusieurs niveaux de résolutions à partir du niveau de base. Le ré-échantillonnage consiste généralement à augmenter le nombre de points qui forment une grille. Ce processus est souvent effectué via des méthodes d’interpolations afin de transformer la représentation discrète en représentation continue (Parker, Kenyon et Troxel 1983). Les méthodes d’interpolation servent à estimer les valeurs d’un phénomène aux localisation dépourvues de capteurs (Ahmed et Buras 2009).

La figure 2.12 représente les trois niveaux de résolution (niveau 2, niveau 3 et niveau 4) générés depuis les points du niveau de résolution de base (niveau 1). La grille au « niveau 1 » est subdivisée, puis les valeurs aux nouveaux points crées sont calculées grâce aux méthodes

d'interpolation. Ainsi, les valeurs à la résolution 2*2 sont utilisées pour estimer les valeurs aux résolutions 3*3, 5*5 et 9*9.

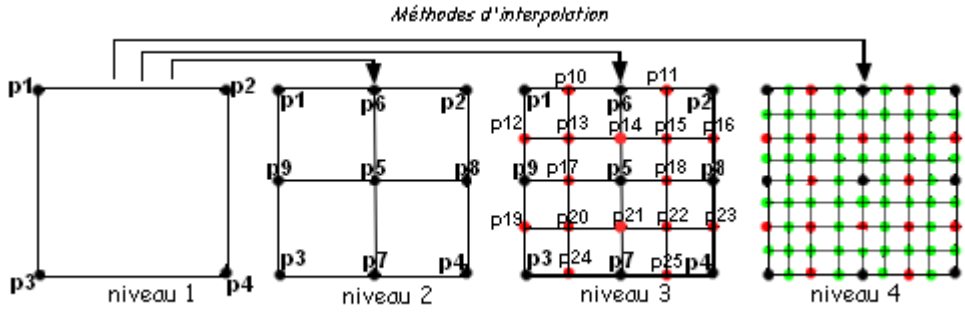


Fig 2. 12 Ré-échantillonnage d'une grille régulière de points

2.3.3 Analyse des champs continus

2.3.3.1 Map Algebra

Le travail le plus célèbre, jusque là, concernant les champs continus est certainement les opérateurs map algebra de Tomlin (Tomlin 1990). Map algebra est une algèbre implémentée dans un langage informatique qui permet l'analyse des données raster et leur calcul. L'avantage de map algebra est qu'elle permet d'effectuer des opérations complexes sur des données raster en combinant plusieurs opérations arithmétiques simples (addition, moyenne, …). Un raster est généralement composé de plusieurs couches, chacune d'elle représente une mesure particulière du phénomène observé. Par exemple, la température peut être représentée avec une couche par jour, où les cellules du raster présentent une valeur différente pour chaque jour.

Tomlin a ainsi défini 3 catégories d'opérateurs pour les données raster (figure 2.13-a) :

(1) Opérateur local : La valeur d'une cellule dans le raster de sortie est calculée en utilisant la valeur à la même cellule dans les rasters d'entrée.

(2) Opérateur focal : La valeur d'une cellule dans le raster de sortie est calculée en utilisant les valeurs des voisins de la même cellule dans le raster en entrée.

(3) Opérateur zonal : La valeur d'une cellule dans le raster de sortie est calculée en utilisant les valeurs dans une zone prédéfinie qui regroupe un ensemble de cellules du raster en entrée.

Ces opérateurs peuvent être utilisés, selon le même principe, avec les grilles régulières de points. En effet, Map Algebra est basée sur la régularité des composantes spatiales. Puisque le

raster est une grille régulière de cellules, les opérateurs Map Algebra exploitent les positions des cellules dans le calcul. Utiliser les opérateurs Map Algebra pour analyser les grilles régulières de points, revient à considérer chaque point de la grille comme une cellule. Ainsi, comme le montre la figure 2.13-b, les opérateurs : *local, focal et zonal*, peuvent être utilisées pour agréger, par exemple, les grilles régulières point par point.

A noter que même si les opérateurs map algebra sont très efficaces pour l'analyse des grilles régulières, ils restent néanmoins inutilisables face à d'autres types de représentations des champs continus (Ledoux 2008). Ainsi, (Ledoux 2008) proposent la « Voronoi-Based Map Algebra », une variante de map algebra adaptée à l'analyse des champs continus représentés par les diagrammes de Voronoi (Aurenhammer, Klein et Lee 2013). (Mennis, Viger et Tomlin 2005) proposent une extension de Map Algebra en 3-D pour l'analyse spatio-temporelle.

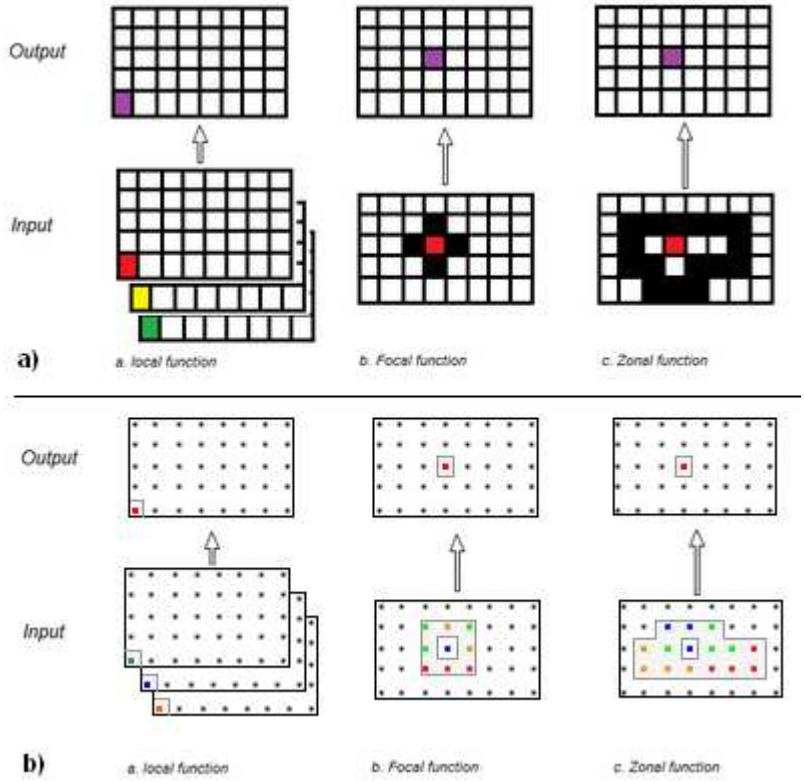


Fig 2. 13 Les opérateurs map algebra de Tomlin. a) Map Algbera avec les données raster, b) Map Algebra avec les grilles régulières de points

2.3.3.2 Méthodes d'interpolation

Il existe plusieurs méthodes d'interpolations qui sont utilisées pour effectuer une analyse continue des grilles régulières de points. Cependant, le choix de l'utilisation d'une méthode en particulier est directement lié au type d'applications à développer et aux besoins d'analyse. (Arnaud et Emery 2000) présentent diverses méthodes d'estimation et d'interpolation spatiales

de données régionalisées, ou géo-référencées. Voici quelques unes des méthodes traditionnelles d'interpolation les plus utilisées :

(1) **Interpolation du plus proche voisin :** L'interpolation par le plus proche voisin est la méthode la plus simple pour le ré-échantillonnage d'une grille de points. Cette méthode consiste à attribuer la valeur du voisin le plus proche au point que nous voulons estimer. L'intérêt de l'utilisation de cette méthode est sa simplicité d'implémentation et le gain de temps qu'elle offre.

(2) **Interpolation bilinéaire :** L'interpolation bilinéaire est une méthode qui est souvent utilisée dans les SIG. Elle fournit un bon compromis entre les performances de calculs et la qualité du rendu. Cette méthode consiste à utiliser les 4 plus proches voisins du point à estimer, pour calculer sa valeur grâce à la moyenne pondérée.

(3) **Interpolation bicubique :** L'interpolation bicubique est utilisée lorsque le temps d'exécution n'est pas une contrainte. Celle-ci fournit des résultats avec une qualité meilleure que celle produite par l'interpolation du plus proche voisin ou l'interpolation bilinéaire, mais son coût d'exécution est plus élevé. Ceci est dû au fait que l'interpolation bicubique, contrairement à l'interpolation bilinéaire, utilise 16 voisins au lieu de 4. Les grilles ré-échantillonnées avec l'interpolation bicubique ont une meilleure résolution compte tenu du nombre de voisins utilisés.

(4) **Interpolation krigeage :** L'interpolation krigeage est une méthode très utilisée dans l'analyse spatiale. Krigeage est une technique d'interpolation géostatistique qui prend en considération la distance et le degré de variation entre les points échantillonnées pour calculer les estimations aux zones inconnues. Ainsi, elle a l'avantage d'être la méthode la plus optimale au sens statistique puisqu'elle suit la tendance générale des valeurs aux échantillons (Gratton 2002). Les inconvénients de l'utilisation de cette méthode sont sa complexité d'implémentation et ses performances de calculs.

Il existe aussi d'autres méthodes pour estimer les valeurs des phénomènes analysés aux localisations non échantillonnées, tel que la méthode d'estimation basée sur les réseaux de neurones proposée dans (Pariente, Servigne et Laurini 1994)(Pariente 1994) et qui permet de satisfaire des contraintes statistiques et morphologiques tel que la préservation de l'intégrité de l'échantillon de points, ou l'estimation des sommets et des fossés. Cette technique permet aussi de réduire considérablement l'erreur d'estimation. Dans (Laurini et Pariente 1996), les auteurs proposent un langage spécifique pour la modélisation et la gestion des champs continus dans les SIG basé la même méthode d'estimation.

2.4 SOLAP – les concepts principaux

L'OLAP spatial est défini comme une plateforme permettant de supporter l'analyse spatio-multi-dimensionnelle rapide et efficace via une modélisation qui intègre les niveaux d'agrégation cartographiques, graphiques et tabulaires (Bédard, 1997).

La visualisation cartographique offre des avantages considérables lors des analyses en vue d'une prise de décision. En effet, elle permet de comprendre la distribution d'un phénomène sur une zone géographique dans un contexte multidimensionnel. Ainsi, un phénomène tel qu'une inondation peut être observé sur une carte selon différentes dimensions (temps, localisation, …) et à différentes granularités géographiques (échelles ou résolutions). Ceci permet aux décideurs d'effectuer des analyses alphanumériques tout en s'appuyant sur des résultats cartographiques pertinents. L'analyse SOLAP se base sur la modélisation spatio-multidimensionnelle.

La modélisation spatio-multidimensionnelle représente une extension de la modélisation multidimensionnelle utilisée pour les entrepôts de données et l'OLAP. Celle-ci enrichi les concepts du modèle multidimensionnel avec des concepts spatiaux tel que les dimensions spatiales, les mesures spatiales, etc.

Nous nous basons sur un exemple d'application concernant la supervision de la pollution de l'air dans les départements français, présenté dans (Sandro Bimonte, Tchounikine et Miquel 2007) pour comprendre les données spatiales et les concepts liés à leur analyse dans un système multidimensionnel.

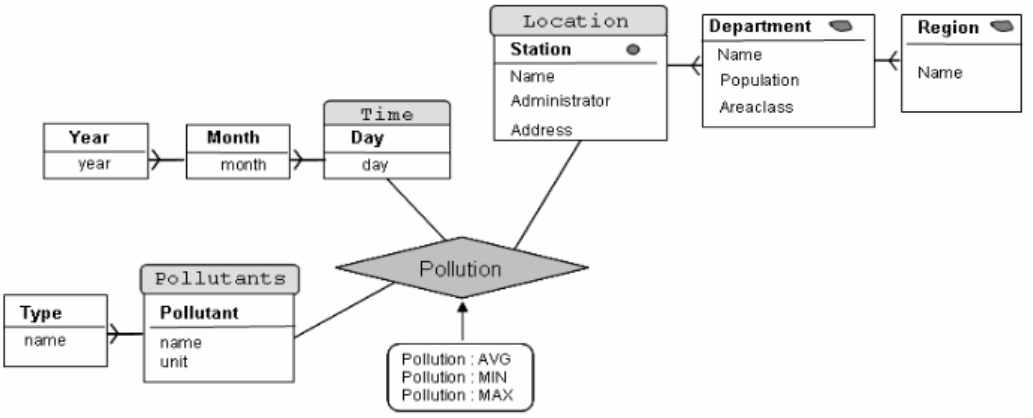


Fig 2. 14 Modèle conceptuel spatio-multidimensionnel pour l'analyse de la pollution de l'air. Issue de (Sandro Bimonte, Tchounikine et Miquel 2007)

Le modèle spatio-multidimensionnel, présenté dans la figure 2.14, permet d'analyser les valeurs de la pollution de l'air selon 3 critères (le temps, le polluant et la localisation). Ces critères sont traduits selon le modèle multidimensionnel en dimensions, puis l'objet de l'analyse (valeur de la pollution) en faits et les besoins d'analyses (ex. la valeur moyenne de la pollution) en mesures.

La dimension spatiale, qui représente la localisation où la valeur de la pollution a été mesurée, contient des données spatiales (ex. une géométrie qui représente une région) qui requièrent une modélisation particulière ainsi que des opérateurs d'analyses adéquats.

Dans ce qui suit, nous décrivons chacun des concepts spatiaux utilisés pour l'analyse et la modélisation des données spatiales. Nous nous basons sur ces concepts et ces définitions dans la suite de nos travaux.

2.4.1 La dimension spatiale

La dimension spatiale représente un axe d'analyse qui valorise l'information spatiale dans le processus d'aide à la décision. Plusieurs définitions de la dimension spatiale ont été proposées dans la littérature.

Une dimension spatiale (ex. localisation) est définie comme une dimension ayant au moins un niveau spatial (ex. villes), qui lui-même est composé d'au moins un membre spatial (ex. Paris) (S. Bimonte, Tchounikine et Miquel 2005) (Malinowski 2008). Ce niveau spatial peut être de type géométrique (ex. polygone qui représente une ville) ou non-géométrique (ex. adresse, nom de la ville, ...). (Bédard, HAN et Merrett 2001).

Dans l'exemple présenté dans la figure 2.14, la dimension « Location » est une dimension spatiale constituée de 3 niveaux d'agrégation spatiaux (« station », « department » et « region »). Chaque niveau spatial possède un type de géométries (ex. polygone), représenté par un pictogramme dans le modèle, en addition aux attributs descriptifs dont il dispose (ex. Name). Cette géométrie est utilisée pour représenter chaque membre de la dimension spatiale dans une carte thématique et pour pouvoir effectuer des analyses spatiales, impliquant des opérateurs topologiques.

Une instance de la dimension spatiale « Location » est illustrée dans la figure 2.15. Celle-ci montre des instances des niveaux spatiaux qui composent cette dimension (region, departement et station) et qui sont représentés par des polygones pour les deux premiers niveaux et par des points pour le dernier, ainsi que la hiérarchie qui les relie. Les différentes

définitions proposées dans la littérature concernant les hiérarchies spatiales sont abordées dans la sous-section suivante.

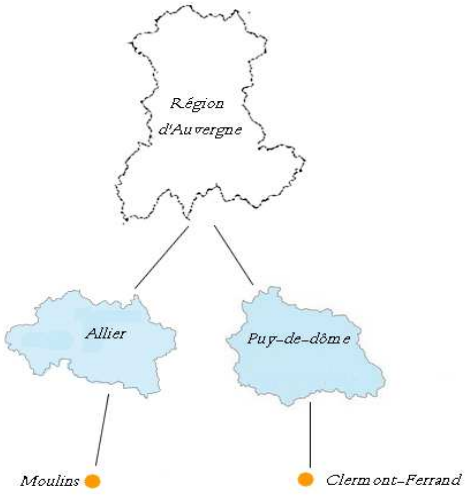


Fig 2. 15 Instance de la dimension spatiale « location »

2.4.1.1 La hiérarchie d'une dimension spatiale

Les dimensions spatiales reposent sur les hiérarchies spatiales. Celles-ci représentent la façon avec laquelle les niveaux de la dimension spatiale sont reliés entre eux. La liaison entre deux niveaux spatiaux successifs implique une relation topologique (ex. inclusion spatiale). Une hiérarchie est dite spatiale si elle est composée d'au moins un niveau qui dispose d'une information spatiale.

(Malinowski et Zimányi 2005) définissent deux types d'hiérarchies, la hiérarchie totalement spatiale et la hiérarchie partiellement spatiale. La hiérarchie totalement spatiale implique que tous les niveaux de la hiérarchie sont spatiaux et la hiérarchie partiellement spatiale implique que la hiérarchie possède au moins un niveau non spatial.

(Fidalgo et al. 2004) définissent la hiérarchie géographique comme une hiérarchie qui n'est composée que de niveaux spatiaux et la hiérarchie hybride comme une hiérarchie qui peut être composée de niveau spatiaux et de niveaux alphanumériques.

Trois types de hiérarchies spatiales sont définies dans (Rivest et al. 2003) : la hiérarchie spatiale non-géométrique, la hiérarchie spatiale géométrique et la hiérarchies spatiale mixte (figure 2.16).

- La hiérarchie spatiale non-géométrique, ne possède aucun attribut géométrique. Elle utilise la représentation nominale pour exprimer la donnée spatiale. Ce qui fait qu'elle n'est pas adaptée à une visualisation cartographique des analyses.

- La hiérarchie spatiale géométrique, implique que tous les membres de tous les niveaux de la dimension spatiale possèdent un attribut géométrique. Celle-ci permet une analyse cartographique plus adéquate.
- La hiérarchie spatiale mixte, représente un compromis entre les deux types d' hiérarchies cités précédemment. Elle consiste à avoir dans la même hiérarchie des niveaux dont les membres possèdent des géométries et d'autres niveaux dont les membres en sont dépourvus.

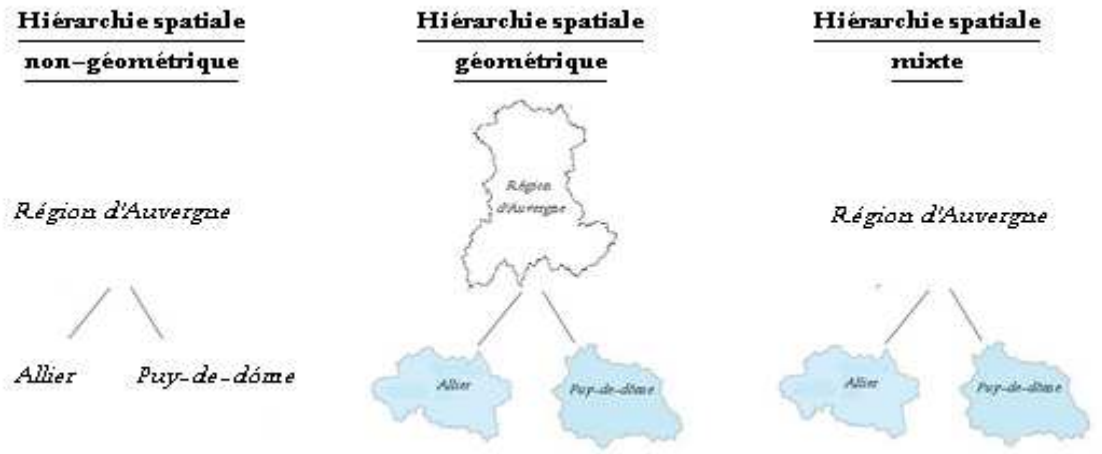


Fig 2. 16 Type d'hiérarchies spatiales (Rivest et al. 2003)

La hiérarchisation des composantes spatiales de l'information géographique permet d'exploiter le potentiel de l'analyse cartographique dans l'analyse multidimensionnelle. Ainsi, celle-ci permet d'utiliser des prédicats spatiaux dans les analyses et de changer la granularité spatiale (ex. région ←→ département) via des opérateurs SOLAP tel que le Roll-up spatial ou le Drill-down spatial, pour avoir différentes perspectives spatiales du phénomène étudié.

2.4.2 Les mesures spatiales

Les dimensions spatiales ne sont pas les seules à posséder une donnée géométrique, le modèle spatio-multidimensionnel prévoit aussi, en plus des mesures conventionnelles, des mesures spatiales. Plusieurs définitions ont été proposées dans la littérature concernant la mesure spatiale.

- (Han, Stefanovic et Koperski 1998) définissent la mesure spatiale comme un ensemble de pointeurs vers des géométries stockées en dehors de la structure multidimensionnelle.

- Une deuxième définition consiste à considérer la valeur d'une opération topologique ou métrique sur une donnée spatiale (ex. la surface d'une zone) comme une mesure spatiale (Rivest et al. 2005).

- La mesure spatiale peut aussi être considérée comme le résultat obtenu à partir des opérateurs spatiaux utilisés dans les SIG. Une intersection ou une union de plusieurs membres spatiaux de différentes dimensions spatiales résulte en un ensemble de coordonnées qui forment la mesure spatiale. Par exemple, comme le montre la figure 2.17, des coordonnées qui forment un polygone peuvent être le résultat d'une intersection entre une parcelle agricole et une zone d'inondation (Bédard, Proulx et Rivest 2005).

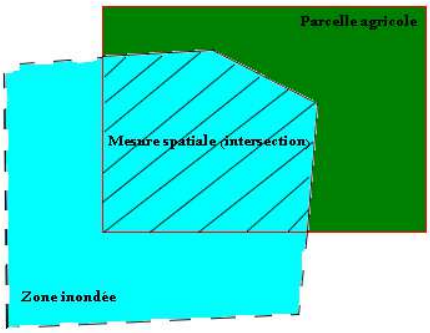


Fig 2. 17 Exemple de mesure spatiale

Ainsi, la mesure spatiale peut être considérée de différentes façons : un ensemble de pointeurs vers des données spatiales, un résultat d'une opération topologique, ou un attribut métrique. Nous pouvons considérer, en général, que la définition de la mesure spatiale appartient à 2 catégories principales : un ensemble d'objets spatiaux, ou le résultat d'une opération spatiale.

Dans la première catégorie, la mesure spatiale est réduite à la composante géométrique et ses caractéristiques alphanumériques sont intégrées à la dimension spatiale. Ce qui engendre une redondance du membre spatiale dans les faits et dans la dimension spatiale (Stefanovic, Han et Koperski 2000). Pour éliminer cette redondance, (Fidalgo et al. 2004) proposent de supprimer la composante spatiale des faits et de la remplacer avec un pointeur vers la dimension spatiale.

La deuxième catégorie implique que la mesure spatiale résulte d'une opération spatiale appliquée sur les membres des dimensions spatiales (Rivest, Bédard et Marchand 2001). Un exemple de ce type de mesures est la distance entre deux membres spatiaux.

L'objectif de la mesure spatiale est de mettre la donnée spatiale au centre de l'analyse multidimensionnelle en la présentant comme le sujet d'analyse.

2.4.3 Les opérateurs spatiaux

Les systèmes SOLAP disposent d'opérateurs spatiaux qui permettent de naviguer dans les données spatiales via une carte. Ces opérateurs permettent d'effectuer les mêmes types de forages utilisés dans les outils OLAP (Roll-up, Drill-dow, Slice et Dice), sur les données spatiales. Ces opérateurs exploitent les caractéristiques spatiales des données (géométries) pour permettre une analyse spatiale à plusieurs niveaux de granularités et selon plusieurs perspectives.

Ainsi, (Bédard, Rivest et Proulx 2007) définissent 3 opérateurs spatiaux fondamentaux :

- **Roll-up spatial :** un opérateur de forage spatial qui permet au décideur de naviguer dans une hiérarchie spatiale d'un niveau détaillé vers un niveau plus général. Il permet, selon l'exemple de la figure 2.14, de naviguer dans la dimension « location » du niveau « Station » au niveau « Région ».
- **Drill-down spatial :** un opérateur de remontage qui permet d'effectuer le processus inverse du Roll-up. Il permet au décideur de naviguer dans une dimension spatiale d'un niveau général vers un niveau plus détaillé. Par exemple, un Drill-down spatial sur la dimension « location » permet d'afficher les valeurs au niveau « Station » depuis les membres du niveau « Région ».
- **Slice spatial :** un opérateur qui permet de sélectionner un sous-ensemble de membres spatiaux d'un même niveau spatial, selon un prédicat spatial tel que la distance. Par exemple, on pourrait effectuer un slice spatial pour explorer les données sur un sous ensemble de régions, plutôt que sur tout le pays. Une requête impliquant un slice spatial, peut consister à récupérer les valeurs d'un phénomène donné, dans un périmètre précis de l'espace d'étude (ex. aux stations se trouvant à une distance inférieur à 1 km autour d'une station donnée).

2.4.4 La modélisation conceptuelle des données spatiales

Plusieurs travaux ont été proposés dans la littérature pour étendre le modèle multidimensionnel avec les concepts spatio-multidimensionnels. Les principaux modèles UML proposés sont ceux de (Boulil 2012), (Pinet et Schneider 2010) et (Glorio et Trujillo 2008).

Les travaux de (Glorio et Trujillo 2008) se focalisent sur l'extension du niveau conceptuel avec les données spatiales. Cette extension permet de générer automatiquement un modèle

logique de données prêt à être implémenté dans un SGBDS depuis le modèle conceptuel. Ils définissent la mesure spatiale et le niveau d'agrégation spatial en utilisant des stéréotypes UML sur lesquels sont définies des contraintes OCL qui garantissent leur bon usage par les concepteurs. Le modèle proposé définit le stéréotype *SpatialMeasure* pour représenter les mesures spatiales et *SpatialLevel* pour représenter les niveaux d'agrégation spatiaux. Le stéréotype *Rolls-upTo* est défini comme une association UML qui permet de relier les niveaux d'agrégation d'une dimension de manière hiérarchique. Chaque niveau d'agrégation est composé de différents attributs (identifiant, des attributs spatiaux, des attributs descriptifs, …).

Dans (Pinet et Schneider 2010), les auteurs introduisent un profil UML générique pour la conceptualisation des composantes de l'analyse multidimensionnelle et la définition des contraintes d'intégrité OCL. Le profil proposé définit les niveaux spatiaux avec des classes identifiées spatiales qui disposent chacune d'un attribut géométrique et les mesures spatiales avec des attributs UML dont le type est géométrique.

De la même façon, (Boulil 2012) propose un Framework basé sur les langages standards UML et OCL qui permet la représentation conceptuel des modèles spatio-multidimensionnels et leurs contraintes d'intégrité. Le profil UML proposé définit 3 méta-modèles :

- Un méta-modèle qui permet de définir la structure spatio-multidimensionnelle des données,
- Un méta-modèle d'agrégation qui permet de spécifier les contraintes d'agrégation, ainsi que le modèle d'analyse,
- Un méta-modèle de contraintes d'intégrité sur les requêtes SOLAP.

L'avantage du profil proposé est qu'il permet une meilleure définition des contraintes d'intégrité relatives à la qualité d'analyse SOLAP, ainsi qu'un formalisme plus complet des concepts et caractéristiques de l'analyse multidimensionnelle. Ainsi, les modèles conceptuels proposés dans cette thèse sont basés sur une extension du profil UML de (Boulil 2012).

2.4.5 La modélisation logique des données spatiales

Plusieurs travaux se sont intéressés à la modélisation logique des données spatiales dans les entrepôts de données (Siqueira et al. 2009) (Malinowski et Zimányi 2007).

La figure 2.18-a montre un modèle logique générique pour l'analyse multidimensionnelle des données spatiales. Ce modèle est un schéma en étoile classique (Kimball 1996) composé de N dimensions thématiques et de N dimensions spatiales. Chaque dimension spatiale est

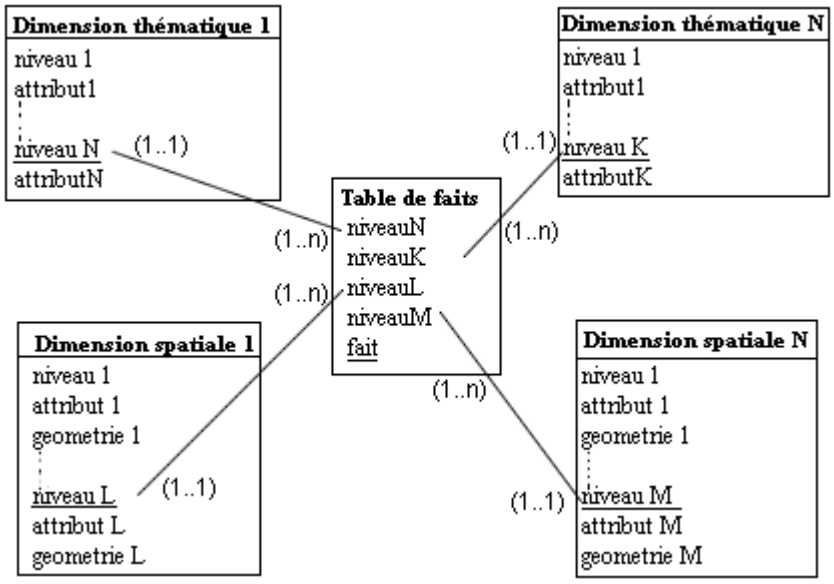
composée de niveaux spatiaux et chaque niveau de la dimension spatiale possède une géométrie qui le représente dans l'analyse cartographique et qui est considérée comme un attribut géométrique, en addition aux attributs descriptifs qu'il possède (Sampaio, de Sousa et Baptista 2006).

Un niveau spatial qui inclut une géométrie peut être représenté dans le modèle logique de deux manières différentes (Malinowski et Zimányi 2007):

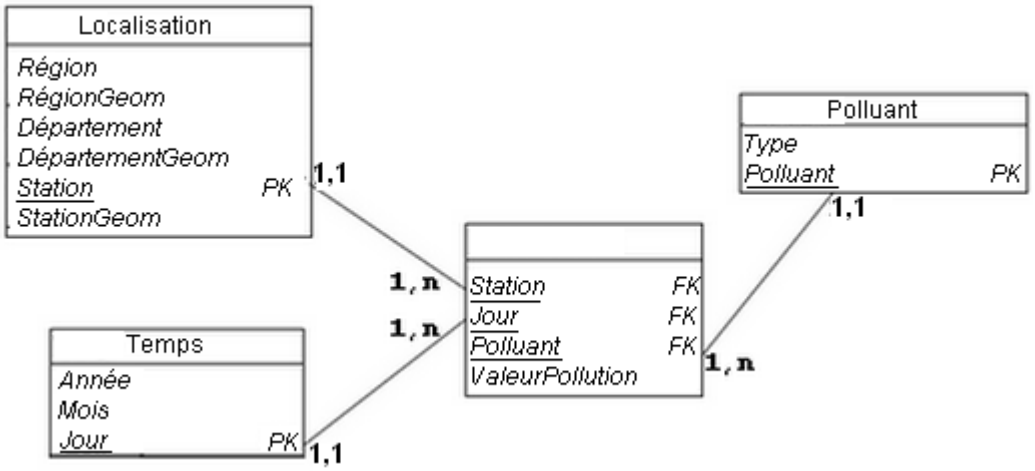
(1) Inclure la géométrie des membres à haut niveau de granularité (ex. les villes) dans la géométrie des membres du niveau parent (ex. pays). Ceci permet d'assurer la contrainte d'inclusion, qui stipule que chaque ville, par exemple, se trouve dans la géométrie qui représente le pays auquel elle appartient.

(2) Attribuer aux membres de chaque niveau de granularité spatiale une géométrie, puis inclure les contraintes topologiques explicitement. Par exemple, veiller à ce que la géométrie d'un membre du niveau ville se trouve dans la géométrie du membre du pays qui lui correspond.

a)



b)



c)

Région	RégionGeom	Département	DépartementGeom	Station	StationGeom
Auvergne	Polygone('')	PuydeDôme	Polygone('')	S1	Point('')
Auvergne	Polygone('')	PuydeDôme	Polygone('')	S2	Point('')
Auvergne	Polygone('')	Allier	Polygone('')	S3	Point('')
Auvergne	Polygone('')	Allier	Polygone('')	S4	Point('')
Paca	Polygone('')	Le Vaucluse	Polygone('')	S5	Point('')
Paca	Polygone('')	Le Var	Polygone('')	S6	Point('')

Fig 2. 18 a) Modèle logique en étoile générique b) Modèle logique en étoile pour l'analyse de la pollution de l'air c) Exemple d'instance de la table de dimension « Location »

La figure 2.18-b montre le modèle logique en étoile correspondant au modèle spatio-multidimensionnel présenté dans la figure 2.14. Ce modèle logique est constitué des tables de dimensions : *Location*, *Polluant* et *Temps* et de la table de faits. Les dimensions sont dé-normalisées et la table de fait est reliée au niveau le plus détaillé de chaque dimension (Station, Jour et Polluant). Les niveaux spatiaux de la dimension *Location* disposent d'attributs géométriques (RégionGeom, DépartementGeom et StationGeom) qui représentent les géométries des membres de chaque niveau.

Ainsi, les valeurs de pollution mesurées peuvent être analysées au niveau des stations, ou agrégées spatialement au niveau des départements ou des régions.

Une instance de la table de dimension *Location* est représentée dans la figure 2.18-c. Cette dimension est constituée de 6 stations (S1, S2, S3, S4, S5, S6) réparties sur les départements de la région d'auvergne et Paca selon une hiérarchie géométrique (cf. section 2.4.1.1). Cette hiérarchisation permet d'analyser les valeurs de pollution selon les critères de temps et du type de polluant à différents niveaux de granularités spatiales.

2.4.6 L'architecture du système SOLAP

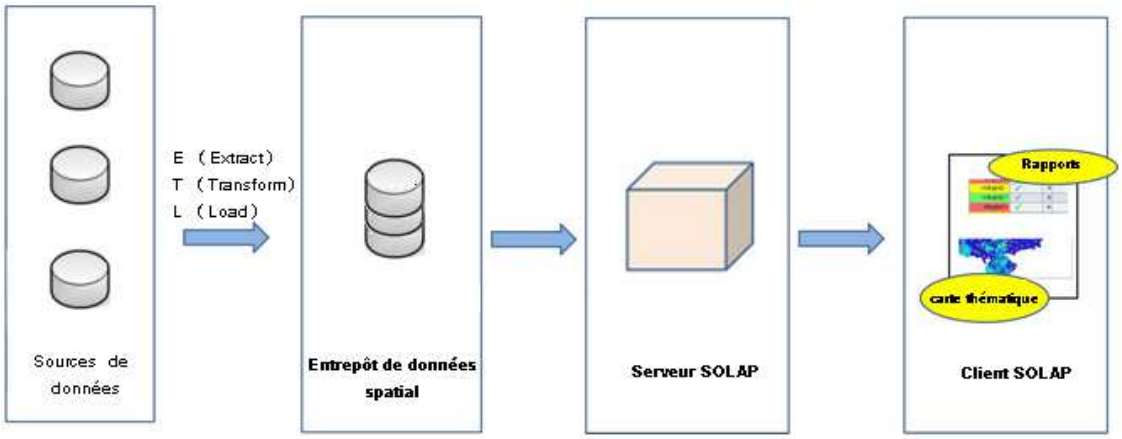


Fig 2. 19 Architecture typique d'un système SOLAP

L'architecture SOLAP typique est représentée dans la figure 2.19. Celle-ci diffère de l'architecture OLAP par une extension spatiale des tiers qui la composent.

Une fois les données sources extraites puis transformées grâce à un ETL spatial, elles sont stockées dans l'entrepôt de données pour qu'elle puisse servir à l'analyse spatio-multidimensionnelle. Cette analyse est alors sujette aux tiers : *Entrepôt de données spatial*, *serveur SOLAP* et *client SOLAP*.

(1) L'entrepôt de données spatial (EDS): Les entrepôts de données spatiales permettent d'historiser et de centraliser les données spatiales en addition aux données non-spatiales ce qui permet de les exploiter dans les systèmes décisionnels spatiaux. L'EDS est généralement implémenté par un système de gestion de base de données spatiale (SGBDS) tel que PostGIS ou Spatial Oracle. Ces systèmes permettent de stocker des informations géométriques, telles que des points, des lignes, ou des surfaces. Ils disposent aussi de fonctions qui permettent d'exploiter les caractéristiques géométriques telles que les coordonnées.

(2) Serveur SOLAP : Le serveur SOLAP implémente les cubes spatiaux ainsi que les opérateurs SOLAP spatiaux. Il permet d'exploiter les données stockées dans l'EDS grâce au schéma spatio-multidimensionnel. Le schéma spatio-multidimensionnel permet le mapping entre la structure SOLAP et la structure relationnelle, tout en définissant les concepts multidimensionnels traités précédemment (dimensions spatiales, mesures spatiales, …).

(3) Client SOLAP : Le client SOLAP représente un enrichissement du client OLAP avec la visualisation cartographique. En addition à la visualisation alphanumérique des résultats d'analyse, le client SOLAP permet d'afficher les membres des dimensions spatiales ainsi que les mesures spatiales dans des cartes thématiques. Chaque interaction de l'utilisateur avec la carte déclenche un ou plusieurs opérateurs spatiaux qui recalculent le résultat de la requête soumise et l'affichent sur la carte.

2.4.6.1 Les solutions SOLAP

Le système SOLAP repose sur l'intégration des capacités des SIG au système OLAP. Cette extension de l'OLAP avec la composante cartographique et les opérateurs qui permettent de la manipuler, permet la visualisation des membres spatiaux et alphanumériques sur une carte thématique. (Bédard, Proulx et Rivest 2005) définissent les 3 catégories de solutions OLAP disponibles : (1) *les solutions OLAP dominant*, (2) *les solutions SIG dominant* et (3) *les solutions hybrides*, qui exploitent autant les fonctionnalités OLAP que SIG.

Dans chaque catégorie, c'est l'outil dominant qui fait appel aux fonctionnalités du moins dominant. Ainsi, chaque catégorie répond à un besoin d'analyse particulier. Dans la première catégorie, la visualisation cartographique est considérée comme accessoire. Dans la deuxième, ce sont les fonctions OLAP qui sont considérées comme accessoires. Dans la dernière, les deux solutions sont considérées essentielles pour permettre une exploitation totale des deux volets.

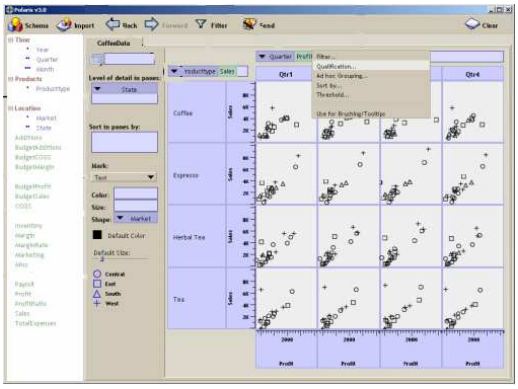
2.4.6.1.1 L'OLAP dominant

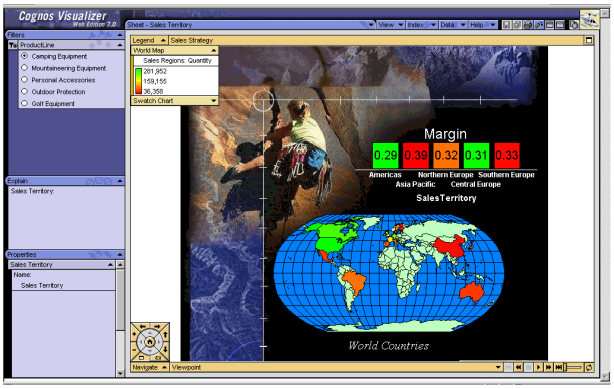
Ce type de solution intègre la totalité des fonctions OLAP et un sous ensemble des fonctionnalités des SIG. En général, les fonctionnalités SIG dont fait appel le OLAP dominant sont les fonctions d'affichage, les fonctions de déplacement dans la carte et de changement d'échelle, la sélection d'objets géométriques, ainsi que quelques fonctions de forage (Bédard, Proulx et Rivest 2005).

L'inconvénient de cette solution dans l'analyse spatio-multidimensionnelle est qu'elle ne fournit pas les outils suffisants pour une analyse spatiale pertinente. En effet, la composante spatiale n'est pas exploitée ce qui limite la solution OLAP dominant. L'utilisateur a besoin des outils SIG pour pouvoir manipuler les membres spatiaux et les mesures spatiales et ainsi pouvoir les intégrer aux requêtes multidimensionnelles.

Les solutions OLAP dominant peuvent être réparties sur 2 catégories : celles qui utilisent les cartes statistiques et celles qui utilisent des cartes interactives (Sandro Bimonte 2007).

Polaris est un exemple de solutions OLAP dominant qui utilisent les cartes statistiques (figure 2.20-a). Polaris est un outil général qui peut être utilisé pour acquérir une compréhension initiale de l'entrepôt de données, l'exploiter visuellement et explorer de façon interactive son contenu. Il peut être utilisé directement comme un outil d'exploration visuelle (Stolte, Tang et Hanrahan 2002).





(a) (b)

Fig 2. 20 (a) Interface utilisateur de POLARIS (b) Interface utilisateur de Cognos Visualizer

Pentaho Business Analytics est un produit de Business Intelligence (BI) open source qui permet l'intégration des données, l'utilisation des fonctions OLAP, le reporting et l'exploration des données, tout en mettant à disposition de l'utilisateur une visualisation cartographique des données géographiques.

KMapX est un plugiciel basé sur la technologie MapX de MapInfo qui permet le forage et la visualisation des membres spatiaux des dimensions spatiales du cube SOLAP (Bédard, Proulx et Rivest 2005).

Cognos a présenté Cognos Visualizer (figure 2.20-b), une solution d'analyse et de gestion d'entreprises qui étend PowerPlay avec des logiciels de reporting et d'analyse visuelle de pointe. Cognos Visualizer exploite la perception visuelle humaine pour permettre aux utilisateurs de visualiser de grandes quantités de données issues de plusieurs dimensions de données alphanumériques ou de données avec des composants géographiques. Entièrement intégré à Cognos Entreprise Reporting et aux solutions de traitement analytique en ligne (OLAP), Cognos Visualizer permet aux utilisateurs d'acquérir rapidement et intuitivement la perspicacité fournit par les cartes, les aidants à prendre de meilleures décisions.

2.4.6.1.2 Le SIG dominant

Le serveur OLAP peut être simulé dans une base de données relationnelle grâce à la modélisation en étoile. Ceci permet d'effectuer des agrégations en utilisant des requêtes SQL sur la base de données de manière contrôlée. Cependant, cette solution ne permet pas d'utiliser des opérateurs OLAP tel que le forage et le remontage, ou l'utilisation des concepts OLAP avancés tel que les mesures dérivées, à cause de l'absence du serveur OLAP. Les solutions SIG dominant offrent toutes la panoplie des fonctionnalités SIG et un sous-ensemble des fonctionnalités OLAP. *Cette solution couple une base de données relationnelle simulant un serveur OLAP à un logiciel SIG ou à un outil de visualisation de données spatiales* (Bédard, Proulx et Rivest 2005). CommonGIS est un exemple de solutions SIG dominant (figure 2.21). CommonGIS est un système puissant qui combine des méthodes de SIG traditionnels avec des outils innovants pour l'analyse visuelle de données et la prise de décision. CommonGIS peut être utilisé en combinaison avec le logiciel SIG commercial (ESRI ArcGIS, MapInfo, etc). Ceci est assuré par sa capacité à traiter les données spatiales dans de nombreux formats standards. Cependant, au-delà des différentes techniques de geovisualization que fournit CommonGIS, celui-ci ne propose aucun affichage tabulaire.

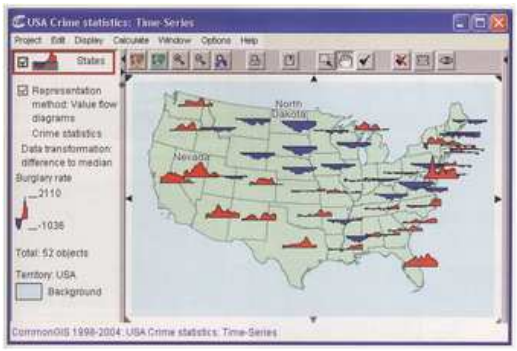


Fig 2. 21 Interface utilisateur de CommonGIS

2.4.6.1.3 La solution Hybride

Ce type de solutions intègre les fonctionnalités SIG et OLAP dans un environnement qui permet d'utiliser constamment les composantes spatiales dans le processus d'exploration et d'analyse des données. Ceci, a fait que ce type de solutions soit le plus adapté pour l'analyse spatio-multidimensionnelle pertinente.

La solution hybride est idéale pour les applications décisionnelles qui disposent d'un grand nombre de composantes spatiales et qui requièrent l'utilisation des fonctionnalités des SIG (ex. des opérateurs topologiques) dans l'analyse multidimensionnelle. L'architecture SOLAP

présentée dans la section 2.4.6, représente l'architecture SOLAP hybride typique qui utilise pleinement les capacités SIG et OLAP.

Pour que la solution hybride soit performante, les fonctionnalités OLAP ont été adaptées de manière à intégrer les opérateurs spatiaux qui caractérisent les SIG. L'utilisateur dispose de l'affichage tabulaire et graphique de l'OLAP, couplé à l'affichage cartographique des SIG. L'exploration multidimensionnelle des données est effectuée grâce à l'interaction de l'utilisateur avec la carte, d'une manière transparente, sans qu'il n'ait besoin de connaitre un langage de requêtage particulier.

Map4Decision (figure 2.22) est la première technologie Web qui intègre complètement les dimensions géo-spatiales dans un environnement d'aide à la décision et de business intelligence. Cette technologie a l'avantage d'être compatible avec la majorité des systèmes de gestion de bases de données relationnelles, ainsi que la plupart des systèmes d'informations géographiques. Map4Decision dispose d'une interface utilisateur intuitive qui permet aux utilisateurs non techniques d'accéder facilement à leurs données spatiales, de les visualiser et de les analyser. Son affichages peut inclure plusieurs cartes thématiques, des tables et des diagrammes statistiques (graphiques à barres, camemberts, etc). Map4Decision supporte la structure multidimensionnelle des données, l'affichage cartographique et non-cartographique, les dimensions et mesures spatiales, ainsi que les mesures calculées. Il effectue aussi une synchronisation de l'exploration interactive entre les différents types affichage pour faciliter l'identification et l'interprétation des données.

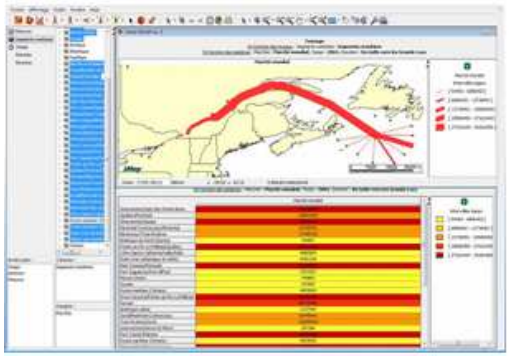


Fig 2. 22 Interface utilisateur de Map4Decision

2.5 La modélisation multidimensionnelle des champs continus dans le SOLAP

Il existe différents travaux et contributions dans la littérature SOLAP pour la gestion des dimensions spatiales, des hiérarchies spatiales et des mesures spatiales dans les analyses spatio-multidimensionnelles (Rivest et al. 2005) (S. Bimonte, Tchounikine et Miquel 2005). Cependant un nouveau besoin concernant les champs continus est apparu. Plusieurs travaux se sont intéressés à la modélisation des champs continus dans les systèmes d'informations géographiques, en encapsulant leurs caractéristiques propres et en implémentant les éléments nécessaires pour leur interaction avec d'autres types d'objet (Gordillo 2001) (Laurini et Gordillo 2000). Cependant, alors que le besoin d'analyse des champs continus dans SOLAP est de plus en plus important, plusieurs auteurs ont contribués dans ce sens ces dernières années, en proposant des approches et des méthodes pour l'intégration et la modélisation des champs continus dans les systèmes SOLAP.

2.5.1 Les travaux existants

(Ahmed et Miquel 2005) proposent un modèle multidimensionnel intégrant les champs continus incomplets (figure 2.23). Ce modèle sert à l'analyse de la pollution de l'air selon 4 dimensions : *Une dimension spatialement continue* (Localization) composée de 4 niveaux spatiaux continus (coordinates, city, department et region), *une dimension temporellement continue* (Time) composée de 4 niveaux temporels continus (hour, day, month et year) et enfin, *deux dimensions thématiques* (Pollutant et Station). L'approche de (Ahmed et Miquel 2005) consiste à stocker un échantillon des points (grille irrégulière de points), pour créer un cube discret (dont les dimensions ne sont pas continues), puis d'utiliser des fonctions d'interpolation dans le tiers client du système pour simuler une continuité des valeurs du champ continu. Les auteurs ont aussi proposé de nouveaux opérateurs, différents des opérateurs topologiques, pour analyser les champs continus, tel que l'opérateur « SpatSpeed » qui permet de calculer la vitesse de propagation d'un phénomène.

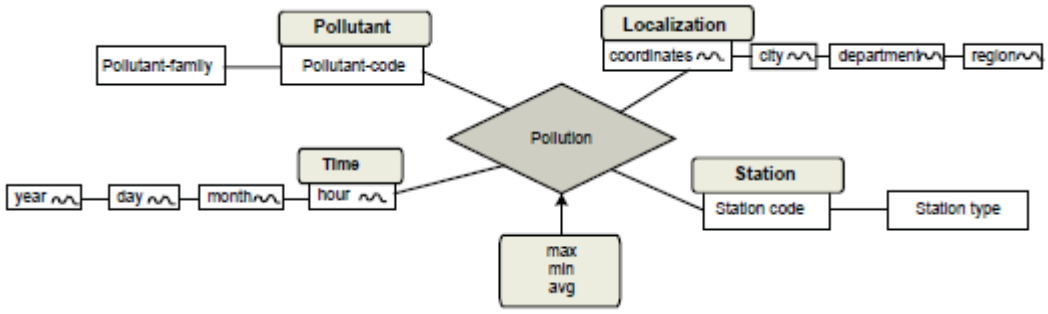


Fig 2. 23 Modèle multidimensionnel intégrant les champs continus. Issue de (Ahmed et Miquel 2005)

Les auteurs dans (McHugh 2008), définissent 4 nouveaux types de dimensions pour représenter les grilles régulières de cellules (raster) :

(1) «La dimension géométrique matricielle », qui consiste en au moins un niveau matricielle (composé de cellules). Dans ce type de dimensions, plusieurs niveaux matriciels peuvent former une hiérarchie qui représente les données à différentes résolutions ;

(2) « La dimension géométrique hybride », qui est une combinaison de niveaux vectoriels et matriciels. Cette combinaison permet de naviguer entre les deux types de représentations pour avoir des perspectives différentes des données ;

(3) « La dimension matricielle mixte », qui consiste en au moins un niveau matriciels et un niveau descriptif - elle ne doit contenir aucun niveau vectoriel ;

et enfin **(4) «La dimension matricielle hybride »**, qui consiste en au moins trois niveaux, un niveau matriciel, un niveau vectoriel et un niveau descriptif.

En combinant ces 4 types de dimensions, nous pouvons représenter les différents types de données géographiques. Les auteurs définissent aussi le « cube matriciel » dont les faits sont les cellules d'un raster. Ils proposent une approche pour analyser un raster sous plusieurs résolutions. Cette approche (cf. section 4.4.2.1) consiste à stocker les cellules du raster et leurs valeurs au niveau de résolution optimale (petites cellules), puis d'utiliser les méthodes d'agrégations spatiales (Roll-up spatial) pour généraliser la représentation (grandes cellules).

(L. Gómez, Vaisman et Zimányi 2010) définissent un modèle conceptuel multidimensionnel pour l'analyse des données spatiales continues qui intègre les données de type champ continu (figure 2.24). Ce support est réalisé par l'extension du modèle multidimensionnel SOLAP classique avec un nouveau type de données (champ continu) et les opérateurs qui lui sont associés. Le modèle conceptuel proposé permet d'analyser le rendement de la production

selon le temps (Time), les parcelles (Land plot) et la récolte (Crop), ainsi que les champs continus température, élévation, précipitations et type de sol.

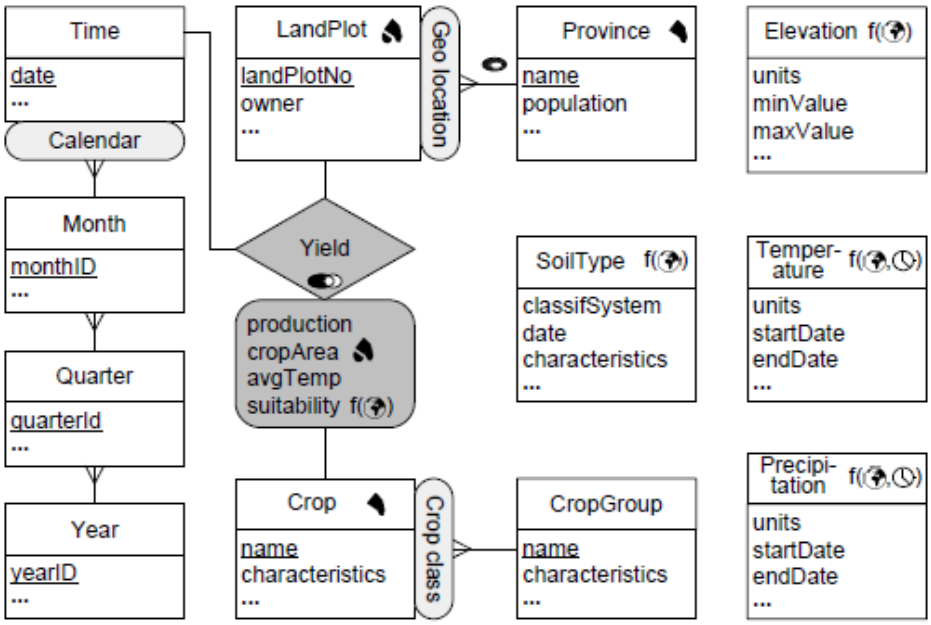


Fig 2. 24 Exemple de modèle multidimensionnel intégrant les champs continus. Issue de (L. Gómez, Vaisman et Zimányi 2010)

Dans ce modèle les auteurs définissent deux types de champs continus, « Field » qui est le champ spatialement continu (ex. Elevation) et « TempField » qui est le champ temporellement continu (ex. Temperature). Dans ce modèle les niveaux et les mesures continus et non-temporels sont identifiés par le pictogramme f (), comme pour la dimension « SoilType », tandis que les niveaux et les mesures continus temporels sont identifiés par le pictogramme f (), comme pour la dimension « Temperature ».

Les auteurs définissent le concept de « field dimension » (dimension continue) comme une dimension qui contient au moins un niveau de type champ continu et la mesure « field measure » (mesure continue) comme une mesure représentée par un champ continu (ex. suitability). La « field hierarchy » (hiérarchie continue) est définie comme une suite de niveaux continus qui permettent au champ continu d'être analysé à différentes granularités.

Les auteurs ont aussi défini plusieurs opérateurs pour l'analyse des champs continus qu'ils ont regroupé en 3 catégories : (1) les *opérateurs de projection, (2) les opérateurs d'interaction et (3) les opérateurs de variation.*

Chaque catégorie représente, respectivement, un ensemble d'opérateurs qui réalisent la projection du champ continu sur l'espace d'étude, un ensemble d'opérateur qui restreignent le

champ continu à un sous-espace et enfin un ensemble d'opérateurs qui calculent comment le champ continu varie dans l'espace (L. Gómez, Vaisman et Zimányi 2010).

Les travaux de (Sandro Bimonte et Kang 2010) présentent un modèle multidimensionnel qui intègre le type de données « champ continu », les dimensions continues, les hiérarchies continues et les mesures continues, indépendamment de leur implémentation. Les auteurs présentent aussi une représentation formelle d'un modèle multidimensionnel qui défini les concepts de dimensions et mesures continus. Pour cela, ils étendent la hiérarchie spatiale classique en définissant un arbre hiérarchique sur les coordonnées géométriques qui représentent le champ continu. De cette manière il est possible de visualiser le champ continu à différentes échelles ou résolutions. En outre, les valeurs alphanumériques associées à chaque point du champ continu sont agrégées en fonction des groupes de coordonnées spatiales définies par l'arbre hiérarchique. Ainsi, le modèle proposé utilise la représentation continue des membres spatiaux (Field objets), permettant de visualiser les mesures continues sous différentes perspectives et à différentes granularités.

(L. I. Gómez, Gómez et Vaisman 2012) présentent un modèle multidimensionnel et un modèle physique pour représenter les champs continus sous une forme discrète. Les auteurs proposent une approche pour l'utilisation des opérateurs classiques OLAP (dice, slice, drill-down,…) avec les champs continus.

(Li et al. 2014) présentent le « Tile Cube » comme une solution destinée à résoudre les problèmes de performances liés aux grands volumes de données spatiales. Le « Tile Cube » offre une combinaison entre les avantages du Map-Reduce, proposé par google et SOLAP. Il permet d'utiliser une approche basée sur le Map-Reduce pour effectuer des opérations de Roll-up et de Drill-across sur des données de type raster et ainsi améliorer leur résolution et leur rendement avec de meilleurs performances.

En se basant sur la nature des phénomènes représentés grâce aux champs continus, il parait naturel que leur modélisation doit prendre en considération leur multi-résolution. En effet, pour une analyse efficace et pertinente d'un phénomène, il faut pouvoir l'analyser sous différentes résolutions, de la plus grossière, pour avoir une vision globale du phénomène, à la plus détaillée, pour avoir une vision précise dans des zones d'intérêt prioritaires. Pour disposer d'une perspective d'analyse pertinente, l'analyste doit avoir à sa disposition des outils pour changer de résolution ou d'échelle pendant qu'il observe le comportement d'un phénomène spatial (Zhou et Jones 2003).

La représentation des données multidimensionnelles avec différents niveaux de résolutions ou d'échelles peut être vue comme de la multi-représentation. Les auteurs de (Bédard Yvan 2002) définissent un modèle conceptuel basé sur UML et implémenté dans un outil appelé « Perceptory », qui comprend plusieurs propriétés géométriques et sémantiques utilisés pour représenter les concepts spatiaux. Ils ont également montré comment modéliser efficacement plusieurs représentations géométriques ainsi que leur généralisation cartographique.

Par ailleurs, (Maryvonne Miquel 2002) suggère d'utiliser un entrepôt de données différent pour chaque représentation spatiale du même phénomène étudié.

Dans (Gascueña et al. 2009), les auteurs proposent un modèle conceptuel pour la multi-représentation des membres spatiaux. Ils définissent le « FactEntity » (FE), un modèle multidimensionnel conceptuel qui supporte la sémantique multidimensionnelle des données, la génération automatique des données, ainsi que la sémantique spatiale des données géographiques à multi-résolution. « FactEntity » permet d'avoir différentes représentations de la même donnée spatiale, dépendamment des besoins de l'application et de la thématique des données qui décrivent les données spatiales. (McGuire et al. 2008) proposent un modèle multidimensionnel en flocon pour les applications environnementales qui permet de représenter dans chaque dimension spatiale, le même membre spatial avec différentes représentations.

2.5.2 Bilan général

La table 2.1 montre une comparaison entre les différents travaux sur l'intégration des champs continus à multi-résolution dans le SOLAP. Elle permet d'identifier leurs limites dans les entrepôts de données spatiales et dans le SOLAP. Cette comparaison a été faite selon trois catégories principales : « la continuité », « les opérateurs » et « et la multi-résolution ». Chaque catégorie regroupe différents travaux qui ont tous comme objectif l'intégration des champs continus dans le système multidimensionnel SOLAP. Ainsi, nous avons comparé ces contributions pour la gestion des champs continus selon les critères suivants : *le modèle conceptuel* proposé, *le modèle logique de l'entrepôt de données*, *le modèle logique SOLAP* (implémentation du schéma multidimensionnel dans le serveur SOLAP) et *les opérateurs* proposés.

Au niveau de la continuité on peut remarquer que la plupart des travaux se sont focalisés sur la proposition de modèles conceptuels SOLAP sans suggérer de modèles logiques. Parmi les modèles conceptuels proposés, certains présentent quelques limites. Par exemple, dans le

modèle présenté par (L. Gómez, Vaisman et Zimányi 2010), les dimensions continues ne sont pas liées à la table de faits et les relations hiérarchiques entre les niveaux de type champ continu, ne sont pas définies, ce qui représente un contraste avec les modèles multidimensionnels traditionnels.

La modélisation logique des champs continus incomplets et particulièrement des grilles régulières de points, nécessite la prise en considération des fonctions de continuité, tel que l'interpolation, en addition à la composante spatiale complexe qui les représente. Par exemple, une requête qui utilise des opérateurs topologiques ne peut pas être effectuée sur des membres spatiaux crées dynamiquement par des fonctions d'interpolation, pour améliorer le résultat visuel de l'analyse, puisqu'ils n'existent pas physiquement ni dans l'entrepôt de données ni dans le serveur SOLAP. Des travaux, tel que (L. I. Gómez, Gómez et Vaisman 2012), se sont aussi intéressé à la représentation discrète des champs continus. Cependant, celle-ci ne permet par de récupérer une valeur pour chaque position de coordonnées (x,y) dans l'espace étudié, ce qui ne permet pas d'effectuer une analyse spatialement continue des données.

En ce qui concerne les opérateurs, certains auteurs ont proposé des opérateurs spécifiques pour la gestion des champs continus. Cependant, l'implémentation de ces opérateurs dans le SOLAP est limitée.

Pour la multi-résolution, on remarque qu'à l'exception des travaux présentés dans (Sandro Bimonte et al. 2012), les approches pour représenter les champs continus incomplets (ex. grille régulière de points) à différentes résolutions dans une architecture ROLAP n'existent pas. Cependant, différents auteurs ont proposé des modèles pour représenter la multi-représentation des données vectorielles. Par exemple, les travaux de (Maryvonne Miquel 2002) suggèrent d'utiliser un entrepôt de données différent pour chaque représentation spatiale pour pouvoir analyser les données spatiales à différentes échelles ou résolutions. Cependant cette méthode ne permet pas de faire une analyse en utilisant différentes représentations spatiales dans la même session - en d'autres termes, l'utilisateur doit naviguer entre plusieurs entrepôts de données spatiaux.

Il existe un manque dans la définition formelle des modèles multidimensionnels, logiques et physiques des champs continus à multi-résolution dans les systèmes d'analyse multidimensionnelle, ainsi que dans leur implémentation. Ceci est dû au fait que l'intérêt pour ce type de données dans l'analyse multidimensionnelle est toujours au stade embryonnaire.

Néanmoins, nous pensons que son utilisation pourrait être utile dans différents domaines d'application, tel que la météorologie, la médecine ou l'agriculture.

			Ahmed, & al. 2005	Mc Hugh,2008	Gomez & al. 2012	Bimonte & al. 2012	Bernier & al. 2002	Bédard & al. 2002	Gascueña & al. 2009	Gomez & al. 2011
Intégration des champs continus dans le **SOLAP**	Continuité	Modèle conceptuel	Oui (discret)	Oui (raster)	Non	Non	Non	Partiel (raster)	Non	Oui
		Modèle logique de l'ED	Non	Non	Non	Non	Non	Partiel (raster)	Non	Non
		Modèle logique SOLAP	Oui (discret)	Oui (raster)	Oui	Non	Non	Non	Non	Non
	Opérateurs sur les champs continus	Map algebra	Non	Oui	Oui	Oui	Non	Non	Non	Non
		Opérateurs spatiaux	Non	Non	Oui	Non	Non	Non	Non	Oui
		Opérateurs field	Oui	Oui (raster)	Oui	Non	Non	Non	Non	Non
	Multi-résolution	Modèle conceptuel	Non	Partiel (raster)	Non	Oui	Partiel (vector)	Partiel (raster)	Partiel (vector)	Non
		Modèle logique de l'ED	Non	Non	Non	Non	Non	Partiel (raster)	Non	Non
		Modèle logique SOLAP	Non	Non	Non	Non	Non	Non	Partiel (raster)	Non

Table 2. 1 Comparaison des différents travaux sur l'intégration des champs continus incomplets dans le SOLAP

2.6 Discussion

L'intégration des champs continus dans le système SOLAP est un concept qui suscite un intérêt grandissant dans différents travaux et pour différents objectifs. Plusieurs auteurs se sont penchés sur les différentes problématiques qui peuvent empêcher l'exploitation pertinente de ce type de données dans le SOLAP. Dans ce travail nous nous somme focalisés sur la représentation des *champs continus incomplets avec des grilles régulières de points*. Nous considérons que ce type de représentations présente un avantage intéressant au niveau

du stockage, puisque seulement un échantillon des données est entreposé dans l'EDS, ainsi qu'un avantage au niveau du choix paramétrable des fonctions d'interpolations utilisées pour l'amélioration de la résolution ou de l'échelle.

En effet, dans la représentation par *champs continus complets* les données de base sont interpolées préalablement, pour partitionner l'espace en régions et affecter une valeur du phénomène étudié à chaque région. Par contre, l'exploitation des *champs continus incomplets*, tels que les grilles régulières de points, permet d'avoir une meilleure maitrise du terrain étudié, puisqu'elle permet de récupérer des valeurs du phénomène analysé même aux espacements entre les points qui constituent la grille. Dans une analyse spatio-multidimensionnelle, l'exploitation des champs continus incomplets peut offrir un nombre de détails infinis aux décideurs si leur modélisation est fidèle au caractère continu et à multi-résolution.

Chapitre 3 Approximation des requêtes d'agrégation dans l'analyse multidimensionnelle avec les méthodes d'échantillonnage

L'approximation des requêtes OLAP est une issue qui est abordée dans plusieurs travaux pour son originalité et ses avantages. Ces techniques améliorent les performances de calcul en se basant sur des échantillons de données. Combiner ces techniques aux opérateurs et aux modèles OLAP permet de bénéficier d'une analyse multidimensionnelle moins coûteuse, en échange d'une perte de précision qui peut être évoluée selon la technique d'approximation utilisée. Cette section présente un ensemble de techniques d'approximation disponibles dans la littérature, ainsi que les différents travaux proposés pour l'intégration de ces techniques dans l'OLAP.

Ce chapitre introduit les principaux travaux proposés pour l'utilisation des techniques d'approximation pour répondre aux requêtes OLAP. La section 3.1 définit les principales techniques d'approximation utilisées pour échantillonner les données. La section 3.2 présente une comparaison des différents travaux proposés dans la littérature. Et enfin, nous concluons dans la section 3.3 avec une discussion générale.

3.1 Les techniques d'approximation

Il existe plusieurs techniques pour l'approximation d'un grand volume de données dans le but de réduire la quantité de stockage ainsi que le temps de calcul des analyses. Parmi ces techniques, il existe des techniques basées sur l'échantillonnage (clusters, random sampling, …) et d'autre sur le non-échantillonnage (ondelettes, histogrammes, …) (Das 2009).

L'utilisation des méthodes d'échantillonnage est effective quand l'analyste ne peut pas observer ou stocker tous les enregistrements liés à tous les membres étudiés et choisi un stockage partiel des données (Altmann 1974). L'échantillonnage est utilisé lorsque le traitement d'un très grand nombre de données est difficile à cause de son coût en termes de stockage et temps d'exécution (Li et al. 2008). En utilisant des méthodes d'échantillonnage on peut réduire le coût de récupération des données à partir du système de gestion de base de données (DBMS). L'échantillonnage peut être utile dans le cas d'applications qui ont besoin de calculer des agrégations sur un ensemble d'enregistrements volumineux. Cette méthode

permet de fournir efficacement et rapidement des réponses approximatives à des requêtes d'agrégation (Olken 1993). Cependant, la réponse estimée d'une requête donnée, ne peut être effective que si la méthode d'estimation utilisée est adéquate (Altmann 1974).

Si la technique d'échantillonnage appropriée est utilisée, on peut être sûr que l'échantillon est représentatif de la population et on peut estimer l'erreur qui peut résulter de cette procédure. (Barreiro et Albandoz 2001) classifient différents types de méthodes d'échantillonnage, parmi celles-ci :

- « **Random sampling with and without replacement » :** Cette technique consiste à choisir arbitrairement un échantillon de la population. Ce qui donne une chance égale à tous les membres de la population de faire partie de l'échantillon. Cette méthode peut être utilisée de deux manières différentes. Quand un membre est tiré plus d'une fois de la population pour faire partie de l'échantillon, on appel ça «random sampling with replacement». Quand un membre ne peut être choisi qu'une seule fois, on appel ça « random sampling without replacement ». Le choix d'utilisation de l'une des deux techniques repose sur la taille de la population. Quand la population est très grande les deux méthodes offrent des conclusions similaires, mais dans le cas contraire la différence entre les deux méthodes devient considérable.

- « **Cluster sampling » :** Cette méthode consiste à diviser les données en groupes d'objets similaires et utiliser un représentant de chaque groupe pour former un échantillon de la population globale. Chaque groupe comprend des objets similaires entre eux et dissimilaires avec les objets des autres groupes. Ainsi, lorsque nous avons à agréger plusieurs objets d'un même cluster, un seul objet du cluster peut être utilisé, puisque nous considérons que tous les objets d'un même cluster sont similaires. Ceci a l'avantage de réduire considérablement les temps d'exécution des agrégations et le stockage des données.

L'utilisation d'un petit nombre de clusters réduit la précision des résultats mais permet la simplification des opérations effectuées sur les données (Berkhin 2006). L'avantage de cette technique est qu'elle simplifie la collecte ou l'extraction de l'ensemble d'échantillons, ce qui nous permet d'éviter d'effectuer une liste complète des membres de la population qui participeront à l'analyse. L'inconvénient est qu'en plus du prétraitement effectué pour créer les clusters, si l'homogénéité des groupes n'est pas assurée l'échantillon final peut ne pas être représentatif de la population.

La figure 3.1 montre le principe général de cette méthode. Cette figure montre comment peut être calculée la somme des valeurs d'une population en utilisant le cluster sampling. Cette

classification permet de réduire le nombre d'opérations requis pour effectuer des calculs sur une population entière. Dans cette figure nous disposons de plusieurs valeurs d'une population (P) que nous voulons sommer. Le cluster sampling permet de grouper les valeurs par similarité (ex. **cluster1** : 4 4 et **cluster2** : 2 2 2 2), puis de pondérer une valeur représentative de chaque cluster avant de faire la somme des résultats obtenus (ex. **cluster1** : 4*2 ou **cluster2** : 2*4).

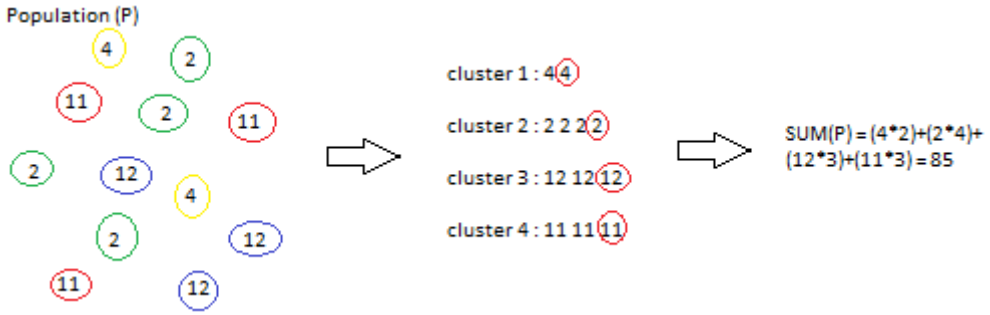


Fig 3. 1 Calcul de la somme d'une population en utilisant le cluster sampling

• **D'autres types de techniques :** Il existe d'autres techniques de sampling comme le « Two-stage » qui est un cas spécial du Cluster sampling, le « Sequential sampling » (Olken 1993), le « Systematic sampling » et le « Stratified sampling » (Barreiro et Albandoz 2001).

3.2 Les travaux sur les techniques d'approximation dans l'OLAP

3.2.1 Les travaux existants

Comme les requêtes OLAP doivent être effectuées dans un délai de temps raisonnable, lorsqu'une requête d'agrégation ne peut pas être évaluée dans le temps souhaité, une approche consiste à utiliser un estimateur statistique et de produire une estimation de la réponse de la requête. Quelques travaux se sont intéressés à l'intégration des méthodes d'échantillonnage dans l'OLAP. Dans ce chapitre nous présentons ces différents travaux afin d'étudier la possibilité de les utiliser dans une architecture SOLAP relationnelle. Nous divisons ces travaux en deux catégories :

- Les travaux qui proposent des techniques et des algorithmes pour l'exploitation des techniques d'approximation dans le but d'améliorer les performances liées aux calculs dans l'OLAP.

- Les travaux qui traitent les problèmes de précision des résultats suite aux approximations.

En ce qui concerne la première catégorie, le calcul des agrégations sur des dimensions contenant un très grand nombre de membres, est un processus qui peut poser des problèmes de performance majeurs dans plusieurs applications OLAP (Vitter et Wang 1999).

(Das 2009) a abordé les problèmes liés aux réponses approximatives des requêtes « Approximate Query Answering » (AQA), dans les applications d'aide à la décision. Les techniques d'approximation AQA sont divisées en deux catégories « sampling based » et « non-sampling based ». La première catégorie consiste à utiliser un échantillon qui sera sujet aux requêtes et la deuxième catégorie consiste à utiliser des techniques plus sophistiquées comme les ondelettes (Vitter et Wang 1999) ou les histogrammes (Ioannidis et Poosala 1999). Les techniques AQA consistent à sacrifier la précision pour améliorer le temps d'exécution en effectuant une compression avec perte de données. Différentes approches pour modéliser un système AQA ont été décrites et plus en particulier celles basée sur des méthodes de sampling. L'architecture d'AQA consiste à construire des échantillons depuis la base de données puis de rediriger les requêtes vers ces échantillons pour obtenir une estimation du résultat avec un gain de temps d'exécution considérable.

Parmi les travaux dédiés à l'utilisation des techniques d'approximation dans l'amélioration des performances des requêtes d'agrégation, (Morgenstein 1981) propose des procédures et des méthodes d'estimation de différentes opérations d'agrégation (ex. COUNT). (Ozsoyoglu et al. 1991) proposent plusieurs estimateurs statistiques pour calculer les requêtes d'agrégations relationnelles tel que la somme ou la moyenne, en se basant sur les méthodes d'échantillonnage du *simple random sampling*, *systematic sampling* et *stratified sampling*. (Han 1997) présente le OLAP mining. Il définit l'OLAP mining comme une intégration des fonctions de data mining dans l'OLAP. Il présente différentes fonctions de data mining et montre comment les intégrer dans l'OLAP mining. Parmi ces fonctions, celle qui est la plus intéressante à utiliser avec les champs continus est la « OLAP-based clustering analysis ». Cette fonction est basée sur la méthode des k-means, efficace dans le traitement de grands ensembles de données.

(Hellerstein, Haas et Wang 1997) proposent d'intégrer différentes techniques d'estimation statistique aux DBMS relationnels dans le but d'optimiser les performances liées aux requêtes d'agrégation tout en procurant un intervalle de précision optimal. Ils ont défini des objectifs

en termes d'utilisation et de performances qui doivent être pris en compte lors du design d'un tel système et ont présenté un prototype d'extension du DBMS Postgres.

(Markl, Ramsak et Bayer 1999) proposent un système de classification des faits du cube selon des dimensions à hiérarchies multiples (MHC), ce qui permet de clusteriser les données et d'améliorer les performances des requêtes OLAP. MHC permet de réduire le nombre d'accès à la table de fait lors des requêtes qui impliquent des jointures ou autre. Les auteurs présentent aussi un prototype d'implémentation du MHC en se basant sur les UB-Trees dans un environnement SQL.

(Naouali et Missaoui 2006) intègrent les principes des ensembles approximatifs au contexte multidimensionnel des données dans le but de fournir des réponses approximatives aux requêtes soumises par le décideur. Ils proposent un enrichissement des techniques OLAP avec de nouveaux opérateurs apportant plus de flexibilité lors des interactions de l'utilisateur avec l'entrepôt de données. Ainsi, ils définissent des vues matérialisées de données pour encapsuler et exploiter les résultats des opérateurs d'approximation à des fins d'extraction de connaissances (segmentation de cubes et calcul des règles d'association).

Les techniques d'approximation sont aussi utilisées dans plusieurs travaux comme une méthode de compression des données. (Shanmugasundaram, Fayyad et Bradley 1999) proposent une technique de compression des cubes OLAP basée sur la modélisation statistique des données. En utilisant la densité de la probabilité des données ils ont crée une représentation des données extrêmement compact et qui supporte les requêtes d'agrégation. Dans ce travail, les auteurs ont démontré que l'utilisation des fonctions de densité économise fortement le stockage des données puisque les données originales ne sont pas utilisées pour répondre aux requêtes. Les auteurs proposent une implémentation SQL de cette approche.

(Vitter et Wang 1999) proposent un algorithme pour créer un cube compact (compressé) depuis le cube original en utilisant des techniques d'estimation basées sur les ondelettes pour l'approximation des résultats des requêtes OLAP. En utilisant la structure hiérarchique de la décomposition en ondelettes, ce travail présente un algorithme performant pour la construction des cubes compacts.

(Feng et Wang 2002) présentent un procédé d'approximation des requêtes OLAP basé sur le clustering qui divise le cube en plusieurs segments de tailles égales puis qui classifie ces segments en plusieurs clusters. Puisque les segments de chaque cluster sont similaires entre eux, le segment centroïd est utilisé pour remplacer tous les autres segments de son cluster.

Ainsi un cube compressé est créé pour gérer des requêtes complexes avec un temps d'exécution réduit.

(Cao 2013) introduit deux nouvelles techniques spécifiques à l'approximation des requêtes OLAP, basées sur l' « uniform sampling » et le « measure based sampling ». Ces techniques sont utilisées pour définir les échantillons au niveau des sources qui alimentent l'entrepôt de données. Ainsi, un cube de données plus petit et plus simple est utilisé pour effectuer les requêtes OLAP, ce qui améliore les performances de stockage et de calcul.

Bien que les techniques d'approximation et d'estimation soient très utiles pour l'amélioration du stockage et du requêtage des données, celles-ci présentent comme inconvénient une perte de précision. La deuxième catégorie de travaux que nous présentons s'est intéressée à cette contrainte en proposant différentes approches.

(Barbará et Sullivan 1997) présentent un système qui fournit un compromis entre la précision des requêtes et la réduction du coût de stockage. Ce système peut garantir que l'erreur de n'importe quelle requête effectuée est inférieure à un certain seuil. Ils ont démontré que l'utilisation des techniques statistiques pour représenter les cubes de données peut fournir un gain de stockage au détriment d'une précision raisonnable des réponses des requêtes.

(Jermaine 2003) présente une technique appelée « Approximate Pre-Aggregation » (APA), basée sur la technique d'échantillonnage du Random Sampling pour améliorer les estimations basée sur le sampling dans les bases de données. APA utilise des statistiques sommaires supplémentaires pour réduire la vulnérabilité de l'échantillonnage à la variance. (Jin et al. 2006) présente différents estimateurs basés sur le sampling pour l'approximation des résultats des requêtes OLAP. Ces estimateurs utilisent des statistiques sommaires simples pour améliorer considérablement la précision. Les auteurs ont aussi introduit l'APA+ comme une alternative à l'APA. APA+ peut gérer des attributs catégoriques ou numériques et remédie à l'absence d'une analyse formelle de la précision de l'approche APA.

3.2.2 Bilan général des travaux existants

Dans cette section, nous comparons les différents travaux présentés sur l'utilisation des techniques d'approximation dans l'OLAP (table 3.1), selon les critères suivants :

(1) L'amélioration des performances de requêtage.

(2) La perte des données originales.

(3) La gestion de la précision des résultats obtenus par le requêtage approximé.

(4) L'implémentation des méthodes d'approximation dans une architecture OLAP relationnelle.

Ces critères représentent les piliers de l'analyse multidimensionnelle approximée. En effet, nous considérons que l'intégration des techniques d'approximation dans l'OLAP est pertinente si elle permet d'améliorer les performances de calcul, tout en gardant une trace des données originales et en offrant une précision optimale malgré l'utilisation d'échantillons de données.

La totalité des travaux sur l'approximation dans l'OLAP ont pour objectifs d'améliorer les performances de requêtage. Mais la différence entre ces travaux se matérialise dans la perte des données originales. La philosophie OLAP est basée sur la possibilité d'analyser des données historisées et hiérarchisées, sous différentes perspectives et à différentes granularités. Donc, la perte de données implique une perte de précision et une modification de la hiérarchisation des données qui risque de ne pas permettre une exploitation optimale de celles-ci.

Les travaux concernant la création de cubes de données compacts (compressés), tel que ceux présentés dans (Vitter et Wang 1999), (Cao 2013) et (Das 2009), se focalisent sur la réduction des données du cube originale, pour n'en garder qu'un échantillon. En effet, cette technique offre un gain de stockage considérable tout en améliorant les performances de calcul, mais l'inconvénient de cette approche est la perte des données originales. En effet, le décideur ne peut pas avoir accès aux données originales et aux données estimées simultanément dans la même section d'analyse. Le décideur doit choisir entre le cube compressé et le cube original ce qui l'empêche de vérifier un résultat estimé, ou au contraire estimer la même requête après s'être rendu compte qu'elle était trop gourmande.

La gestion de la précision des résultats des requêtes est une étape tout aussi importante que les performances dans le processus d'exploration des cubes approximés. En effet, le décideur a besoin d'un intervalle de confiance associé aux requêtes approximées qu'il effectue pour que son analyse garde sa pertinence. Ainsi parmi les travaux qui ont proposé des méthodes pour améliorer la précision des résultats, les travaux de (Jermaine 2003), (Jin et al. 2006) et (Hellerstein, Haas et Wang 1997) permettent d'exploiter les avantages des techniques d'approximation (amélioration des performances) tout en gardant une précision modérée, ou en proposant un intervalle de confiance pour les résultats estimés.

Bien que plusieurs auteurs ont prouvé l'efficacité et l'utilité des approximations dans l'analyse impliquant de grands volumes de données, au niveau du stockage et des

performances de calcul, la modélisation et l'implémentation de ces méthodes dans une architecture ROLAP classique n'a pas été proposée. L'architecture ROLAP offre une optimisation des tables relationnelles, ainsi que des techniques d'indexation optimisées pour les requêtes d'agrégation (cf. section 2.2.1.5.2). Intégrer les techniques d'approximation dans une architecture ROLAP permet de bénéficier des avantages d'une analyse multidimensionnelle optimisée. Certains auteurs, tel que (Markl, Ramsak et Bayer 1999), ont proposé des implémentations dans un environnement SQL, mais les problématiques liées à l'intégration des méthodes d'approximation dans les requêtes multidimensionnelles tel que MDX n'ont pas été abordées.

	Hellerstein 1997	Vitter 1999	Markl 1999	Feng 2002	Jermaine 2003	Jin 2006	Naouli 2006	Das 2009	Cao 2013
Amélioration des performances	×	×	×	×	×	×	×	×	×
Perte des données originales		×		×				×	×
Gestion de la précision	×				×	×			
Implémentation dans ROLAP	× (SQL only)		× (SQL only)						

Table 3. 1 Comparaison des différents travaux sur l'utilisation des techniques d'approximation dans l'OLAP

3.3 Discussion

Plusieurs travaux se sont intéressés à l'intégration des méthodes d'approximation dans l'OLAP dans le but de réduire le coût de stockage des données et le temps d'exécution des requêtes. Cependant, d'après nos connaissances, les travaux proposés n'offrent aucune intégration réelle de ces méthodes dans une architecture SOLAP relationnelle utilisant le langage multidimensionnel MDX.

La majorité des travaux présentés dans ce chapitre présentent des algorithmes et des méthodes pour prouver l'efficacité de l'utilisation des techniques d'approximation dans les systèmes OLAP et dans les entrepôts de données. Cependant, pour l'optimisation de l'analyse spatiale

des champs continus à multi-résolution dans le SOLAP les techniques proposées dans ce chapitre doivent être adaptées à l'analyse spatiale et introduites dans le modèle spatio-multidimensionnel d'analyse.

Ainsi, l'approche que nous proposons est basée sur les mêmes critères, cités dans la section 3.2.2. Celle-ci propose d'intégrer une méthode d'échantillonnage à l'analyse spatio-multi-dimensionnelle en permettant :

- Une amélioration des performances de calcul des requêtes.
- Une exploitation totale des données originales.
- Une implémentation dans une architecture OLAP SIG hybride relationnelle.

Nous proposons dans cette thèse une modélisation et une implémentation réelle de la méthode d'approximation, dite du cluster sampling, pour l'optimisation des requêtes d'agrégation dans une architecture ROLAP spatiale. L'approche que nous proposons vise à permettre une analyse des grilles régulières de points continues à multi-résolution, en utilisant un cube compact basé sur des échantillons pour améliorer les performances de calcul des requêtes d'agrégation. Cette approche permet d'obtenir des résultats réels depuis le cube original, ou de rediriger les requêtes d'agrégation jugées couteuses vers le cube compact pour être estimées.

Partie III : Contributions

Chapitre 4. Intégration des champs continus représentés par des grilles régulières de points dans le SOLAP : de la modélisation conceptuelle à l'implémentation

4.1 Introduction

Pour une étude efficace des phénomènes spatiaux, leur analyse continue à différents niveaux de détails (Levels Of Details - LOD) est essentielle, car elle permet de comprendre les différentes tendances et caractéristiques du phénomène étudié (Chaudhry 2009). Ainsi, plusieurs auteurs étudient les questions liées aux champs continus et aux LOD, proposant des modèles logiques et physiques et des techniques d'analyse dans le cadre des systèmes SIG et des systèmes de gestion des bases de données spatiales (SDBMS) (Parent, Spaccapietra et Zimányi 2006). L'intégration des champs continus dans les systèmes SOLAP offre des capacités d'analyse importantes. En effet, des études récentes ont porté sur l'extension des modèles spatio-multidimensionnelles et des opérateurs SOLAP pour gérer les représentations complètes et incomplètes dans les champs continus (L. I. Gómez, Gómez et Vaisman 2012). La gestion de la multi-résolution des données spatiales dans les modèles multidimensionnels a été suggérée dans certains travaux qui proposent des modèles conceptuels pour représenter un entrepôt de données spatiales avec plusieurs représentations (échelles, résolutions, etc) pour les dimensions spatiales et les mesures spatiales (Bédard Yvan 2002) (Gascueña et Guadalupe 2009).

Motivés par le besoin de modèles conceptuels (tels que UML, ER, etc) pour les applications SOLAP, en tant qu'outils essentiels permettant aux experts des ED et aux décideurs d'échanger en utilisant un langage simple commun et non ambiguë, nous proposons une extension du profil UML pour les applications SOLAP proposé par (Boulil 2012) pour la modélisation spatio-multidimensionnelles des champs continus incomplets représentés par des grille régulières de points à multi-résolution. Par ailleurs, encouragé par la nécessité des standards dans les EDS et les systèmes SOLAP, nous présentons de nouveaux modèles logiques et un langage de requêtage basé sur des standards (FieldMDX) pour représenter et interroger des champs continus incomplets à multi-résolution. En particulier, FieldMDX est une extension de MDX (le langage de requête standard OLAP de facto); il permet de générer

la continuité d'un champ incomplet avec des fonctions d'interpolation spatiale. Les modèles logiques proposés étendent le schéma en étoile relationnel classique pour représenter des champs continus incomplets à multi-résolution tout en fournissant de bonnes performances de stockage et de calcul.

Pour résumer, l'analyse spatio-multidimensionnelle des données de type champs continus doit supporter : (i) les opérateurs OLAP classiques comme map algebra, (ii) la vue continue des données spatiales, (iii) les opérateurs spatiaux (ex. slice spatial) et (iv) l'interrogation des données à différentes résolutions prédéfinies.

Ce chapitre est organisé comme suit. La section 4.2 décrit les exigences pour la modélisation logique et conceptuelle des champs continus incomplets à multi-résolution à travers une étude de cas réel. La section 4.3 présente la modélisation conceptuelle et logique de la continuité ainsi que la fonction de continuité "FieldMDX". La section 4.4 présente la modélisation de la multi-résolution du champ continu incomplet, ainsi que deux approches pour la modélisation logique de celle-ci (FISS et FASS). L'architecture relationnelle utilisée est présentée dans la section 4.5. Les performances de stockage et de calculs liées à l'analyse des champs continus incomplets représentés par des grilles régulières de points à différentes résolutions via le "FieldMDX", sont présentées à la section 4.6. La section 4.7 présente une conclusion de ce chapitre.

4.2 Besoins d'analyse

Dans cette section, nous présentons les besoins de modélisation et de requêtage multidimensionnels pour l'analyse des champs continus incomplets dans le système SOLAP en se basant sur un cas d'étude réel concernant l'analyse des données d'odeurs urbaines fournies par la société Agaetis.

4.2.1 Cas d'étude

Pour démontrer les capacités analytiques des grandes quantités de données des champs continus incomplets à multi-résolution, représentés avec des grilles régulières de points, dans les systèmes d'analyse multidimensionnelle en ligne, nous présentons un cas d'étude réel basé sur les données de l'analyse des odeurs dans des zones urbaines (figure 4.1).

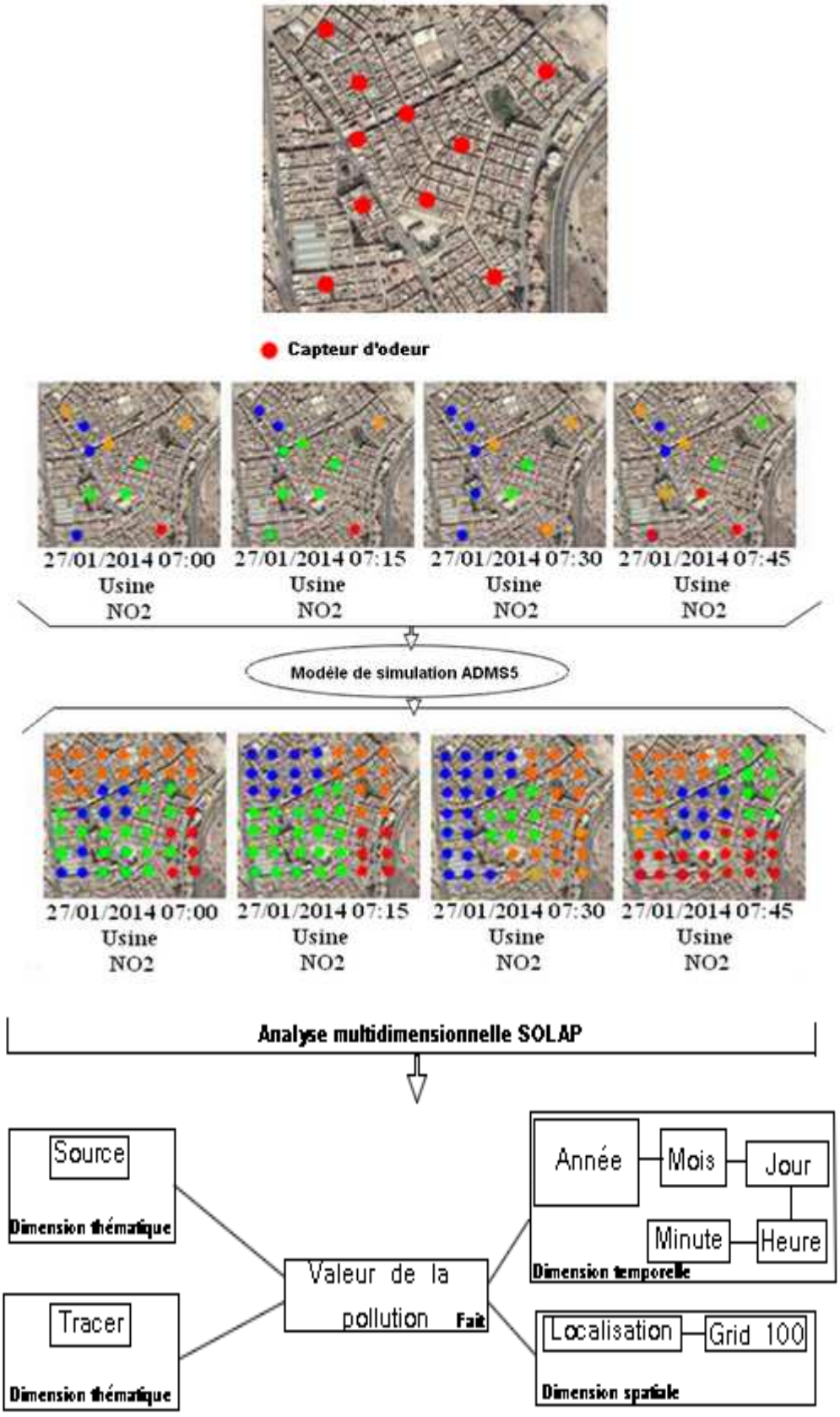


Fig 4. 1 Cas d'étude : Analyse spatio-multidimensionnelle de l'odeur sur une zone urbaine

Pour chaque période de 15 minutes, pour chaque source d'odeur et chaque type de mauvaise odeur (ex. NO2), un ensemble de valeurs sont générées depuis des capteurs depuis différentes localisations. Ces échantillons de valeurs géo référencées dans l'espace d'étude, sont utilisés dans un modèle de simulation (ADMS 2014), afin de générer des grilles régulières de points appelées *champs continus incomplets*, dont le type exacte est DiscretGridPointCoverage (ISO 19123:2005(E)). Le modèle de simulation est utilisé pour estimer les valeurs de l'odeur pour toute une zone urbaine et de produire des grilles thématiques à résolution 100 * 100 (c'est à dire, une zone carrée de 10.000 points). Le but de nos travaux est de pouvoir effectuer des analyses spatio-multidimensionnelles sur ce type de données particulier dans un système SOLAP selon les critères de temps, de source du polluant et de type de polluant (tracer). Ainsi, nous proposons un modèle spatio-multidimensionnel qui intègre les différentes dimensions ainsi que la mesure de l'odeur qui sera le sujet de l'analyse.

Le modèle spatio-multidimensionnel UML de cette application SOLAP est représenté dans la figure 4.2-a. Le fait « fact » ("Fact Table"), contient une mesure numérique (NumericalMeasure) appelée "odorValue" qui représente la valeur de l'odeur transmise par les capteurs. Celui-ci est analysé selon 4 dimensions:

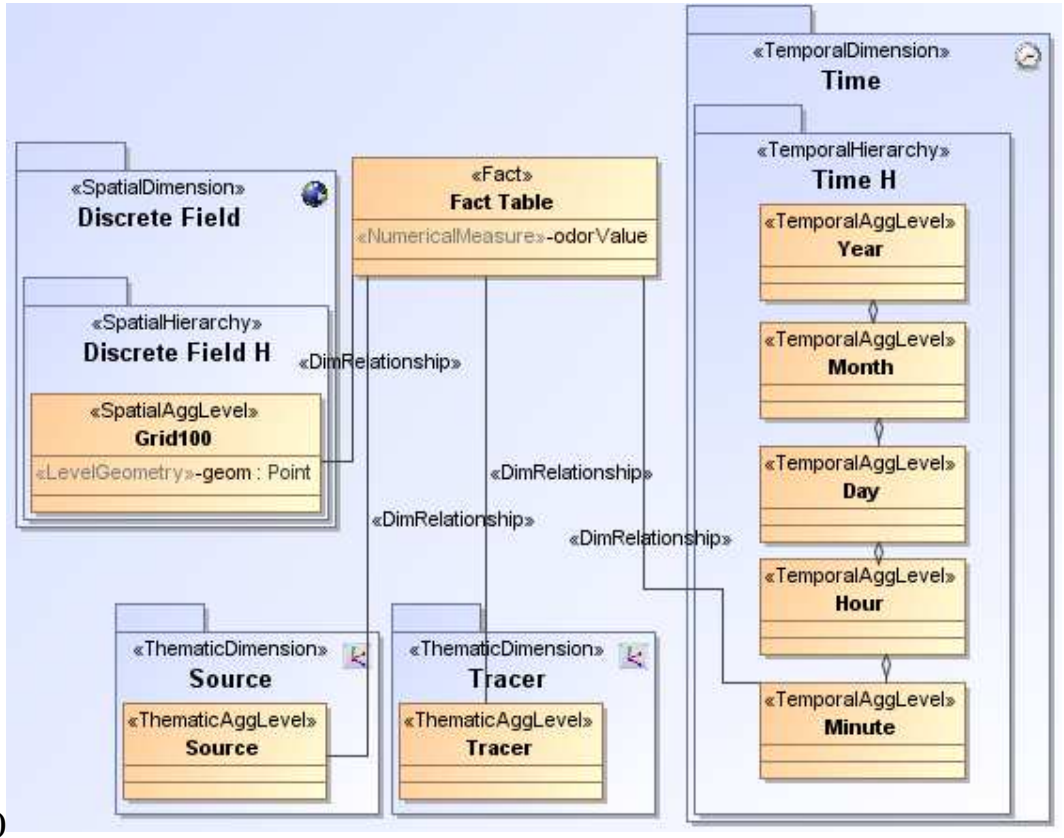
(1) Une dimension temporelle "Time" avec les niveaux "Minute", "Hour", "Day", "Month" et "Year" (« TemporalAggLevel ») ;

(2) Deux dimensions thématiques (« ThematicDimension » "Tracer" et "Source"), représentant le type de polluant (ex. NO2) et la source de l'odeur (ex. une usine) ;

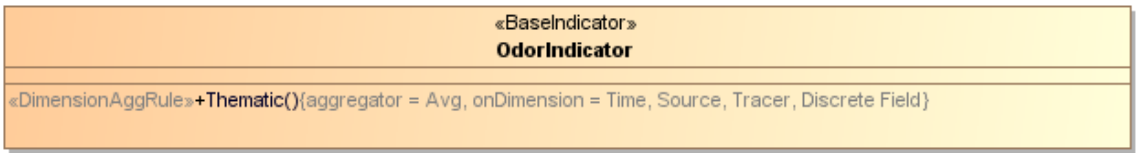
(3) Une dimension spatiale (« SpatialDimension » "Discrete Field") avec un niveau spatial discret (« SpatialAggLevel » "Grid100") représentant une grille de points discrète (« LevelGeometry » "Point").

La mesure "odorValue" est agrégée en utilisant la règle d'agrégation définie par le « BaseIndicator » comme le montre la figure 4.2-b. Le « BaseIndicator » ("OdorIndicator") défini la relation «DimensionAggRule» qui représente une règle d'agrégation pour la mesure "odorValue" sur les dimensions du cube. Cette règle d'agrégation stipule que l'opération d'agrégation (aggregator) à utiliser est la moyenne (Avg) et que cette règle est valable sur les dimensions Time, Source, Tracer et Discrete Field.

Ce modèle spatio-multidimensionnel SOLAP permet de répondre à des requêtes telles que: « Quel est la valeur "odorValue" moyenne à la 10ème heure, pour chaque point de la grille », ou bien, « Quel est la valeur "odorValue" moyenne par jour et par tracer ». La visualisation de la première requête SOLAP est représentée dans la figure 4.3-a.



(a)



(b)

Fig 4. 2 a) Modèle multidimensionnel pour les grilles régulières de points b) Règle d'agrégation sur les dimensions

4.2.2 Les requis pour l'analyse d'un champ continu incomplet à multi-résolution

Analysons maintenant les requêtes SOLAP qui agrègent les données le long d'une dimension temporelle en utilisant la moyenne pour obtenir une carte agrégée des valeurs d'odeur. Ceci est une opération OLAP appelée RollUp sur la dimension temporelle qui correspond à une opération d'agrégation map algebra, comme indiqué sur la figure 4.3-a.

Puisque l'espace est représenté d'une manière continue pour les champs continus, les décideurs doivent être en mesure d'obtenir le résultat d'une requête OLAP à n'importe quelles coordonnées dans l'espace (par exemple, ils peuvent être intéressés par la valeur de l'odeur à l'heure 10: 00 dans une zone située derrière un immeuble). Ils devraient également être en

mesure d'utiliser l'opérateur "slice spatial" sur la dimension spatiale (c.-à-d. utiliser un prédicat spatial pour sélectionner un sous-ensemble de l'espace), pour extraire, par exemple, la valeur maximale du phénomène dans une zone précise (figure 4.3-c).

Supposons qu'un décideur, ayant une vision continue du niveau spatial, demande une valeur par minute, tracer et source à un point non échantillonné comme indiqué sur la figure 4.3-b. Dans ce cas, les méthodes d'interpolation spatiale sont nécessaires, parce que pour un champ continu incomplet, seulement un échantillon de valeurs fournies par le modèle de simulation est stocké (dans notre cas, une grille régulière de points de résolution 100*100) (O'Sullivan et Unwin 2010). Dans notre approche, nous avons utilisé les fonctions d'interpolation bilinéaire et bicubique, qui sont des méthodes déterministes locales. Elles utilisent un échantillon de la grille de points (les 4 plus proches voisins pour la méthode bilinéaire et les 16 plus proches voisins pour la méthode bicubique) et calculent une moyenne pondérée en utilisant la distance pour déterminer l'influence de la valeur d'un voisin sur la valeur du point à estimer.

Enfin, comme indiqué dans la section précédente, puisque la visualisation des données spatiales à différentes résolutions est fondamentale dans le processus d'exploration / analyse, les décideurs doivent être en mesure d'exploiter les données spatiales continues à différentes résolutions. Autrement dit, ils doivent être capables d'améliorer le nombre de points qui forment la grille régulière et donc améliorer le niveau de détails (LOD) de celui-ci. Un faible niveau de détails fournit une vue globale mais floue du phénomène. Un niveau de détails élevé fournit une vue détaillée et précise.

Dans notre cas d'étude, par exemple, les décideurs ont besoin d'analyser les données à des résolutions différentes (ex. 100 * 100, 200 * 200 et 400 * 400, avec n*n une grille régulière de n^2 points) comme illustré dans la figure 4.3-d où la résolution de la grille régulière (figure 4.3-a) est améliorée de (3*3) points à (9*9) points.

Il est important de noter que, en général, pour chaque phénomène spatial, un ensemble de résolutions utiles et connues existent, de sorte qu'elles peuvent être prédéfinies selon les données et les besoins des utilisateurs. Ainsi, pour calculer les valeurs à des résolutions supérieures, les fonctions d'interpolation spatiale, telles que décrites dans la section 2.3.2.3, peuvent être utilisées.

Pour résumer, l'analyse spatio-multidimensionnelle des champs continus incomplets doit fournir:

(i) les opérateurs Map Algebra,

(ii) la représentation continue des données spatiales,

(iii) les opérateurs OLAP Spatiaux et

(iv) l'exploration à multi-résolution des données.

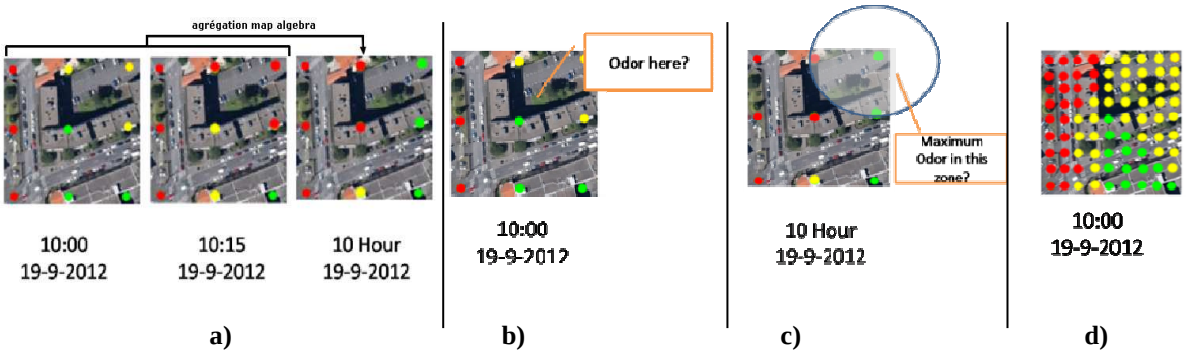


Fig 4. 3 a) Map Algebra b) La continuité c) opérateur spatial (slice) d) multi-résolution

Dans les sections suivantes nous entamerons la modélisation conceptuelle et logique des champs continus incomplets représentés par des grilles régulières de points au niveau de la continuité puis de la multi-résolution.

4.3 La Continuité

Dans cette section, nous étendons le profil UML proposé par (Boulil 2012), pour y intégrer les champs continus incomplets (Section 4.3.1), puis nous présentons le modèle logique (section 4.3.2) et l'extension MDX (section 4.3.3) qui l'implémente.

4.3.1 Le modèle conceptuel

L'idée principale consiste à améliorer le niveau spatial discret avec une fonction d'interpolation spatiale afin qu'elle soit utilisée pour estimer la valeur à chaque point non échantillonné de l'espace étudié. Nous proposons une extension du profil CI SOLAP qui est représentée sur la figure 4.4.

Le modèle CI SOLAP présenté dans (Boulil 2012), définit un profil UML pour la modélisation des entrepôts de données spatiaux qui permet de représenter conceptuellement tous les aspects avancés de la modélisation spatio-multidimensionnelle. Le profil présente des stéréotypes pour chaque élément spatio-multidimensionnel, par exemple : "Fact" pour le fait, "SpatialAggLevel" pour les niveaux des dimensions spatiales, "AggRelationship" pour les relations hiérarchiques, etc.

Dans notre extension nous définissons un niveau « FieldIncomplete » (champ continu incomplet) comme un type spécial de niveau spatial dans lequel chaque membre dispose d'une fonction d'interpolation pour représenter sa continuité (l'opération « ContinuityInterpolation ») et du niveau de résolution auquel il appartient («résolution»). Nous définissons le stéréotype « RegularGridPoint » comme un type de représentations qui hérite de la classe « FieldIncomplete » et qui est basé sur un ensemble de points, contenant chacun une valeur, répartis sur une grille régulière. Le cas d'étude dont nous disposons est basé sur ce type de représentations.

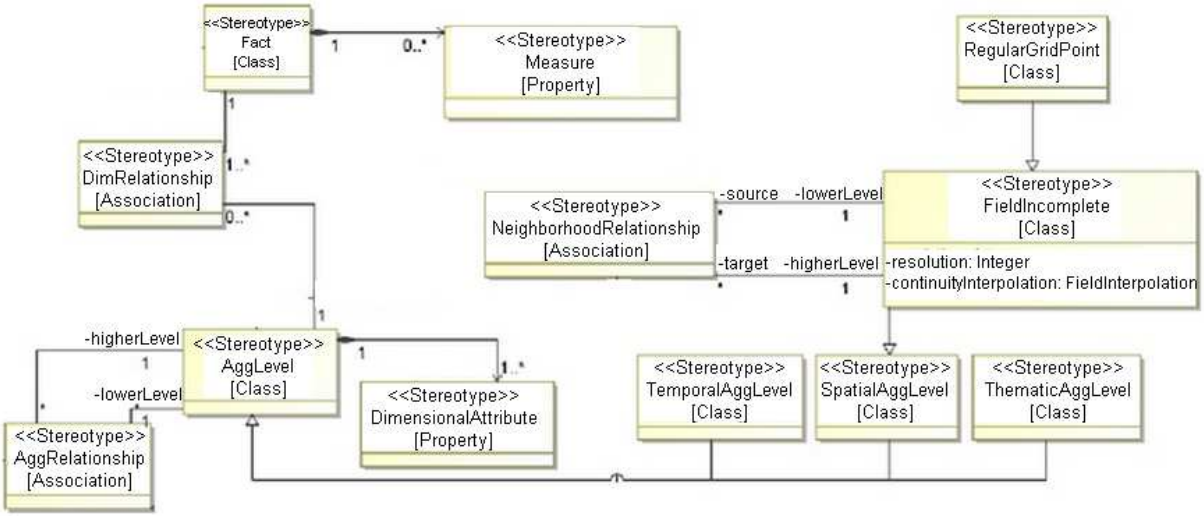


Fig 4. 4 Extension du modèle (Boulil 2012) avec les champs continus incomplets

Nous pouvons aussi préciser des restrictions sur le modèle UML proposé grâce aux contraintes exprimées via le langage formel OCL (Object Constraint Language). Les contraintes permettent d'identifier la nature des relations entre les éléments UML (Warmer et Kleppe 1998). Par exemple, la contrainte OCL suivante indique qu'un niveau de type « champ continu incomplet » dispose d'au moins une méthode d'interpolation de type « FieldInterpolation » pour générer sa continuité :

```
Context FieldIncomplete inv ContinuityOCL:
self.ownedOperation->exists (a |
a.oclIsTypeOf(FieldInterpolation))
```

Une instance du profil pour notre cas d'étude est illustrée dans la figure 4.5. Le cube est composé de la dimension spatiale « Location » qui représente un champ continu incomplet

représenté par une grille régulière de points. Celle-ci est composée d'un niveau ("Grid100") de type « RegularGridPoint », qui est composé de la géométrie "Point", son niveau de résolution et la fonction d'interpolation bilinéaire (« ContinuityInterpolation ») qui assure sa continuité. La mesure et ses règles d'agrégation sont les mêmes que celles décrites dans la figure 4.2.

En utilisant ce modèle, le décideur peut soumettre une requête telle que: "Quelles sont les valeurs de l'odeur au point de coordonnées (721148, 3140020) pour l'année 2012".

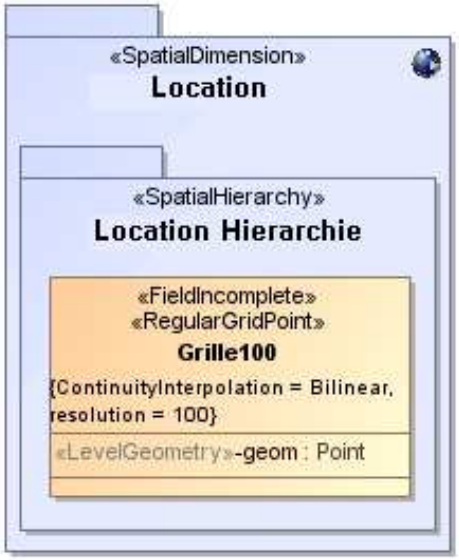


Fig 4. 5 Instance de la dimension spatiale « Location » étendue avec un niveau « FieldIncomplete » de type « RegularGridPoint »

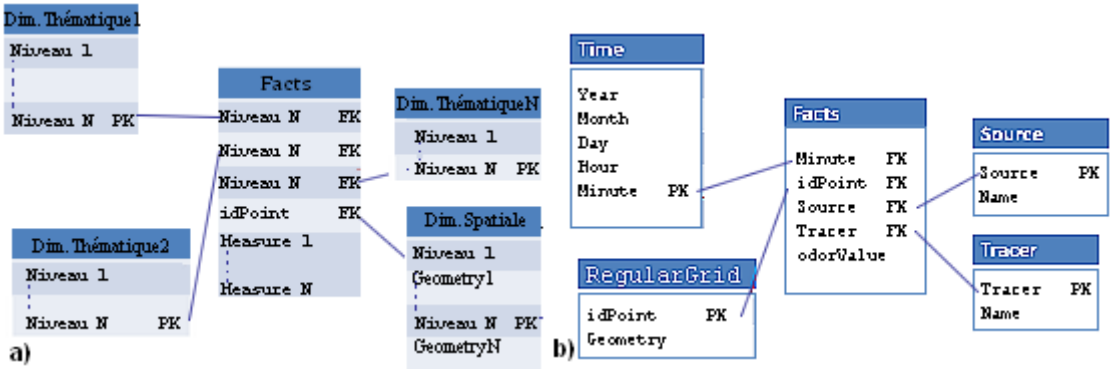
4.3.2 Le modèle logique

Le modèle logique générique pour une analyse multidimensionnelle des données spatiales est représenté dans la figure 4.6-a. Ceci est un schéma traditionnel en étoile composé de N dimensions thématiques, chacune composée de plusieurs niveaux de granularité et d'une dimension spatiale (Dim. Spatiale) composée de plusieurs niveaux spatiaux et d'un ensemble de géométries pour représenter chaque niveau. La table de faits contient des mesures et des clés étrangères des tables des dimensions. Chaque table de dimension est dénormalisée et possède des attributs qui représentent ses niveaux. La table de faits est classiquement reliée au niveau le plus bas (le plus détaillé) de chaque dimension.

La figure 4.6-b représente une instance du modèle logique de la figure 4.6-a pour notre cas d'étude. Dans cette instance le modèle est composé de la dimension temporelle "Time", des dimensions thématiques "Source" et "Tracer" et de la dimension spatiale "RegularGrid" qui représente un champ continu incomplet sous forme d'une grille régulière de points.

La dimension "RegularGrid" se compose d'un niveau représentant une grille régulière de points à la résolution 100 * 100 (10 000 points). Chaque membre de cette dimension est constitué d'un identifiant et d'une géométrie qui représente un point. Ce modèle logique nous permet de retrouver la valeur "odorValue" à chaque point ("idPoint") de la table "Regular Grid". Cependant, ce modèle logique tel que défini ne peut répondre qu'aux requêtes impliquant des membres spatiaux existants dans la dimension spatiale. Ce modèle ne peut en aucun cas fournir une valeur "odorValue" à une position de la grille régulière non-échantillonnée. La figure 4.6-c montre une instance des tables du modèle logique (figure 4.6-b) dans l'EDS.

Pour que la dimension "Regular Grid" soit continue, celle-ci doit permettre de retrouver une valeur "odorValue" même aux points non échantillonnés et ceci en utilisant les coordonnées (x,y) de la localisation où nous voulons observer le phénomène. C'est ici qu'intervient le "FieldMDX" que nous présentons dans la section suivante et qui nous permet d'interroger le modèle logique que nous présentons puis de récupérer des valeurs estimées aux zones non-disponibles dans l'ED.



Time

Id	Year	Month	Day	Hour	Minute
1	2014	12	12	10	15
2	2014	12	13	9	00

Source

Source	Name
1	Usine
2	Déchets

Tracer

Tracer	Name
1	NO2
2	CO2

RegularGrid

idPoint	Geometry
1	POINT('0 1')
2	POINT('1 0')

c)

Fig 4. 6 a) Modèle logique générique avec une dimension de type champ continu incomplet « RegularGrid », b) instance du modèle logique générique, c) instance des tables

4.3.3 FieldMDX

GeoMDX est une extension du langage MDX qui fournit des opérateurs spatiaux (ex. intersect, union, etc) susceptibles d'améliorer l'analyse spatiale. Par exemple, le slice spatiale

permet de d’effectuer un slice sur un cube de données spatiales en utilisant des prédicats spatiaux (ex. distance). Cependant, dans certaines situations, nous avons besoin d'utiliser des données ou des fonctions qui ne sont pas prises en charge dans les requêtes MDX. A cet effet, MDX peut être étendu en utilisant les fonctions définies par l'utilisateur (UDF). Par exemple, nous pourrions vouloir utiliser une opération qui n'est pas gérée par MDX ou une donnée qui n'existe pas dans l'entrepôt de données. Les fonctions définies par l'utilisateur peuvent être utilisées pour ajouter cette opération ou cette donnée au cube OLAP. Les UDFs sont un moyen puissant d'améliorer les capacités de MDX pour une application particulière.

Comme présenté dans les sections suivantes, nous avons exploité cette propriété du langage MDX pour inclure de nouveaux opérateurs (interpolation), ce qui nous permettra de générer la continuité et améliorer la résolution d’un champ continu incomplet dans le système SOLAP. La représentation du champ continu incomplet tel que présentée dans le modèle logique (figure 4.6) permet l'utilisation des opérateurs map algebra dans des requêtes MDX (agrégation point par point) de la manière suivante:

Requête 1: *La valeur moyenne de l’odeur à chaque membre du champ continu incomplet pour l’année 2012*

```
SELECT [RegularGrid].[idPoint].Members ON ROWS, {[time].[2012]} ON COLUMNS
FROM [odorCube] WHERE [Measures].[odorValue]
```

Pour avoir une représentation continue de la dimension de type champ continu incomplet, nous avons défini le FieldMDX comme une extension de MDX avec de nouvelles fonctions définies par l'utilisateur qui gèrent l'interpolation spatiale tel que:

```
NumericType InterpolatePoint (geometry)
```

La fonction "InterpolatePoint" prend en entrée une géométrie (point (x, y)) et retourne une valeur numérique. Cette valeur est une mesure dérivée dans le modèle de données OLAP qui est estimée en utilisant les valeurs du voisinage du point donné en entrée. Notez que le voisinage est défini différemment selon l’implémentation de la fonction abstraite "InterpolatePoint" (bilinéaire, bicubique, krigeage ...).

Comme indiqué plus haut, nous avons choisi de représenter les points composant la grille régulière avec des membres spatiaux classiques et leurs valeurs avec des mesures numériques. Par exemple, supposons que nous voulons récupérer une valeur à une localisation non échantillonnée du champ continu (un point qui n'existe pas dans la table "RegularGrid") dont la géométrie est définie par les coordonnées (42,3521 -72,1235). Pour répondre à ce besoin en utilisant l'interpolation bilinéaire, les décideurs doivent simplement utiliser une fonction GeoMDX telle que InterpolatePointBilinear (POINT (72,1235 42,3521)). Cette technique permet aux utilisateurs de retrouver une valeur à chaque localisation du domaine spatial étudié de manière transparente, comme si elle existait déjà dans l'entrepôt de données spatial.

Ainsi, la fonction proposée permet de récupérer les voisins du point donné en paramètre depuis le niveau champ continu incomplet ([RegularGrid]. [idPoint]), sur la base de la distance (st_distance de GeoMDX), puis d'extraire leurs valeurs depuis la table de faits et enfin d'utiliser ces valeurs pour calculer la valeur à la localisation du point en paramètre. Ceci est un exemple d'une requête en utilisant une implémentation de la fonction "InterpolatePoint" avec une interpolation bilinéaire:

Requête 2: *La valeur moyenne de l'odeur aux coordonnées (721148 3140020) pour l'année 2012*

```
With member [Measures].[odorValue] as
'InterpolatePointBilinear(ST\_GeomFromText("POINT (721148
3140020)"))'
SELECT [Measures].[odorValue]ON ROWS,[time].[2012] ON COLUMNS FROM
[odorCube]
```

Dans cette section nous avons répondu aux pré-requis concernant la représentation continue des grilles régulières dans le SOLAP. Cette représentation supporte les opérateurs map algebra dans le cas où l'utilisateur demande une estimation agrégée de la valeur du phénomène à une localisation non-échantillonnée, celle-ci est calculée en effectuant une agrégation map algebra sur les voisins de cette localisation.

La prochaine section est dédiée à l'exploitation des champs continus incomplets représentés par des grilles régulières de points sous différentes résolutions, ce qui permet d'améliorer le niveau de détails de celles-ci.

4.4 La multi-résolution

Les échelles et les résolutions ont toujours été des questions clés dans le domaine de la géographie. Le développement rapide des systèmes d'information géographiques (SIG) au cours des dernières années a poussé à une reconsidération de ces concepts. La multi-résolution est une approche qui consiste à définir plusieurs niveaux de résolutions pour améliorer ou généraliser le rendu des requêtes soumises par l'utilisateur. Cette section se concentre sur l'adaptation de la résolution d'un champ continu représenté par une grille régulière de points, au niveau de résolution choisie par l'utilisateur. L'idée principale est d'exploiter les capacités du "FieldMDX", en utilisant les fonctions d'interpolation pour générer des niveaux de détails élevés à partir du niveau initial. Il est très important de noter que, généralement, pour chaque phénomène spatial un ensemble de résolutions utiles et connues existent, afin qu'elles puissent être prédéfinies en fonction des données et des besoins des utilisateurs. Pour une analyse effective des données, le système OLAP doit être préparé à répondre à tous les besoins définis préalablement par le décideur. Ainsi, le système doit disposer de tous les membres des différentes résolutions qui représenteront le champ continu. Puisque chaque nouveau point créé pour améliorer la résolution a un ensemble de voisins qui appartiennent à la résolution originale et qui sont utilisés dans le calcul de sa valeur, nous avons effectué la modélisation de cette relation de voisinage ainsi que les règles qui nous permettent de naviguer d'un niveau de résolution à un autre. Nous proposons une extension du modèle de données OLAP pour gérer la multi-résolution des champs continus incomplets représentés par des grilles régulières de points.

4.4.1 Modèle conceptuel

Dans notre extension, nous avons défini un niveau "FieldIncomplete" (champ continu incomplet) comme un type spécial de niveau spatial qui dispose d'une fonction d'interpolation pour représenter la continuité ("ContinuityInterpolation"), le niveau de résolution auquel il appartient ("résolution") et une relation de voisinage ("NeighborhoodRelationship") qui dépend de la méthode d'interpolation utilisée (cf. section 4.3.1).

En effet, chaque fonction d'interpolation spatiale utilise un nombre de voisins différent, (par exemple, une fonction bilinéaire utilise une grille de 2 * 2), pour estimer une valeur à un point donné. Dans notre approche, la navigation d'un niveau champ continu incomplet à un autre

(multi-résolution) n'implique pas nécessairement l'utilisation d'une opération d'agrégation, mais plutôt l'utilisation d'une interpolation ("ContinuityInterpolation").

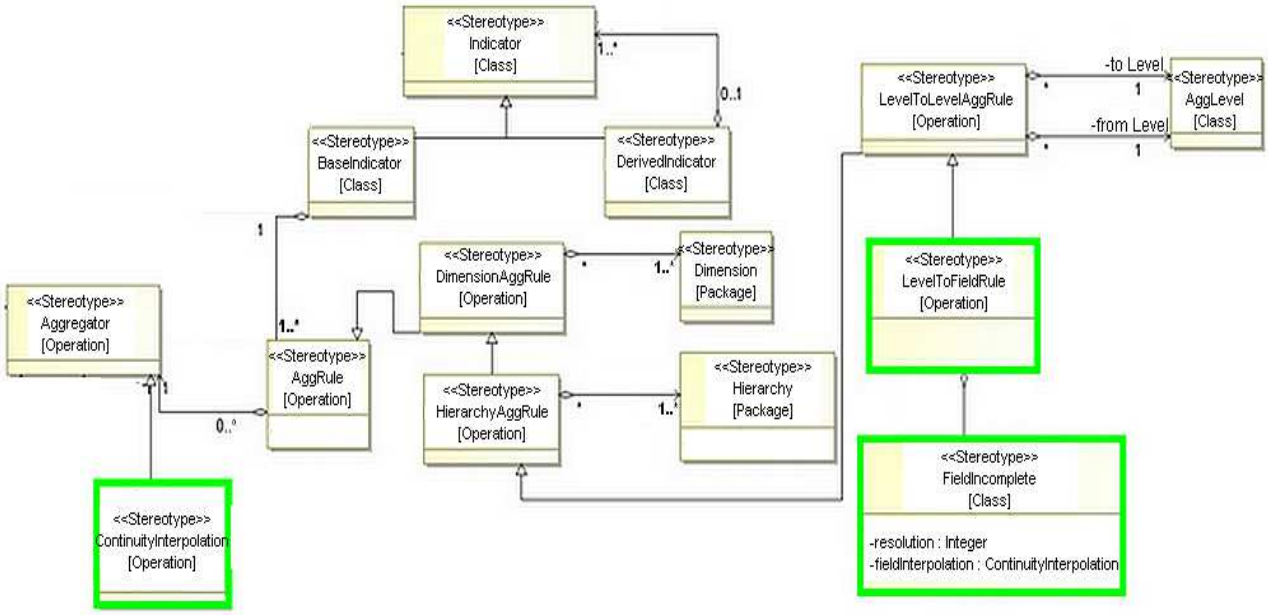


Fig 4. 7 Extension du profil UML de (Boulil 2012) avec les champs continus incomplets à multi-résolution

Ainsi, nous avons redéfini les règles d'agrégation entre les niveaux proposées par (Boulil 2012) (figure 4.7) en définissant le stéréotype "LevelToFieldRule" comme une extension de la relation "LevelToLevelAggRule" pour les niveaux de type "FieldIncomplete". En effet, la représentation conceptuelle de l'agrégation d'une mesure utilise un "BaseIndicator" qui est associé à une mesure ("SMDAttribute"), ainsi qu'une fonction d'agrégation ("Aggregator") et la dimension sur laquelle l'agrégation est appliquée ("DimensionAggRule").

Comme le montre la figure 4.7, nous avons étendu ce modèle comme défini précédemment pour y ajouter la nouvelle opération "ContinuityInterpolation", qui hérite de la classe "Aggregator" et qui sert de règle lors de la navigation entre les niveaux de type "FieldIncomplete".

Une instance du profil pour notre cas d'étude est illustrée dans la figure 4.8. La dimension spatiale est étendue avec trois niveaux de résolution (Grid100, Grid200 et Grid400) de type "FieldIncomplete", composés chacun d'un ensemble de géométries (Points), d'un niveau de résolution et d'une fonction d'interpolation (FieldInterpolation). Ainsi, chaque niveau "FieldIncomplete" à résolution élevée (ex. Grid200 et Grid400) peut utiliser une fonction d'interpolation différente pour estimer ses valeurs depuis le niveau "FieldIncomplete" à la résolution initiale (ex. Grid100). Cela nous permet d'adapter la méthode d'interpolation à utiliser au nombre de membres qui constituent le niveau. Si le niveau de résolution est

composé d'un grand nombre de membres, par exemple, nous pouvons choisir une fonction d'interpolation moins complexe (ex. la méthode du plus proche voisin).

L'indicateur "OdorIndicator" est formulé de manière à indiquer la règle à utiliser pour la navigation dans une hiérarchie de résolutions de champs continus incomplets (ex. interpolation bilinéaire) ou la navigation dans les hiérarchies non spatiales (ex. agrégation par la moyenne).

Lors de la navigation dans les niveaux d'une hiérarchie de type champ continu incomplet ("Field Incomplete"), la règle qui est appliquée implique l'interpolation ("ResolutionChange" indique que l'opération effectuée lors de l'exploration des niveaux continus, est l'interpolation bilinéaire). Cette opération va utiliser les relations de voisinage existantes entre les niveaux de résolutions plus élevées et le premier niveau de résolution (le plus grossier) pour estimer des valeurs aux points constituants les résolutions élevées. Cette relation de voisinage est définie selon la fonction d'interpolation utilisée. Par exemple, avec une interpolation bilinéaire, la relation de voisinage relie chaque point d'un niveau de résolution élevé à ses 2 ou 4 plus proches voisins au niveau de résolution original (le plus grossier).

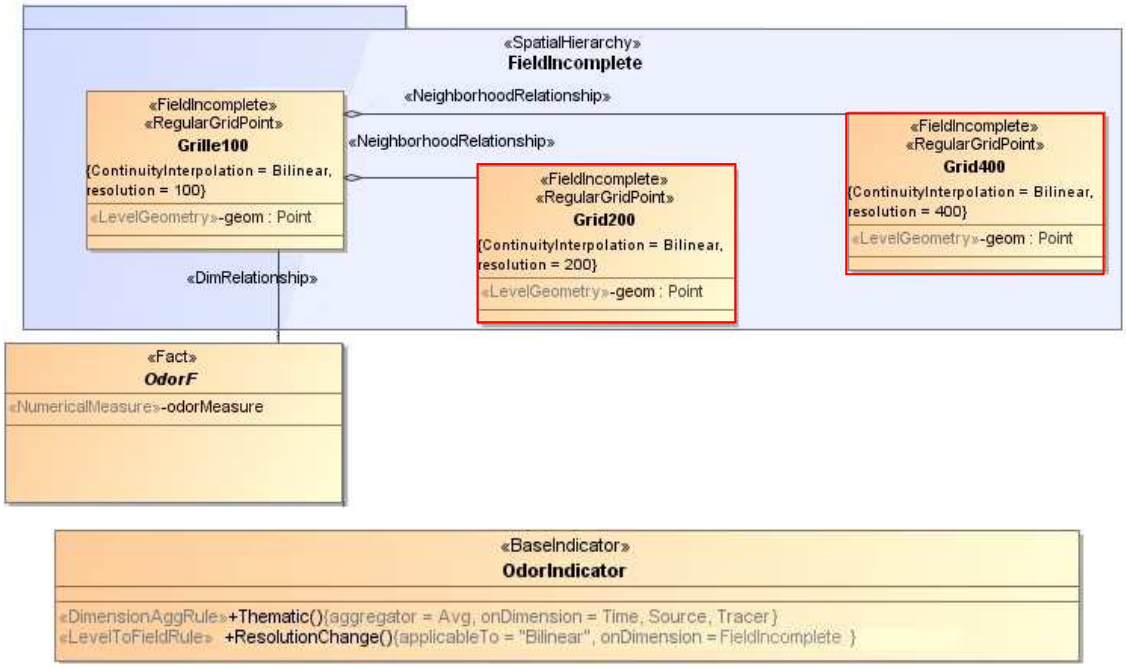


Fig 4. 8 Extension du modèle d'agrégation de (Boulil 2012) pour les champs continus incomplets à multi-résolution

Cette approche est souvent utilisée dans les systèmes d'information géographique, elle constitue un moyen de générer de l'information dans les zones non-échantillonnées, puisqu'il est impossible de recueillir des données à toutes les localisations de l'espace. L'avantage de l'utilisation de cette approche est que la classe qui représente les faits observés

("odorMeasure"), est liée seulement au niveau de résolution le moins fin, donc celui qui est composé du nombre minimum de membres. Ainsi, nous pouvons stocker un plus petit nombre de faits, que nous pouvons considérer comme un échantillon, puis les méthodes d'interpolation ("ContinuityInterpolation") correspondantes à chaque niveau de résolution élevée se chargent d'estimer le reste des valeurs.

4.4.2 Approche d'agrégation Field et approche d'interpolation Field

En fonction du type d'analyses à effectuer par l'utilisateur, une résolution plus ou moins détaillée peut être requise. Bien que conceptuellement, nous avons défini le processus d'interpolation comme une règle lors de la navigation dans la hiérarchie de la dimension de type « FieldIncomplete », cette règle peut être évitée si les valeurs des membres aux résolutions élevées sont pré calculées. En effet, il existe deux approches principales au niveau logique de la modélisation, pour fournir à l'utilisateur un champ continu à multi-résolution :
i) *l'approche à la volée*, dans laquelle différentes représentations à résolutions élevées sont calculées en temps réel à partir d'une représentation de base puis transmis au client en réponse à une requête et ii) *l'approche de pré-calcul*, dans laquelle des représentations à différentes résolutions sont pré-calculées et stockées dans le serveur en vue de fournir au client le niveau de résolution adéquat à chaque requête (Follin et Bouju 2007).
Ceci permet une navigation classique dans la dimension "FieldIncomplete", allant de la résolution la plus détaillée à la moins détaillé. Ainsi, pour modéliser un champ continu incomplet représenté par une grille régulière de points avec plusieurs résolutions dans un modèle multidimensionnel, nous proposons deux approches au niveau logique (Zaamoune et al. 2013a), basées sur la modélisation classique en étoile. La première est l'approche *"Field Agregation Star Schema" (FASS).* Celle-ci est basée sur la structure des pyramides de résolutions (Jolion et Rosenfeld 1994) où toutes les mesures sont pré-calculées au niveau le plus détaillé de la dimension "FieldIncomplete" puis agrégées afin d'obtenir des résolutions de plus en plus globales (figure 4.9-a). La deuxième est l'approche *"Field Interpolation Star Schema" (FISS),* où seules les valeurs au niveau de résolution le moins détaillé sont stockées et où l'interpolation basée sur le voisinage est utilisée lors de la navigation entre les niveaux de la dimension "FieldIncomplete" (figure 4.9-b).
La modélisation en flocon, qui est une variante du modèle en étoile, peut aussi être utilisée pour modéliser les deux approches, mais celle-ci ne dispose pas de l'avantage de la dé-

normalisation qui permet une économie de jointures à l'interrogation et ainsi une optimisation des requêtes d'analyse.

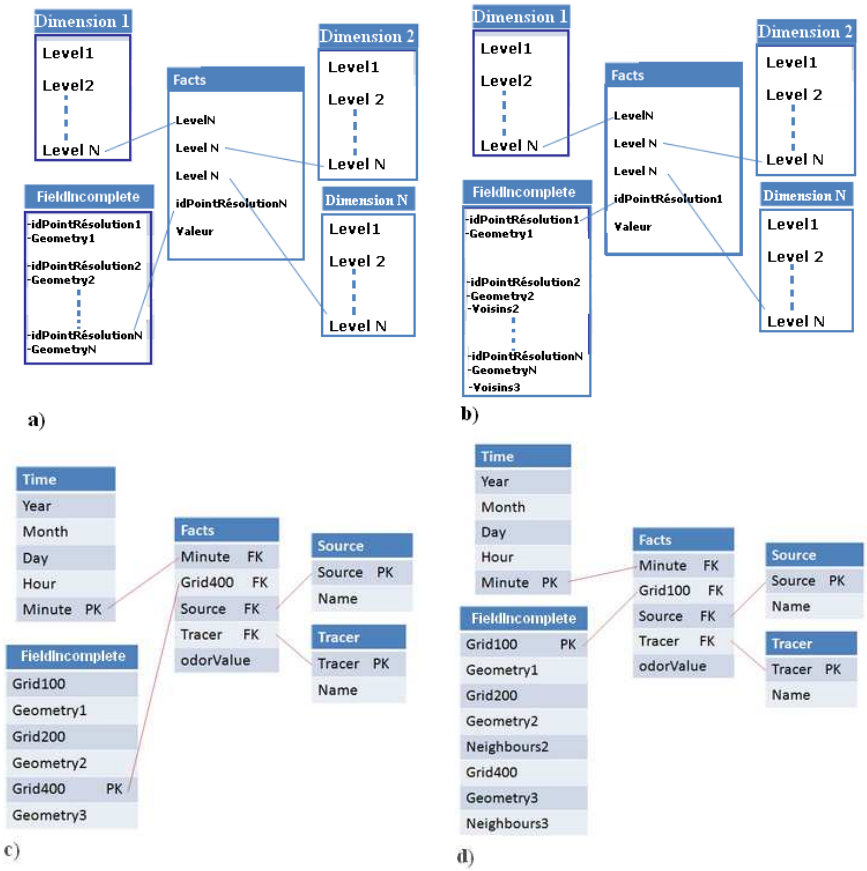


Fig 4. 9 a) « FASS » générique, b) « FISS » générique (Zaamoune et al. 2013a), c) instance du « FASS », d) instance du « FISS »

4.4.2.1 L'approche "Field Aggregation Star Schema" (FASS)

Le schéma en étoile figure 4.9-a illustre un schéma en étoile composé de N dimensions dont une de type "FieldIncomplete". Chaque dimension est composée de plusieurs niveaux de granularité. Les niveaux de la dimension "FieldIncomplete" représente les différentes résolutions des données. Chaque niveau dispose des identifiants de ses membres et de leurs géométries. La table de fait qui contient la mesure "Valeur" est reliée au niveau le plus détaillé de chaque dimension. En se basant sur ce modèle nous proposons une instance de celui-ci pour notre cas d'étude où la mesure "odorValue" représente la valeur du champ continu et où la dimension spatiale "FieldIncomplete" représente des niveaux du champ continu à différentes résolutions (figure 4.9-c). Ce modèle étend la dimension spatiale continue de la figure 4.6, avec deux niveaux supplémentaires, chacun représentant un niveau

de résolution différent ([FieldIncomplete].[Grid200] et [FieldIncomplete].[Grid400]). Chaque niveau de la dimension "FieldIncomplete" est constitué d'un identifiant (ex. Grid100) et d'une géométrie qui représente un point (ex. geometry1). La mesure "odorValue" est stockée au niveau le plus détaillé dans la table de faits. Ainsi, le décideur peut analyser les données spatiales à différentes résolutions dans la même session d'analyse MDX. L'approche FASS est classique et ne nécessite aucune extension du langage multidimensionnel MDX puisque la dimension de type champ continu incomplet est composée d'un ou plusieurs niveaux de résolution et toutes les valeurs sont stockées dans la table de faits. Il suffit de changer le niveau de la résolution dans la requête MDX pour modifier le niveau de détails du résultat. En utilisant cette approche, on précise simplement le niveau de résolution approprié du champ continu dans la requête MDX de la manière suivante:

Requête 3: *les valeurs moyennes de l'odeur aux points du champ continu à la résolution **400 * 400** pour l'année **2012***

```
SELECT [FieldIncomplete].[Grid400]. Members ON ROWS, {[time].[2012]}
ON COLUMNS FROM [odorCube]
WHERE [Measures].[odorValue]
```

Requête 4: *les valeurs moyennes de l'odeur aux points du champ continu à la résolution **200 * 200** pour l'année **2012***

```
SELECT [FieldIncomplete].[Grid200]. Members ON ROWS, {[time].[2012]}
ON COLUMNS FROM [odorCube]
WHERE [Measures].[odorValue]
```

4.4.2.2 L'approche "Field Interpolation Star Schema" (FISS)

4.4.2.2.1 Modèle logique

Comme indiqué dans la section 2.3.2.3, les méthodes d'interpolation spatiale sont souvent utilisées pour améliorer la résolution d'un champ continu incomplet. Nous proposons donc une variante du schéma proposé précédemment (FASS), afin de mettre à disposition de l'utilisateur plusieurs niveaux de résolution. Ceci en reliant la table de fait au niveau de

résolution le moins détaillé de la dimension spatiale, comme le montre le modèle générique figure 4.9-b et son instance figure 4.9-d. Dans notre approche, le calcul des valeurs des points ajoutés à la résolution d'origine pour l'affiner implique l'application d'une interpolation spatiale. L'interpolation spatiale utilise le voisinage d'un point donné pour estimer sa valeur. Puisque les membres des résolutions élevées peuvent être définis à l'avance (idPointRésolution2 et idPointRésolution3), leurs voisins au niveau « idPointRésolution1 » sont aussi défini dans le modèle préalablement (voisins2 et voisins3).

4.4.2.2.2 FieldMDX

Pour implémenter le modèle proposé (figure 4.9-b), nous définissons une fonction GeoMDX de la même manière que celle définie dans la section 4.3.3 :

```
Numeric-type InterpolateResolution (FieldIncomplete Member)
```

Cependant, la fonction "InterpolateResolution" reçoit en paramètre un membre d'un niveau de type "FieldIncomplete" au lieu d'une géométrie, puis retourne une valeur interpolée de ce membre.

La fonction "InterpolateResolution" peut être implémentée de différentes manières, en fonction de la fonction d'interpolation à utiliser. Nous avons implémenté cette fonction avec une interpolation bilinéaire en utilisant les valeurs des 4 voisins qui entourent le point à estimer et avec une interpolation bicubique qui utilise les valeurs de 16 voisins. Cela permet de trouver les valeurs de tous les membres du niveau "Grid400" (résolution 400 * 400) en utilisant leurs voisins dans le niveau "Grid100" (résolution 100 * 100) (Neighbors3). Ainsi, la requête 3 peut être réalisée tel que:

```
SELECT {[FieldIncomplete].[Grid400].Members} ON ROWS,
{[time].[2012]} ON COLUMNS FROM [odorCube]
Where [Measures].[EstimatedValue]
```

Notez que le système SOLAP utilise un fichier XML spécifique qui définit une balise XML spécifique pour chaque concept spatio-multidimensionnel (dimensions, mesures ...). Ainsi, dans le schéma XML SOLAP multidimensionnel, la fonction "InterpolateBilinear" est appelée, dans la formule de la mesure calculée "EstimatedValue", comme suit:

```
<CalculatedMember name="EstimatedValue" dimension="Measures"
formula = "InterpolateBilinear ([FieldIncomplete].[Grid400].CurrentMember">
</CalculatedMember>
```

L’utilisation d’une mesure calculée permet de récupérer les valeurs d’un phénomène selon différentes dimensions dans une échelle donnée de manière transparente pour les décideurs, de la même manière que pour une agrégation normal (SQL). Une autre approche serait de combiner les approches FASS et FISS. Ainsi, par exemple, pour des données au niveau de résolution (200 * 200), nous pouvons utiliser l'approche FASS pour l'agrégation spatiale (généralisation) et ainsi générer le niveau de résolution (100 * 100), ou utiliser l'approche FISS pour interpoler (désagrégation) et générer le niveau de résolution (400 * 400).

Notez que nous ne disposons pas des valeurs aux zones non-échantillonnées pour effectuer une comparaison. Les seules données disponibles sont celles à la résolution originale (par exemple, 100 * 100). Cependant, pour chaque méthode d'interpolation, il existe des méthodes pour calculer son taux d'erreur, mais nous ne nous focalisons pas sur cet aspect dans ce travail.

Dans cette section nous avons répondu aux pré-requis concernant la représentation à multi-résolution des grilles régulières dans le SOLAP. Nous avons démontré que le changement de résolution d’une grille régulière représentant un champ continu peut être effectué à la volée lors des analyses MDX grâce à de nouvelles fonctions "FieldMDX". Ce changement de résolution est soumis aux opérateurs Map Algebra ainsi qu’aux opérateurs spatiaux de GeoMDX (ex. slice spatial). Ceci permet d’utiliser les fonctions "FieldMDX" de manière transparente sans avoir à modifier les opérateurs spatiaux ou les règles d’agrégation lors de l’analyse d’un champ continu.

4.5 L’architecture relationnelle OLAP « FieldMDX »

Dans cette section, nous présentons l'architecture relationnelle SOLAP (Figure 4.10) pour la mise en œuvre de notre approche. L'idée principale consiste à utiliser des standards classiques afin de fournir une approche générique à notre proposition. Les données spatiales et alphanumériques sont stockées selon les modèles logiques présentés dans les sections 3 et 4 au niveau de l’ED spatial implémenté en utilisant PostGIS. PostGIS est un logiciel open source qui ajoute un support géographique aux bases de données PostgreSQL. En effet,

PostGIS spatialise le serveur PostgreSQL, lui permettant d'être utilisé comme une base de données d'un SIG.

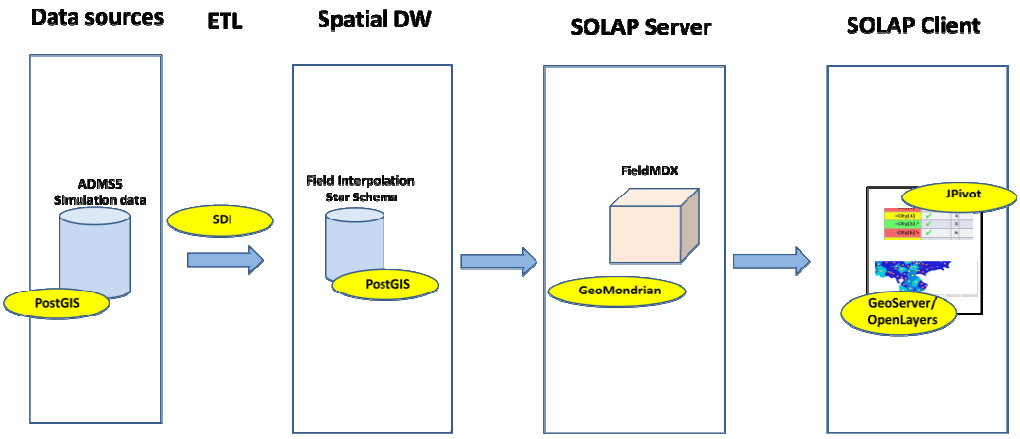
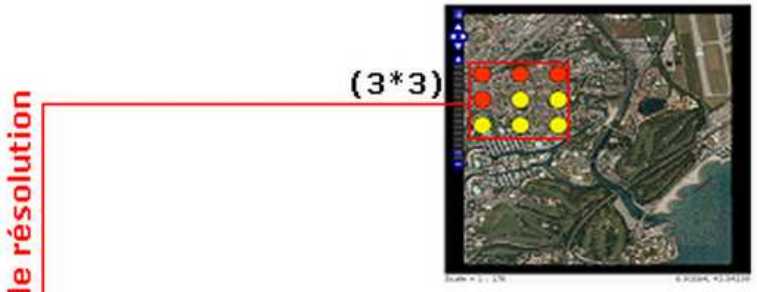


Fig 4. 10 Architecture FieldMDX

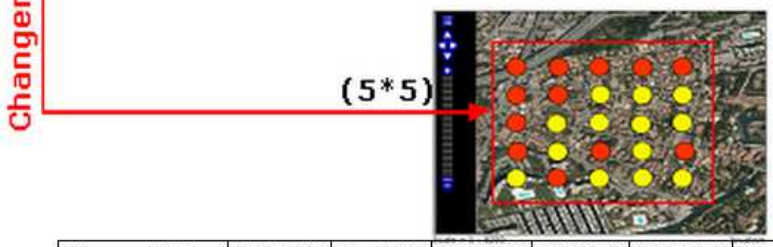
Nous utilisons GeoMondrian comme serveur SOLAP. GeoMondrian est une version Open Source de serveurs SOLAP, une version spatiale de "Pentaho Analysis Services" (Mondrian). GeoMondrian fournit une intégration cohérente des données spatiales vectorielles dans la structure de données du cube OLAP afin de définir des niveaux spatiaux. GeoMondrian implémente la structure logique du serveur SOLAP (par exemple, les cubes, les dimensions et les mesures) sur la base de PostGIS. Notez que généralement MDX permet de réaliser des fonctions définies par l'utilisateur dans plusieurs langages de programmation (par exemple, Java et. NET) en fonction du serveur OLAP utilisé. Dans ce travail, nous avons utilisé une implémentation Java dans GeoMondrian. Plus exactement, l'interpolation est effectuée à l'aide d'une API Java existante "api javax.media.jai» (JAI).

D'après nos tests expérimentaux que nous avons menés sur des clients SOLAP commerciaux, nous concluons qu'il n'y a pas de solution existante qui permet de visualiser un champ continu incomplet. Les clients existant ne sont pas assez performants pour la représentation cartographique du grand nombre de géométries contenues dans chaque dimension spatiale continue. Ainsi, nous utilisons un client ad hoc basé sur OpenLayers et Geoserver développé au sein de notre équipe. Nous utilisons GeoServer, un serveur web cartographique open source permettant aux utilisateurs de partager et modifier des données géospatiales et de créer des fichiers GML contenant les géométries des membres spatiaux. Les cartes thématiques sont créées manuellement au moyen d'un fichier SLD qui permet d'attribuer une couleur différente à chaque membre spatial selon un intervalle de valeurs.

Le résultat est présenté avec OpenLayers dans la figure 4.11. Les représentations tabulaires et graphiques sont fournies avec JPivot, qui est un client web OLAP fourni avec (Geo) Mondrian.



[Measures].[Value]	[Field].[p01]	[Field].[p11]	[Field].[p21]	[Field].[p31]	[Field].[p41]	[Field].[p51]	[Field].[p61]	[Field].[p71]	[Field].[p81]
[Time].[2013]	74.2	48.3	49.3	41.7	31.9	47.7	41.9	44.2	61.7



[Measures].[Value]	[Field].[p0]	[Field].[p1]	[Field].[p2]	[Field].[p3]	[Field].[p4]	[Field].[p5]	[Field].[p6]	[Field].[p7]	[Field].[p8]
[Time].[2013]	48.8	48.6	26.5	41.5	57.4	54.5	44.6	46.6	39.5

[Field].[p9]	[Field].[p10]	[Field].[p11]	[Field].[p12]	[Field].[p13]	[Field].[p14]	[Field].[p15]
32.8	49.8	47	36.4	51.1	44	52.3

Fig 4. 11 Visualisation d'un champ continu incomplet avec 2 niveaux de résolution dans OpenLayers et JPivot

4.6 Expérimentations

Dans cette section, nous montrons les performances des deux approches (*FASS* et *FISS*) proposées dans la section 4.4.2 en termes de stockage et de temps de calcul. La taille de la base de données utilisée pour les expérimentations est de 1,17 GB.

L'ordinateur utilisé a la configuration suivante: processeur Intel Core i3 2.20 GHz, 4 Go de RAM, système d'exploitation Windows 7 Professional OS 64 bits.

Pour tester notre proposition, nous avons défini différents cas où les dimensions spatiales de type champ continu incomplet contiennent respectivement: un niveau de résolution 100 * 100, deux niveaux de résolution 100 * 100 et 200 * 200 et enfin, trois niveaux de résolution 100 * 100, 200 * 200 et 400 * 400. Nous avons également fait varier la méthode d'interpolation pour comprendre l'impact du nombre de voisins utilisés dans l'interpolation sur les performances obtenues.

4.6.1 Le stockage

Pour les performances de stockage, la figure 4.12 montre la taille de la table de faits mesurée selon le nombre de membres spatiaux et temporelles (la résolution spatiale la plus fine / la granularité temporelle la plus fine) en utilisant les deux approches (FASS et FISS). Nous pouvons facilement voir deux différences importantes entre les deux approches : i) l'approche FASS est plus coûteuse en termes de stockage que l'approche FISS, parce que dans cette dernière les faits ne sont stockés qu'au niveau de résolution spatiale le moins fin et ii) dans l'approche FISS, la taille de la table de faits varie uniquement en fonction de la taille des dimensions non-spatiales. Ainsi, même en augmentant la taille de la dimension spatiale (par exemple, en ajoutant un niveau de résolution), la table de fait ne change pas de taille, car elle ne contient que les valeurs au premier niveau de résolution.

4.6.2 Temps d'exécution

En ce qui concerne la continuité, nous avons implémenté la fonction "InterpolatePoint" de deux manières différentes, en utilisant une interpolation bilinéaire et une interpolation bicubique. D'autre part, pour évaluer les performances de temps d'exécution liées au changement de résolution, nous avons exécuté les requêtes mentionnées ci-dessus, où nous avons combiné des opérations d'agrégation (roll-up) sur les dimensions non-spatiales et des opérateurs spatiaux (slice) sur un champ continu incomplet à différentes résolutions. La figure 4.13 montre le temps d'exécution de la requête 3, qui consiste à générer les valeurs des membres d'un champ continu à des résolutions différentes, en utilisant des dimensions temporelles à différentes tailles. Cette figure montre une certaine approximation dans le temps d'exécution entre les deux approches (FISS et FASS) à un certain niveau. Au-delà de ce niveau, on remarque que l'écart se creuse considérablement. Ainsi, la minimisation du stockage et donc du nombre de jointures entre la table de faits et les dimensions, a permis à la méthode FISS que nous proposons d'avoir de meilleurs temps d'exécution que l'approche FASS à tous les niveaux de résolution (100 * 100, 200 * 200 et 400 * 400). La figure 4.13 montre qu'avec l'approche FASS, le temps d'exécution augmente avec le nombre de membres spatiaux et non-spatiaux, tandis que dans l'approche FISS, il augmente sensiblement selon les tailles des dimensions non-spatiales. En effet, la taille de la dimension spatiale n'a pas

beaucoup d'influence sur les performances, car il n'ya pas de lien direct entre la table de faits et les membres qui appartiennent aux résolutions élevées.

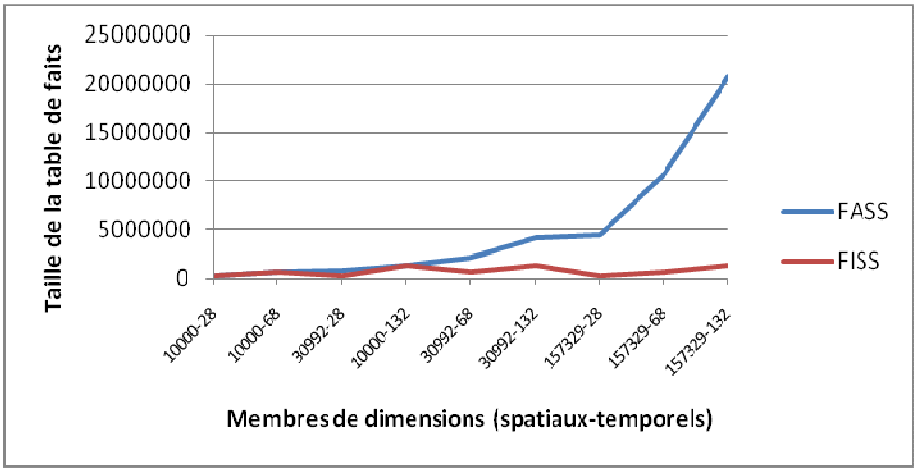


Fig 4. 12 Taille de la table de faits (FASS et FISS)

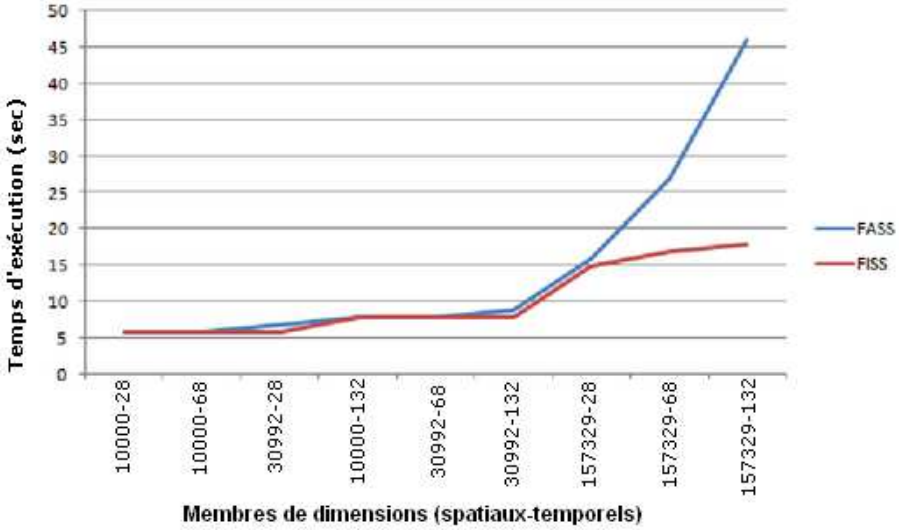


Fig 4. 13 Temps d'exécution de la requête 3 avec l'approche FASS et l'approche FISS

4.6.2.1 Temps d'exécution (FISS bilinéaire, FISS bicubique et FASS)

Pour démontrer l'impact du nombre de voisins utilisés dans le procédé d'interpolation sur les performances, la figure 4.14 montre les performances obtenues pour la même requête présentée précédemment avec l'approche FISS implémentée de deux manières différentes, en utilisant une interpolation bilinéaire comme dans l'expérience précédente (4 voisins) et en utilisant une interpolation bicubique (16 voisins). Cette expérience montre qu'il existe une constante ε, entre les performances liées à l'utilisation de l'interpolation bicubique et celles

liées à l'utilisation de l'interpolation bilinéaire, qui dépend de la fonction d'interpolation utilisée tel que:

$$Perf_{Bicubique} = Perf_{Bilinéaire} + \varepsilon \text{ avec } NbVoisins_{Bilinéaire} < NbVoisins_{Bicubique}$$

Dans notre cas, la constante ε = 500 ms. Par conséquent, la fonction bicubique a besoin de 500 ms de plus que la fonction bilinéaire pour récupérer les voisins et calculer une estimation en utilisant leurs valeurs.

Ainsi, une fonction d'interpolation qui utilise un grand nombre de voisins implique un ε plus grand, ce qui signifie que les performances seront plus coûteuses.

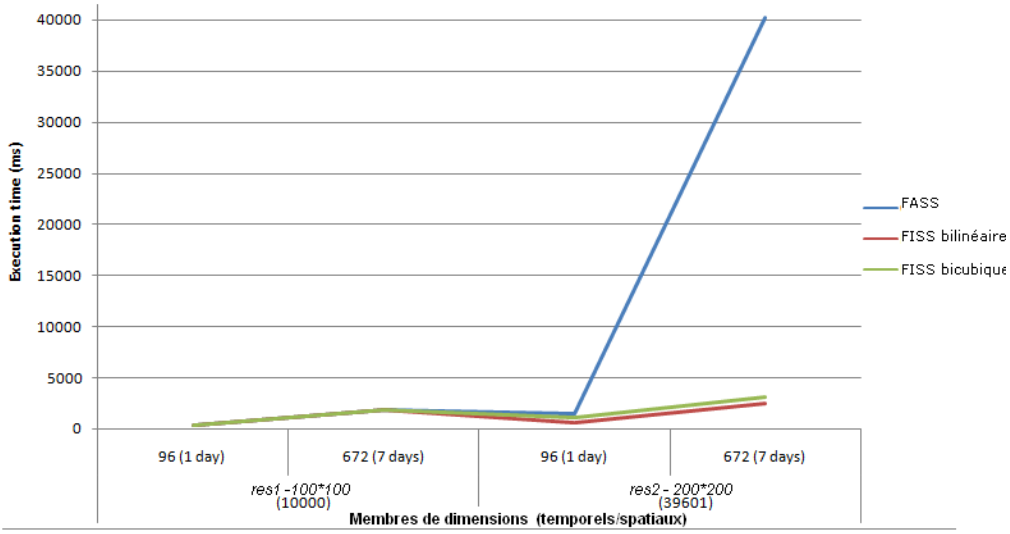


Fig 4. 14 Temps d'exécution de la requête 3 avec l'approche FISS (bilinéaire et bicubique) et l'approche FASS

4.7 Conclusion

Dans cet article, nous présentons un modèle multidimensionnel pour les champs continus incomplets à plusieurs résolutions. Ce modèle est implémenté dans une architecture ROLAP basée sur des standards (SQL et MDX). Nous proposons une approche basée sur l'interpolation pour générer la continuité des champs continus incomplets et nous présentons deux approches basées sur le schéma en étoile, FASS et FISS, pour générer et améliorer les résolutions du champ continu dans le serveur SOLAP. Les objectifs de l'approche FISS sont d'améliorer le niveau de détail (LOD) du champ continu incomplet tout en optimisant les performances, en se basant uniquement sur la modélisation des données. Nous terminons ce chapitre avec des expérimentations qui démontrent la faisabilité et les performances liées à l'utilisation des deux approches (FASS et FISS) en utilisant deux méthodes d'interpolation différentes (bilinéaire et bicubique). Nous avons également démontré que le nombre de

voisins utilisés dans le procédé d'interpolation a un impact sur les performances. Ainsi le choix de la technique d’interpolation à utiliser doit être fait selon le type de données à analyser et le type d’analyses à effectuer.

Chapitre 5. Approximation des opérations d'agrégation des grilles régulières de points par l'échantillonnage dans le SOLAP

5.1 Introduction

Comme appliqué pour l'optimisation des opérations d'agrégations en SQL (Kang et al. 2013), nos travaux se basent sur le principe de l'échantillonnage pour optimiser les opérations d'agrégation dans le système SOLAP. Dans ce chapitre, nous présentons la problématique liée à l'agrégation des champs continus représentés par des grilles régulières de points dans le SOLAP. Cette problématique est traitée selon les critères définis dans la section 3.2.1. Nous montrons comment la technique d'échantillonnage, en particulier le « Cluster Sampling », peut être intégrée à un système SOLAP d'une manière générale dans le but de réduire le temps d'exécution des requêtes impliquant un très grand nombre de données. Bien qu'il ait été prouvé que ces techniques permettent effectivement d'améliorer considérablement les performances de stockage et de temps d'exécution des opérations sur de grands volumes de données (Das 2009), l'intégration de ces techniques dans une architecture ROLAP n'a pas été réalisée.

Dans la section 5.2 nous présentons la problématique et l'approche proposée, ainsi que l'architecture ROLAP adoptée. La section 5.3 présente la méthodologie qui englobe les 2 étapes les plus importantes du processus d'optimisation de l'agrégation par le clustering : *(1) la réduction des faits par l'échantillonnage et (2) la modélisation conceptuelle du ClusterCube*.

Le ClusterCube est cube SOLAP adapté pour approximer les résultats des agrégations en se basant sur les clusters obtenus dans la première étape. Le modèle logique proposé pour l'intégration des données clustérisées à l'entrepôt de données et au serveur SOLAP, ainsi que les expérimentations réalisées, sont présentés dans la section 5.4. Nous présentons dans la section 5.5 le modèle conceptuel du ClusterCube&Field, un cube SOLAP qui combine les capacités de l'analyse des champs continus présentées dans le chapitre 4 et celles du ClusterCube. Enfin, dans la section 5.6 nous concluons avec un bilan général.

5.2 Problématique et méthodologie

5.2.1 Cas d'étude et motivation

Dans ce chapitre, nous allons nous baser sur le cas d'étude fourni par la société Agaetis pour présenter la problématique liée à l'agrégation des valeurs d'un champ continu dans le système SOLAP. La figure 5.1 montre le modèle de notre entrepôt de données de grilles régulières de points composé d'une dimension spatiale « GrilleDimension » avec un niveau « Grille500 » de type « RegularGridPoint » (cf. chapitre 4) qui représente une grille régulière de résolution 500*500, d'une dimension temporelle « Temps » avec 3 niveaux de granularité (année, mois et jour) et du fait « OdorF » dont la mesure « odorMeasure » correspond à la valeur d'air pollué. En utilisant ce modèle, nous pouvons, par exemple, effectuer une requête d'agrégation qui calcule la valeur agrégée de chaque point de la grille pour une année donnée. Prenons comme exemple une requête qui consiste à agréger par la moyenne les valeurs des points pour l'année 2014. Celle-ci est alors calculée en utilisant de nombreuses liaisons entre la table de faits et les membres des dimensions pour retrouver les valeurs impliquées dans la requête demandée (500*500*365= 91250000 valeurs).

En effet, comme représenté dans le « BaseIndicator », l'agrégation de la mesure « odorMeasure » est effectuée en utilisant la moyenne (aggregator=Avg) sur les valeurs liées aux membres des dimensions « GrilleDimension » et « Temps ». Ceci est une agrégation map algebra locale (Tomlin 1990), qui consiste en une agrégation point par point des grilles en entrée pour calculer une grille agrégée en sortie comme décrit dans le chapitre 2 (section 2.3.3).

Pour obtenir le résultat (exact), cette agrégation utilisera l'ensemble des valeurs des points de la grille aux différents jours (Grille 1, Grille 2 et Grille 3) dans son calcul pour générer une seule valeur agrégée sur chaque point.

Comme le montre la figure 5.2, l'agrégation locale au point p1 consiste à récupérer ses valeurs aux jours t1, t2 et t3, puis d'en faire la moyenne pour calculer sa valeur agrégée en sortie. Dans le cas d'une agrégation d'une grille spatiale à résolution élevée sur plusieurs jours (ex. des années), les temps de calcul peuvent être très couteux.

Ainsi dans la prochaine section, nous présentons le principe général de notre approche qui vise à réduire le temps de calcul des opérations d'agrégation sur les données volumineuses, en nous basant sur l'approximation. Celle-ci permettra de réduire le nombre de valeurs

nécessaires pour effectuer l'agrégation en n'utilisant qu'un échantillon de celles-ci dans le calcul.

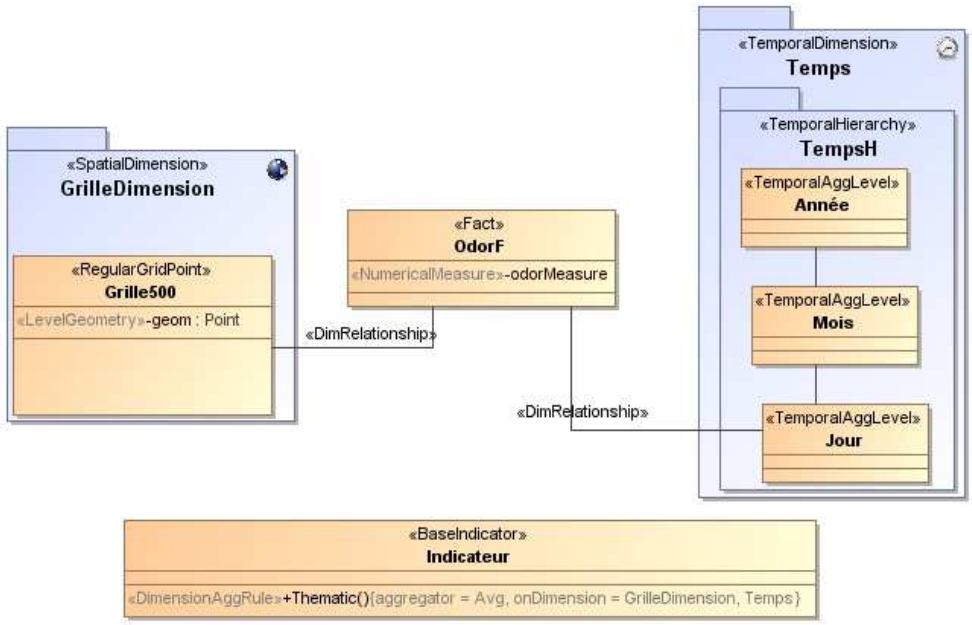


Fig 5. 1 Modèle de l'entrepôt de données pour l'analyse d'un champ continu incomplet représenté par une grille régulière de points

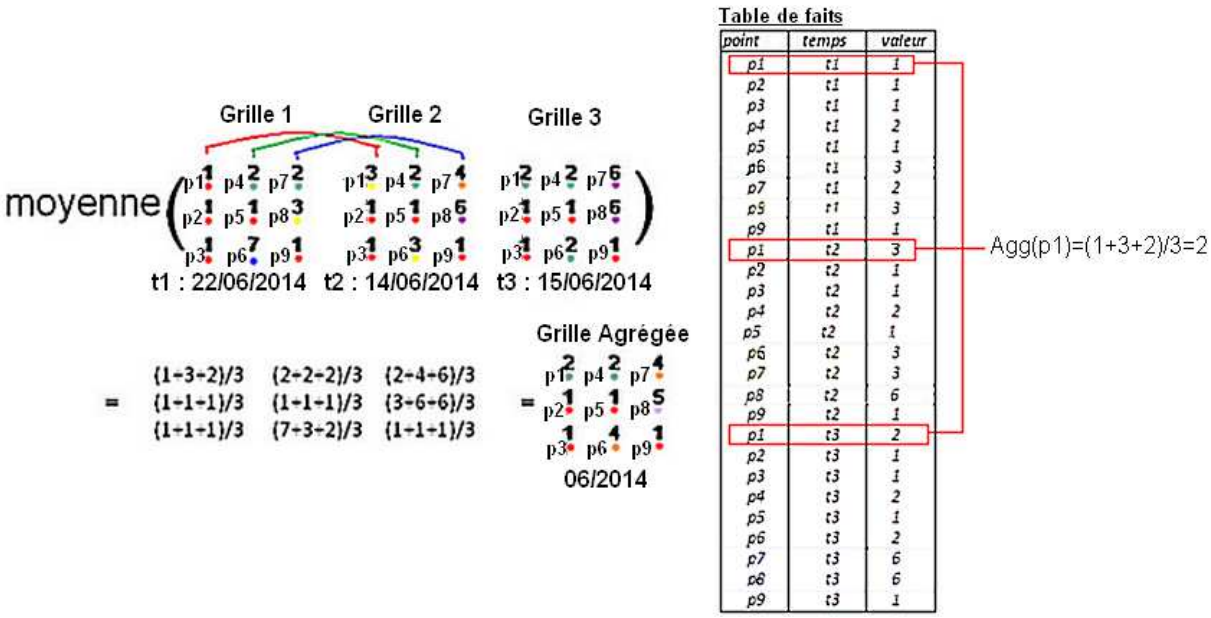


Fig 5. 2 Agrégation par la moyenne de 3 grilles régulières de points

5.2.2 L'approche proposée

L'approche que nous proposons permet d'exploiter la similarité entre les faits pour les classifier dans des groupes inter-similaires, en utilisant la méthode d'échantillonnage du cluster sampling (cf. section 3.1), afin de réduire le nombre de faits qui sont impliqués dans

l'exécution d'une requête d'agrégation. Cette approche permet de réduire le nombre de faits requis par le processus d'agrégation pour assurer un temps d'exécution adéquat au dépend d'une perte de précision paramétrable.

Nous tenons à préciser que cette approche est utilisable avec d'autres techniques d'approximation différentes du clustering. En effet, bien que la technique d'approximation utilisée influence sur la nature et le volume des échantillons générés, l'exploitation de ces échantillons dans une architecture ROLAP selon la méthodologie que nous proposons est la même. La figure 5.3 illustre le principe général de notre approche.

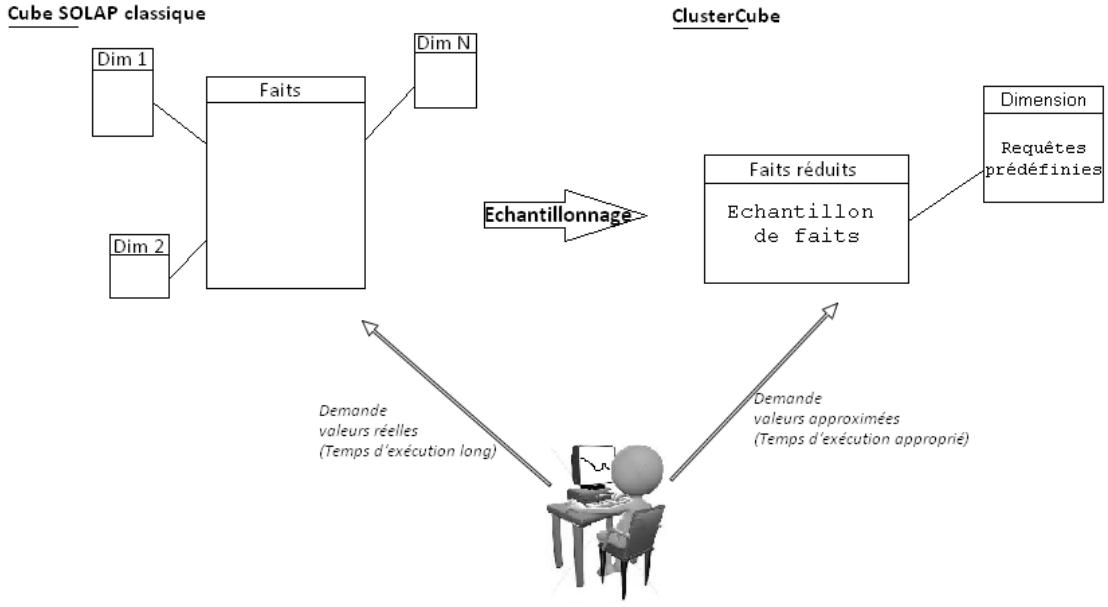


Fig 5. 3 Principe général de l'optimisation des agrégations via l'échantillonnage

Comme le montre la figure 5.3, notre méthode consiste à appliquer une méthode d'échantillonnage sur les faits d'un cube classique. Nous proposons d'utiliser un entrepôt de données réduit qui ne garde qu'un échantillon des faits du cube original. Cet échantillon de données (c'est-à-dire ce cube réduit) pourra être utilisé pour calculer les résultats des requêtes d'agrégation ; les résultats de ces requêtes seront donc une estimation du résultat réel. L'objectif de cette méthode est de donner la possibilité au décideur de rediriger ses requêtes d'agrégation vers le cube classique ou le cube réduit (ClusterCube) selon le temps d'exécution toléré et le degré de précision des résultats désiré.

L'avantage de l'approche que nous proposons est qu'elle permet de garder les données réelles à disposition de l'utilisateur pour qu'il puisse demander des analyses précises lorsque l'approximation n'est pas souhaitée. Ainsi, le ClusterCube est considéré comme un relais, dont le rôle est de fournir des réponses aux requêtes d'agrégations qui risquent de ralentir

l'analyse en cours. Il permet de garantir une analyse rapide en effectuant des approximations des résultats des agrégations d'une manière transparente pour l'utilisateur.
Dans les sections suivantes, nous présenterons avec plus de détails les différentes étapes qui permettent de : i) réduire le nombre de faits d'un cube classique en utilisant l'échantillonnage, puis de ii) modéliser le ClusterCube ainsi que les relations entre les requêtes à estimer et les échantillons à utiliser dans l'estimation et enfin iii) implémenter le ClusterCube dans une architecture ROLAP classique basée sur des standards tel que MDX et SQL.

5.2.3 Architecture proposée

Pour permettre une analyse parallèle des données échantillonnées et non échantillonnées dans une même architecture ROLAP, nous proposons une extension de celle-ci.
L'avantage de l'architecture ROLAP (qui repose sur des bases de données relationnelles) est qu'elle est bien standardisée (Buzydlowski, Song et Hassell 1998), ce qui permet d'utiliser notre extension quel que soit les tiers qui la composent (SGBD, serveur SOLAP, client SOLAP). Cette architecture représente une solution OLAP hybride (cf. section 2.4.5.1.3) qui permet de d'intégrer les fonctionnalités OLAP et SIG en vue d'une analyse spatio-multidimensionnelle adaptée. Ainsi, nous avons décidé de doter cette architecture de l'approche que nous proposons.
L'architecture proposée (figure 5.4) implique une phase de clustering qui est réalisée sur les données de l'entrepôt de données classique. Cette phase permet de déterminer quelles données sont choisies, parmi les données originales, pour servir d'échantillon dans le cube réduit.
Le stockage relationnel des données échantillonnées et non échantillonnées est effectué dans l'entrepôt de données. Ces données sont réparties sur deux modèles logiques dans l'entrepôt de données. Le premier modèle logique en étoile, « modèle logique classique », permet une analyse multidimensionnelle classique des données non échantillonnées et le deuxième modèle logique, « modèle logique ClusterCube », permet d'utiliser les échantillons de données.
Pour pouvoir utiliser ces modèles logiques via le serveur SOLAP, deux cubes sont crées, le « cube classique » et le « ClusterCube », chacun est lié au modèle logique qui lui correspond. Cependant pour que le passage d'un cube à un autre soit transparent pour l'utilisateur, l'architecture dispose du module « Switcher » (figure 5.5) qui permet d'éviter toute intervention de l'utilisateur lors de la redirection des requêtes d'agrégation vers l'un des deux

cubes. Ce module relie le client SOLAP au serveur SOLAP et se charge de comparer la requête émise par l'utilisateur avec celles présentes dans un modèle de coûts avant de la rediriger vers le cube qui offre les meilleures performances.

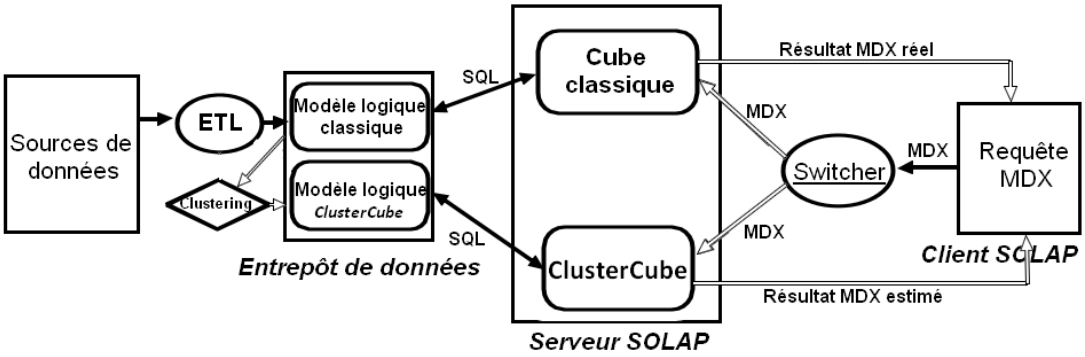


Fig 5. 4 Architecture ROLAP étendue

Le module « Switcher » effectue une lecture de la requête MDX soumise par l'utilisateur via le client SOLAP, extrait les membres requis par la requête (ex. [Grille].[AllGrilles], [Temps].[2014], [Measures].[Value]), puis les utilise pour extraire leur coût d'exécution depuis un modèle de coûts.

En fonction d'une contrainte de temps d'exécution maximale définie à l'avance par l'utilisateur, le « Switcher » redirige la requête vers le « cube classique » sous sa forme originale, ou vers le « ClusterCube » sous une forme réécrite. Le principe de fonctionnement du « Switcher » est illustré figure 5.5.

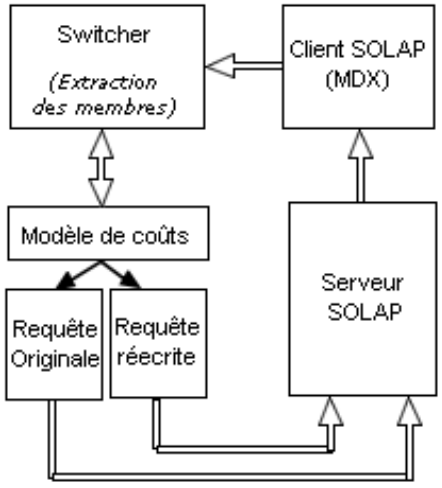


Fig 5. 5 Extraction, réécriture et redirection des requêtes via le Switcher

5.3 Méthodologie

Dans cette section nous présentons de façon plus détaillée l'étape de réduction des faits du cube original par l'échantillonnage et notamment par le clustering, ainsi que la modélisation du « ClusterCube ». Nous présentons un méta-modèle qui définit les concepts principaux du ClusterCube (dimension, niveau, mesure, …), ainsi que son instance pour notre cas d'étude. Puis nous proposons un modèle de mapping qui permet de transformer les requêtes effectuées sur le cube classique en membres de dimension du ClusterCube.

5.3.1 Réduction des faits par l'échantillonnage

Dans cette première étape, nous regroupons les valeurs des points des grilles régulières en utilisant la méthode du « Cluster Sampling » basée sur les k-moyennes (ou K-means en anglais) (Han 1997). Dans nos expérimentations, nous avons choisi d'utiliser cette méthode d'échantillonnage pour sa simplicité d'implémentation ainsi que sa robustesse et sa rapidité. Néanmoins notre méthode d'optimisation des requêtes d'agrégation peut parfaitement être considérée avec d'autres techniques d'échantillonnage (cf. section 3.1).

Ce processus du clustering est un prétraitement. Il est effectué une seule fois en mode offline. Il consiste à classifier les valeurs de la table de faits dans des clusters en fonction de leurs similarités (c'est-à-dire « les ressemblances »). En d'autres termes, les valeurs des points de la grille qui se ressemblent sont classifiées dans le même cluster. Dans le cas des phénomènes continus qui ne varient pas fréquemment dans l'espace (ex. la température), si nous exploitons le fait qu'à plusieurs localisations du phénomène étudié les valeurs se ressemblent, nous pouvons grouper ces valeurs similaires dans un même cluster et utiliser une seule valeur pondérée qui représentera ses jumelles dans le même cluster. Ainsi, nous pouvons approximer le résultat de l'agrégation et nous aurons réduit le nombre de faits nécessaires à l'agrégation.

La similarité peut être définie de plusieurs manières, selon le type de données étudiées et la marge de précision acceptée. Elle peut être basée sur les différences entre les entités ou les points en commun qu'elles partagent, ou sinon, elle peut signifier que les entités sont identiques, ce qui représente la similarité maximale (Lin 1998).

Lors du processus de clustering, le choix du critère de similarité peut être d'une grande influence sur les performances obtenues. Les données doivent être regroupées dans des clusters, de manière à ce que l'intervalle de similarité choisie puisse produire un nombre de clusters minimal, pour maximiser les performances de calcul tout en gardant une inter-

similarité adéquate pour minimiser la perte de précision. Un critère de similarité très récurrent dans les données peut générer un plus petit nombre de clusters qui contiennent chacun un plus grand nombre de données.

La similarité stricte (c-à-d un ensemble de valeurs sont considérées similaire si et seulement si elles sont identiques) est utilisée si nous considérons que la précision est plus importante que les performances.

La figure 5.6 montre le prétraitement effectué sur les 3 grilles régulières de la figure 5.2, en vue de regrouper leurs valeurs dans des clusters inter-similaires.

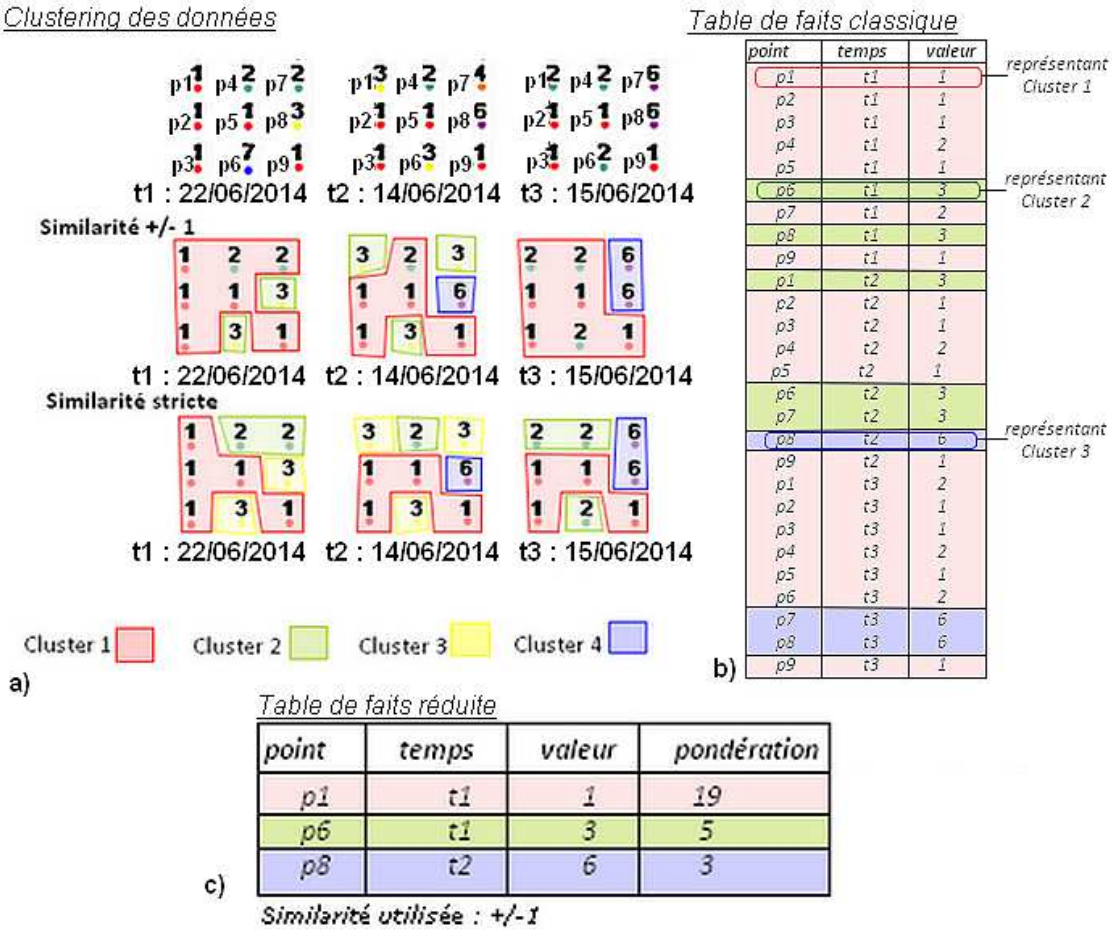


Fig 5. 6 Clustering de 3 grilles régulières avec une similarité stricte et avec une similarité de +/-1

Les valeurs de la grille régulière (figure 5.6-a) correspondante respectivement aux 3 jours : t1 (22/06/2014), t2 (14/06/2014) et t3 (15/06/2014), sont soumises au processus de clustering qui regroupe dans chaque cluster les valeurs similaires. Cette similarité peut être stricte ou approximative. Avec une similarité stricte les données d'un même cluster sont strictement égales, par contre avec une similarité approximative de +/-S, un même cluster peut accueillir des valeurs dont la différence ne dépasse pas la valeur absolue de S. Ainsi, dans notre

exemple, pour un critère de similarité stricte nous obtenons 4 clusters, contre 3 clusters pour un intervalle similarité de +/- 1. En effet, plus l'intervalle de similarité +/-S est grand et plus nombreux sont les faits regroupées dans chaque cluster et ainsi plus petit est le nombre de clusters.

Selon l'exemple illustré (figure 5.6-b) les faits sont assignés à leurs clusters respectifs en utilisant une similarité approximative de +/-1 (Cluster 1, Cluster 2 et Cluster 4). Puis, un fait (que nous appellerons centroid) représentant chaque cluster est choisi pour représenter les faits du cluster auquel il appartient (<p1/t1/1> représente le Cluster 1, <p6/t1/3> représente le Cluster 2 et <p8/t2/6> représente le Cluster 3).

L'objectif est de minimiser les jointures entre la table de faits et les dimensions. Ainsi, une nouvelle table de faits est créée depuis la table de faits originale (figure 5.6-c), qui ne comporte que les faits « centroids » et leurs coefficients de pondération qui permettent de pondérer la valeur de chaque centroid selon le nombre de faits qu'il représente pour une requête prédéfinie. Par exemple, la requête illustrée dans la figure 5.6, qui consiste en l'agrégation de toutes les valeurs qui correspondent à la grille régulière (sur tous les jours), utilise tous les faits disponibles. Ainsi, cette requête utilise les valeurs de tous les centroids, chacun pondéré selon le nombre total de faits qu'il représente (<p1/t1/1> - **19**, <p6/t1/3> - **5** et <p8/t2/6> - **3**). Une requête différente utilisera un sous-ensemble de centroids avec des coefficients de pondération différents.

5.3.2 Le Cluster Cube

Une fois le processus de clustering décrit, les représentants choisis et les requêtes jugées couteuses définies au préalable, nous arrivons à la modélisation du « ClusterCube » qui permet d'utiliser les clusters dans le calcul des agrégations couteuses. Dans cette section, nous montrons aussi comment s'effectue le mapping entre le cube de données originales et le ClusterCube en vue d'extraire les requêtes à estimer depuis le cube original et les représentants à utiliser dans le processus d'agrégation optimisée.

5.3.2.1 Méta-modèle du ClusterCube

Le cube « ClusterCube » permet de retrouver pour chaque requête MDX qu'il reçoit du module « Switcher », les centroids des clusters qui lui correspondent, puis de les utiliser pour approximer le résultat de l'agrégation.

Bien que le principe de l'agrégation en utilisant le ClusterCube présente une certaine ressemblance avec celui des vues matérialisées (Chaudhuri et al. 1995) au niveau des pré-requis (définition des requêtes coûteuses, redirection des requêtes) et des objectifs (optimisation des performances de calcul), la différence entre les deux méthodes se matérialise dans le compromis précision/performances. En effet, les vues matérialisées génèrent des résultats précis, mais elles sont coûteuses en ce qui concerne le stockage, puisqu'elles disposent d'un grand nombre de valeurs pré-agrégées et stockées au préalable dans la table de faits. Ceci n'est pas le cas du ClusterCube dont la taille de la table de faits (Fact Odeur) est égale au nombre de clusters définis lors de l'étape du clustering

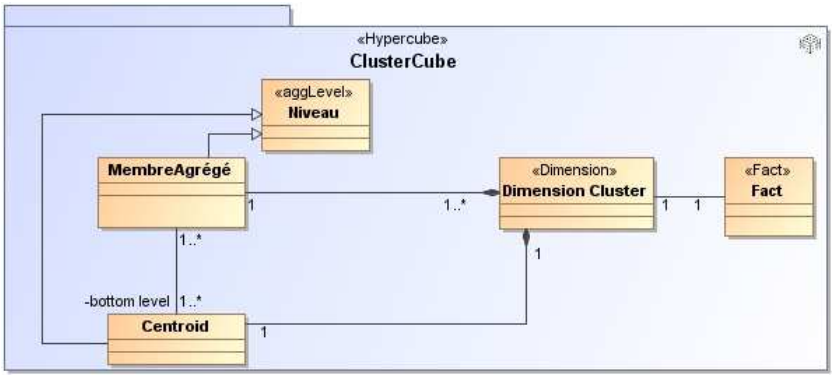
Ainsi, afin de pouvoir effectuer des opérations d'agrégation qui utilisent des faits réduits, grâce au clustering dans le ClusterCube, nous proposons le méta-modèle illustré dans la figure 5.7-a. Ce méta-modèle définit le ClusterCube comme un cube qui dispose de faits liés à une seule dimension (Dimension Cluster). La Dimension Cluster est composée de plusieurs niveaux hiérarchiques, dont le plus détaillé est le niveau Centroid. Celui-ci est composé d'un échantillon de faits qui est utilisé pour calculer l'agrégation au niveau Membre Agrégé. La Dimension Cluster dispose d'un ou plusieurs niveaux Membre Agrégé, ce qui permet de hiérarchiser les requêtes à estimer, par exemple, par échelles ou résolutions dans le cas d'une analyse spatiale. Le niveau Membre Agrégé est composé d'un ensemble de membres qui représentent des requêtes d'agrégation à différentes granularités. Sa valeur est calculée grâce à l'agrégation « pondérée » des échantillons de faits issues du cube original.

La figure 5.7-b montre le modèle multidimensionnel du ClusterCube décrit précédemment pour l'analyse de l'odeur. Ce modèle est composé d'une table de faits « Odeur », reliée aux centroids de la dimension Cluster. La mesure dérivée « ValeurApproximée » est chargée du calcul de l'agrégation demandée en utilisant les faits des centroids.

Le niveau spatial « Temps-Grille » est une instance de la classe MembreAgrégé. Il est composé des combinaisons de membres des requêtes d'agrégation liées aux dimensions « Temps » et « Grille » du cube original (figure 5.1).

Une instance de la hiérarchie de la dimension Cluster est illustrée dans la figure 5.8-a. Celle-ci est composée de deux niveaux d'agrégation, le niveau MembreAgrégé « Temps-Grille » et le niveau « Centroid ».

a)



b)

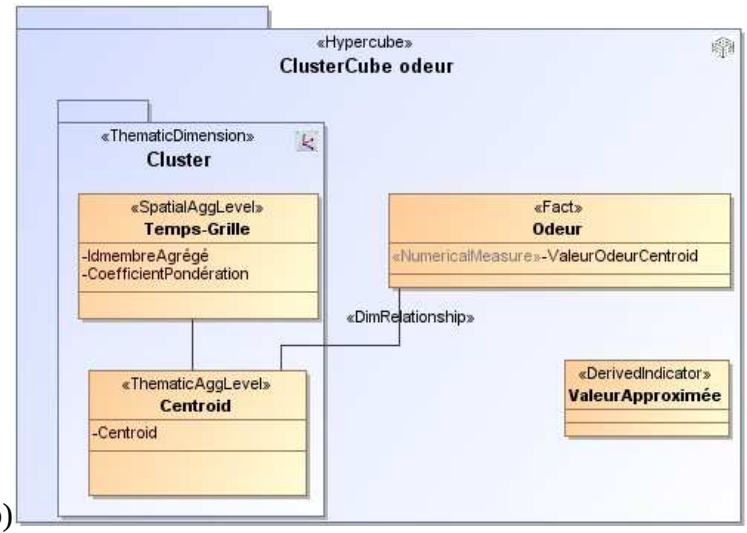


Fig 5. 7 a) Méta-modèle du ClusterCube, b) Modèle multidimensionnel du ClusterCube

En utilisant le modèle multidimensionnel du cube original (figure 5.1), une requête d'agrégation MDX portant sur l'agrégation de tous les membres de l'entrepôt de données afin de récupérer une seule valeur agrégée, est exprimée en langage MDX de la manière suivante :

Requête1:

```
Select [Grille].[AllGrilles] on rows,
[Temps].[AllTemps] on columns
From CubeOriginal
Where [Measures].[odorMeasure]
```

Le membre agrégé qui correspond à cette requête est exprimé dans la dimension Cluster de la manière suivante : `[AllTemps-AllGrilles]`.

Ainsi, Le membre agrégé « [AllTemps-AllGrilles] » est lié aux centroids : p1/t1/1, p6/t1/3, p8/t2/6. Ceci veut dire que les valeurs qui correspondent à ce niveau d'agrégation sont réparties sur tous les clusters représentés par ces centroids.

La figure 5.8-b indique les faits originaux reliés à chaque centroid, ainsi que le nombre de faits parmi eux qui correspondent à la requête à estimer. Ce nombre est utilisé pour pondérer la valeur du centroid. Dans le cas de la requête [AllTemps-AllGrilles] tous les faits de chaque cluster sont requis.

Ainsi, pour chaque couple « MembreAgrégé» et « Centroid » nous ajoutons la propriété de pondération qui est utilisée pour pondérer la valeur du centroid.

Par exemple, pour une agrégation de toutes les valeurs (requête [AllTemps-AllGrilles]), le centroid « p1/t1/1 » est pondéré par 19, le centroid « p6/t1/3 » est pondéré par 5 et le centroid « p8/t2/6» est pondéré par 3.

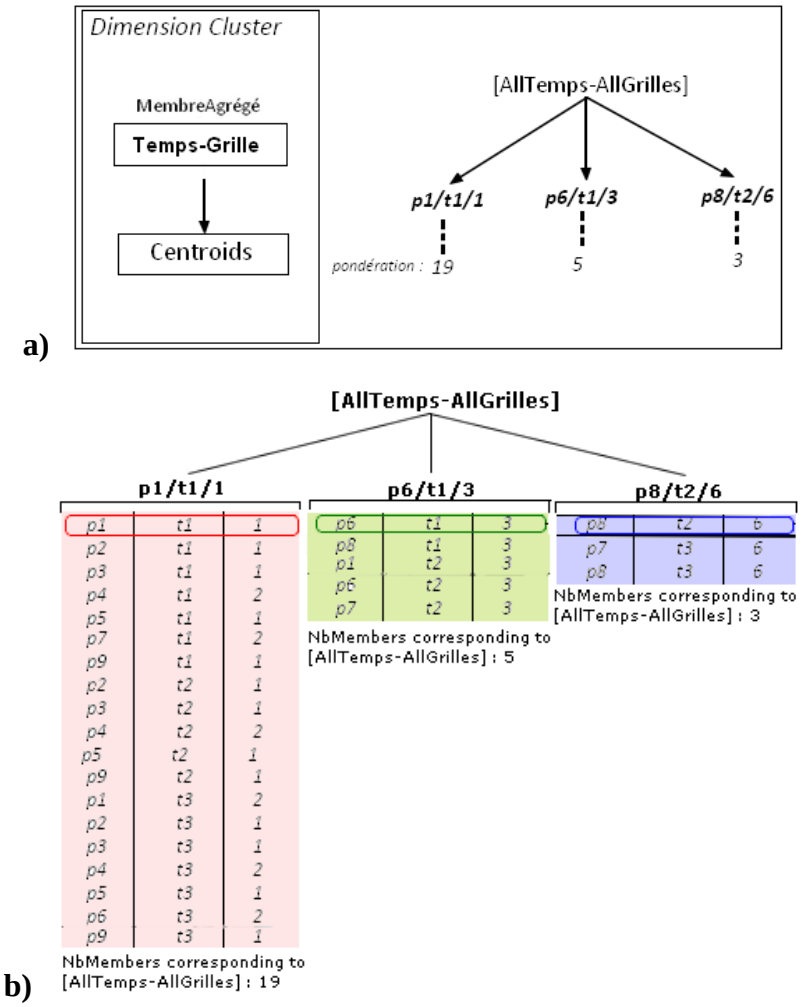


Fig 5. 8 Hiérarchie de la dimension Cluster

Un deuxième exemple consiste en l'agrégation de tous les faits au point p1 dans le temps (requête [AllTemps-p1]). Cette requête requiert 3 valeurs de p1 : une valeur au jour t1, une valeur au jour t2 et une valeur au jour t3. Les valeurs de p1 aux jours t1 et t3 appartiennent au cluster représenté par « p1/t1/1 » et la valeur de p1 au jour t2 appartient au cluster représenté

par « p6/t1/3 ». Ainsi, le centroid « p1/t1/1 » est pondéré par 2 et le centroid « p6/t1/3 » est pondéré par 1.

Le mapping entre le ClusterCube et le cube original est illustré figure 5.9. Ce mapping permet de transformer les membres des requêtes émises vers le cube original en membres du ClusterCube et les faits du cube original en échantillons (centroids) du ClusterCube.

La figure 5.9 montre la relation entre le cube original (faits, dimensions et membres) et le ClusterCube (Centroid et MembreAgrégé).

Un membre agrégé est une combinaison des membres de plusieurs dimensions du cube original, à un niveau d'agrégation élevé, qui répondent à certaines requêtes qui sont considérées coûteuses. Chaque centroid est associé à un fait du cube original et à plusieurs membres agrégés. Les centroids représentent un échantillon des faits qui est utilisé pour calculer une estimation des opérations d'agrégation sur les membres agrégés.

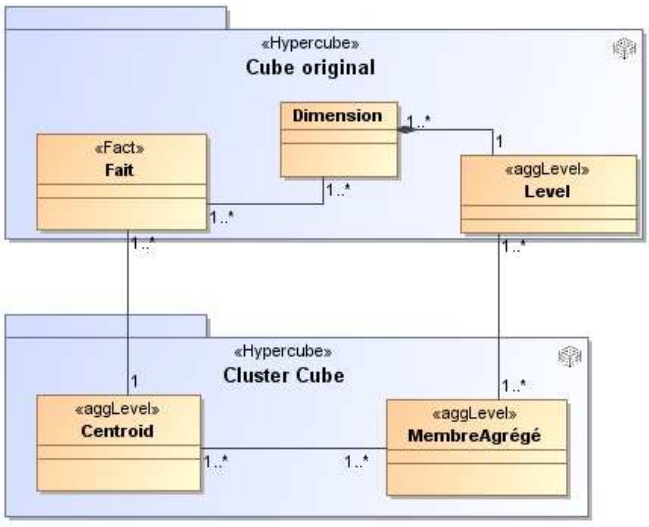


Fig 5. 9 Modèle multidimensionnel du ClusterCube

Ainsi, la requête 1 peut être exprimée dans le ClusterCube, en utilisant le membre agrégé [AllGrilles-AllTemps] qui est l'objet de la requête et la mesure dérivée « ValeurApproximée » qui utilise les centroids et leurs valeurs dans le calcul, de la manière suivante :

Requête 2 :

```
Select [Cluster].[AllGrilles-AllTemps] on rows,
[Measures].[ValeurApproximée] on columns
From ClusterCubeOdeur
```

5.4 Mise en œuvre de notre méthodologie dans une architecture ROLAP étendue

5.4.1 Mise en œuvre dans ROLAP

Pour intégrer la méthodologie proposée précédemment dans une architecture ROLAP, nous nous sommes basés sur des standards afin de fournir une approche générique. En effet l'utilisation de standards tels que SQL et MDX permettent de transformer une requête émise par l'utilisateur et de la rediriger vers le cube SOLAP qui procure les meilleures performances d'une manière transparente pour l'utilisateur quelle que soit l'architecture ROLAP utilisée.

Dans cette section, nous décrivons la mise en œuvre de la méthodologie proposée en commençant par la modélisation logique du cube original puis du ClusterCube au niveau du SGBD, puis leur modélisation physique au niveau du serveur SOLAP.

La modélisation logique d'un cube dans le SGBDS est effectuée, le plus souvent, selon un schéma en étoile. La figure 5.10-a montre un modèle en étoile générique d'un entrepôt de données composé de N dimensions. En se basant sur ce modèle, nous présentons le modèle logique (figure 5.10-b) du cube original décrit figure 5.1. Ce modèle permet d'interroger les données originales et d'effectuer des agrégations exactes.

Lorsqu'une requête d'agrégation est trop couteuse pour qu'elle soit calculée en utilisant ce modèle, celle-ci est redirigée vers le modèle logique de ClusterCube décrit figure 5.10-c.

Dans le modèle logique du ClusterCube, nous définissons les combinaisons des membres de dimensions qui constituent les requêtes à estimer (ex. p1 et AllTimes). Ces combinaisons sont transformées en membres du niveau « MembreAgrégé » de la dimension Cluster.

Le modèle logique du ClusterCube est un modèle logique en étoile classique qui dispose d'une table de faits reliée à la dimension Cluster afin de fournir les valeurs des centroids qui correspondent à chaque membre du niveau « Membre Agrégé ». Une instance de ce modèle est présentée dans la figure 5.10-d. Celle-ci montre les membres agrégés [AllGrilles-AllTemps] et [p1-AllTemps] qui correspondent à deux requêtes d'agrégation jugées coûteuses, à estimer. La première requête consiste à agréger tous les faits et la deuxième requête consiste à agréger toutes les valeurs au point p1 (figure 5.6).

Le membre agrégé [AllGrilles-AllTemps] est associé aux centroids : *p1/t1/1* pondéré par 19, *p6/t1/3* pondéré par 5 et *p8/t2/6* pondéré par 3.

Le membre agrégé [p1-AllTemps] est associé aux centroids : *p1/t1/1* pondéré par 2 et *p6/t1/3* pondéré par 1.

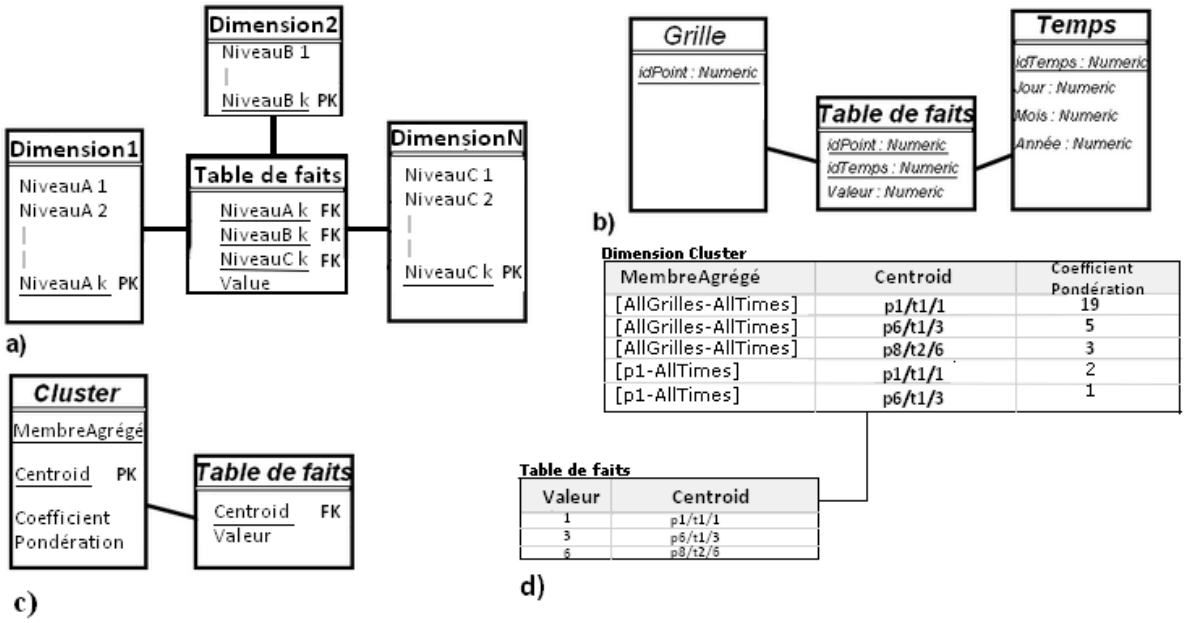


Fig 5. 10 a) Modèle logique en étoile classique b) Modèle logique classique du cas d'étude c) Modèle logique du ClusterCube d) Instance du modèle logique du ClusterCube

Afin d'exploiter le modèle logique proposé (figure 5.10-c) dans l'architecture SOLAP relationnelle, un modèle de mapping est utilisé pour lier les tables du SGBDS aux concepts spatio-multidimensionnels du serveur SOLAP (dimension, hiérarchie, niveaux, mesures, …) :
Ce modèle utilise des balises pour décrire le ClusterCube (ex. *Cube name*), sa table de fait (ex. *Table name*), la dimension Cluster (ex. *Dimension name*), sa hiérarchie et ses niveaux (ex. *Level name*), ainsi que la propriété du coefficient de pondération (ex. *Property name*) pour chaque combinaison d'un membre agrégé et d'un centroid.
La mesure dérivée « ValeurApproximée », décrite dans le modèle du ClusterCube (figure 5.9) est ajoutée au modèle physique du ClusterCube. Celle-ci utilise dans sa formule de calcul une deuxième mesure « ValeurPondérée ».
La mesure « ValeurPondérée », effectue la pondération des valeurs des centroids en utilisant la valeur de chaque centroid et le coefficient de pondération qui lui correspond.
Puis, la mesure «ValeurApproximée », effectue l'opération requise pour l'agrégation (somme, moyenne, …) en utilisant les valeurs pondérées obtenues. Les deux mesures sont représentées en utilisant le langage MDX de la manière suivante :

ValeurPondérée:

```
WITH
   MEMBER [Measures].[ValeurPondérée] AS
[Measures].[Value]*[Cluster].currentmember.Properties('Coef')
```

ValeurApproximée:

```
WITH
   MEMBER [Measures].[ValeurApproximée] AS
SUM([Cluster].currentmember.children,[Measures].[ValeurPondérée])
```

L'utilisation de la mesure calculée « ValeurApproximée » implique un appel à la mesure « ValeurPondrée ». Donc il suffit de faire appel à la mesure « ValeurApproximée » dans une requête MDX classique en utilisant le ClusterCube pour répondre aux requêtes d'agrégation par l'approximation de la manière suivante :

```
Select [MembreAgrégé].[AllGrilles-AllTemps] on rows,
[Measures].[ValeurApproximée] on columns
From [ClusterCube]
```

5.4.2 Expérimentations et implémentations

Dans cette section nous étudions les performances du ClusterCube. L'ordinateur utilisé pour le requêtage du cube original et du ClusterCube a la même configuration utilisée dans le chapitre 3 : processeur Intel Core i3 2,20 GHz, 4 Go RAM, Système d'exploitation Windows 7 professionnel, système OS 64 bits.

Les données spatiales et alphanumériques sont stockées dans le SGBD PostgreSQL, en utilisant PostGIS pour la gestion des composantes géométriques. Le serveur SOLAP utilisé est GeoMondrian. Cette architecture est la même utilisée dans les implémentations présentées dans le chapitre 3.

Pour nos expérimentations, nous avons effectué une requête d'agrégation, par la somme, du membre [AllGrilles-AllTimes] qui correspond à l'agrégation de toutes les valeurs de l'entrepôt de données. Nous avons effectué nos tests en nous basant sur cette requête parce qu'elle représente la requête la plus coûteuse à effectuer lors d'une agrégation dans le cube de données originales. Nous avons utilisé un entrepôt de données de 10 200 000 valeurs (10000

points de la grille à résolution 100*100 sur 1020 jours) et un entrepôt de données de 15 000 000 de valeurs (10000 points de la grille à résolution 100*100 sur 1500 jours).

Pendant la phase de réduction des faits, nous avons généré un ClusterCube avec 1000 clusters puis un deuxième ClusterCube avec 400 000 clusters en utilisant une similarité stricte (cf. section 5.3.1). Donc la requête d'agrégation optimisée s'effectuera sur 1000 valeurs, puis 400 000 valeurs, dans la table de faits.

Le critère de similarité stricte que nous utilisons implique que toutes les données représentées par un même centroid ont la même valeur et que les résultats des requêtes obtenus sont exacts et n'ont pas besoin de définir un intervalle d'erreur. Ainsi, nos expérimentations sont présentées comme preuve de faisabilité de la méthodologie proposée. Elles montrent le gain de performance obtenu, selon le nombre de clusters utilisés et le nombre de données entreposées dans l'EDS. L'analyse de la précision des résultats dépend de la méthode d'échantillonnage utilisée et peut varier d'une application à une autre. Des travaux tels que (Jermaine 2003) (Jin et al. 2006) et (Hellerstein, Haas et Wang 1997) se focalisent sur la précision des résultats des estimations basées sur l'échantillonnage dans l'OLAP.

La figure 5.11 montre le gain en temps d'exécution offert par le ClusterCube. Ce gain augmente quand le nombre de clusters est réduit, mais ceci au dépend de la précision du résultat. Par exemple, l'agrégation de 10 200 000 de faits en utilisant 400 000 clusters (25,5 faits par cluster en moyenne), présente un gain en temps de calcul de 74,34% avec une précision exacte (similarité stricte).

Un autre avantage du ClusterCube est que sa taille et ses performances varient peu malgré la mise à jour de l'entrepôt de données. En effet, lors de l'insertion de nouvelles données, le processus de mise à jour des clusters attribut chaque nouvelle donnée ajoutée à un cluster, puis recalcule le coefficient de pondération pour chaque couple Centroid/MembreAgrégé. Ainsi, si les nouvelles données sont homogènes avec les clusters existants, les performances du ClusterCube ne changent pas. Sinon, si les nouvelles données ne peuvent pas être représentées pas les clusters existants, de nouveau clusters sont crées, puis leurs centroids sont ajoutés à la table de faits du ClusterCube.

[AllGrilles-AllTimes]	Sans clusters	400 000 clusters	1000 clusters
10200000 valeurs	83868	21513	363
15000000 valeurs	127395	21513	363

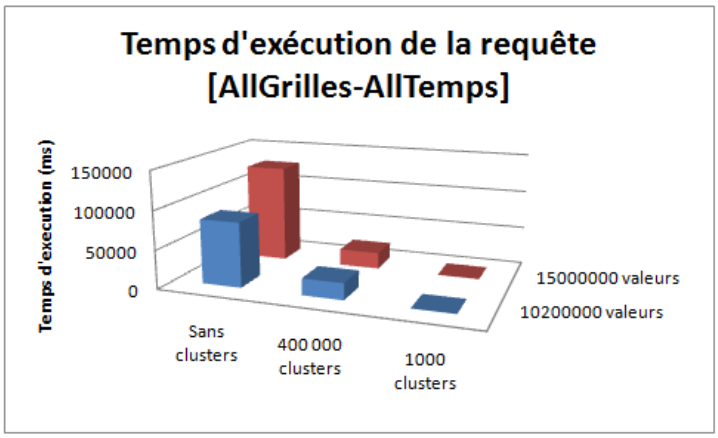


Fig 5. 11 Temps d'exécution de la requête [AllGrilles-AllTemps] avec et sans ClusterCube

5.5 Le ClusterCube pour l'analyse des champs continus « ClusterCube&Field »

En ce qui concerne la compatibilité du ClusterCube avec les opérateurs SOLAP, tel que le Slice spatial, le Roll-Up spatial, ou le Drill-Down spatial et les opérateurs FieldMDX présentés dans le chapitre 4, qui permettent la gestion de la continuité et de la multi-résolution des champs continus incomplets, le ClusterCube peut répondre à ce type de requêtes si tous les besoins d'analyse sont identifiés au préalable (niveaux de résolution, méthode d'interpolation à utiliser, …).

En effet, le rôle du ClusterCube est d'améliorer les performances des requêtes d'agrégation en utilisant des échantillons de données. Cependant, si l'échantillonnage est effectué exclusivement sur les dimensions non spatiales (ex. dimension temporelle) et que le ClusterCube dispose de tous les membres spatiaux dans ses dimensions spatiales, alors les opérateurs spatiaux tel que le slice spatial peuvent être utilisés classiquement.

Pour que le ClusterCube bénéficie d'un support qui lui permet d'analyser les champs continus incomplets à plusieurs résolutions, celui-ci doit intégrer la structure du champ continu incomplet et les fonctionnalités du FieldMDX. En d'autres termes le modèle conceptuel doit intégrer l'opérateur *InterpolatePoint* qui optimise l'analyse continue des données via diverses fonctions d'interpolation et l'opérateur *InterpolateResolution* qui implémente la méthode FISS pour l'optimisation du changement de résolution lors des analyses.

Ainsi, comme le montre la figure 5.12, contrairement au cube multidimensionnel classique, la méthode FISS offre une analyse optimisée en réduisant le volume des faits liés aux données spatiales et le ClusterCube offre une analyse optimisée en réduisant les faits liés aux données non spatiales.

Par exemple, la méthode FISS permet, grâce à la fonction *InterpolateResolution,* de retrouver les valeurs du phénomène analysé à une résolution élevée sans que les faits associés à celle-ci ne soit disponible dans l'entrepôt de données (cf. section 4.4.2.2). Ce qui permet d'optimiser l'analyse en réduisant le volume des faits liés aux composantes spatiales.

Le ClusterCube, permet par exemple, d'estimer une valeur moyenne du phénomène étudié à une localisation et période données, en utilisant un sous ensemble des faits.

Les deux approches, étant basées sur l'estimation, elles présentent un compromis entre la précision et les performances. Ce compromis doit être paramétré selon le type d'analyses à effectuer et le type de données à analyser.

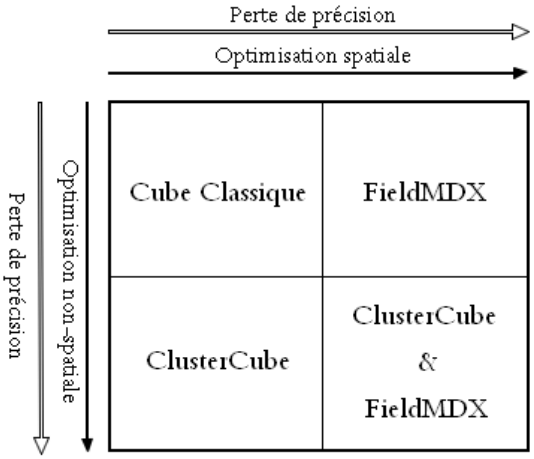


Fig 5. 12 Optimisation spatiale et non-spatiale des analyses multidimensionnelles

Le ClusterCube&Field représente une combinaison entre les avantages du ClusterCube et ceux de l'analyse spatiale continue, en permettant d'utiliser l'approche de modélisation FISS et les opérateurs FieldMDX en addition à la réduction des données non-spatiales fournie par le ClusterCube.

Ainsi, dans cette section, nous présentons une extension du modèle conceptuel UML du ClusterCube&Field, afin d'obtenir une optimisation spatio-temporelle et thématique qui permet d'utiliser pleinement les capacités spatiales du serveur SOLAP dans l'analyse des champs continus incomplets tant que l'agrégation est effectuée selon la technique d'approximation proposée par le ClusterCube.

5.5.1 Modélisation du ClusterCube&Field

Le modèle spatio-multidimensionnel que nous proposons dans la figure 5.13, est une instance du méta-modèle du ClusterCube présenté dans la figure 5.7-a. Il représente le cube spatio-multidimensionnel "ClusterCube&Field". Celui-ci est composé des mêmes concepts que le ClusterCube (c.-à-d. *dimension Cluster (niveau MembreAgrégé, niveau Centroid) et Fact*) à la seule différence près que la dimension Cluster dispose d'une hiérarchie spatiale de type "FieldIncomplete" qui est composée de deux niveaux qui étendent le "MembreAgrégé" avec une méthode d'interpolation, une résolution et une donnée géométrique (GrilleAgrégée100 et GrilleAgrégée200), en addition au niveau Centroid.

Le niveau "GrilleAgrégée100" de type "FieldIncomplete" est une "RegularGridPoint" de résolution 100*100 (10000 points) et le niveau "GrilleAgrégée200", du même type, représente une "RegularGridPoint" dont la résolution est de 200*200 (40000 points). Chaque niveau dispose d'une fonction d'interpolation bilinéaire pour générer sa continuité, un attribut qui représente sa résolution et une géométrie qui sera utilisée pour l'analyse spatiale.

Le niveau "Centroid" dispose des centroids obtenus suite au clustering des membres non-spatiaux. Ainsi les valeurs des membres agrégés du niveau "GrilleAgrégée100" (ex. « point1/ALL » ou « point1/2014 ») sont calculées en utilisant les valeurs des centroids dans les faits "Fact". Les faits sont reliés au niveau "Centroid" et ne disposent que des valeurs des centroids des clusters générés pour le niveau de résolution (100*100). Ce modèle que nous proposons, tel que le montre le "BasicIndicator" (ClusterCube&FieldIndicator), permet d'utiliser les niveaux "GrilleAgrégée100" et "Centroid" pour calculer une agrégation basée sur l'échantillonnage, puis d'utiliser les niveaux "GrilleAgrégée100" et "GrilleAgrégée200" pour changer le niveau de résolution en utilisant l'approche FISS présentée dans le chapitre 3.

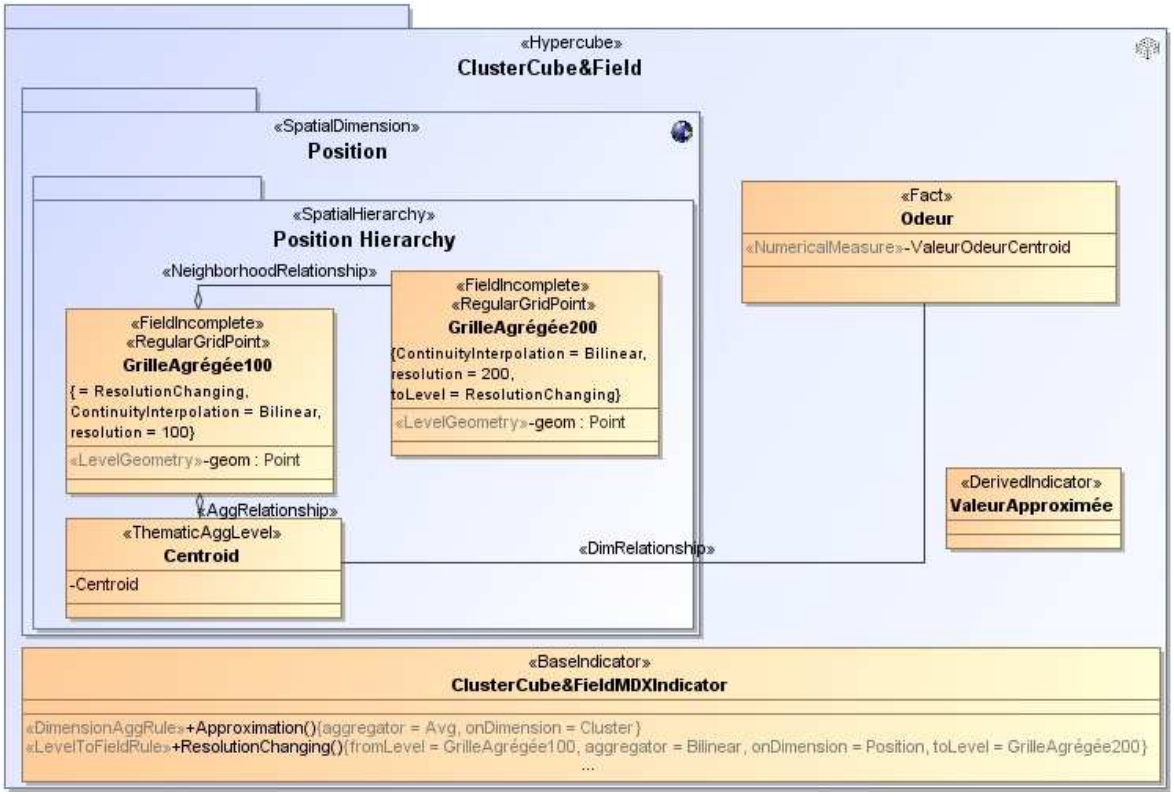


Fig 5. 13 Modèle conceptuel du ClusterCube&Field

Quelque soit le nombre de niveaux de résolutions requis lors de la modélisation du ClusterCube&Field, le principe de la dimension "Cluster" reste le même. Une relation d'agrégation (AggRelationship) relie le niveau "Centroid" au niveau de résolution le plus grossier et permet d'obtenir les valeurs agrégées des points de la "RegularGridPoint" de base (la moins fine) , puis une relation de voisinage relie les niveaux de type "FieldIncomplete" à résolutions élevées au niveau de résolution le plus grossier (NeighborhoodRelationship) et permet d'obtenir les valeurs agrégées à haute résolution grâce à l'interpolation.

La continuité de chaque niveau agrégé de type "FieldIncomplete" représenté par une "RegularGridPoint" est générée en utilisant la fonction d'interpolation "ContinuityInterpolation".

La prochaine sous section, explique avec plus de détails comment le ClusterCube&Field peut être utilisé pour optimiser l'analyse spatiale continue à multi-résolution dans le SOLAP.

5.5.2 L’analyse spatiale continue à multi-résolution avec le ClusterCube&Field

L’analyse spatiale des grilles régulières de points dans le ClusterCube&Field consiste, en premier lieu, à pouvoir utiliser les opérateurs spatiaux, tel que le slice spatial, classiquement, sans interférence avec l’approche d’échantillonnage.

Le ClusterCube&Field, tel que défini dans le modèle spatio-multidimensionnel de la figure 5.13, implique que chaque niveau spatial continu est composé de l’ensemble des points qui constituent la grille.

Par exemple, supposons que la grille de base analysée est à la résolution 6*6. Les faits sur cette grille sont analysés aux jours 12/02/2014, 19/02/2014 et 28/02/2014.

La figure 5.14-a montre comment on peut effectuer une agrégation approximée grâce aux capacités du ClusterCube, puis utiliser les opérateurs spatiaux et les opérateurs FieldMDX sur les résultats obtenus.

Le slice spatial (figure 5.14-b) dans le ClusterCube&Field (*Agrégation+slice spatial*) consiste en un slice spatial classique sur les points de la grille à un niveau d’agrégation élevé. Ainsi les faits de la grille de points aux jours 12/02/2014, 19/02/2014 et 28/02/2014 sont agrégés pour obtenir les valeurs de la grille pour l’année 2014, puis le slice spatial est effectué sur la base de la distance (prédicat de distance) pour récupérer un sous-ensemble des points de la grille (résolution 4*4).

En ce qui concerne l’analyse continue des données spatiales, qui est la caractéristique la plus importante des champs continus et de leur analyse, le ClusterCube&Field dispose de l’ensemble des membres spatiaux susceptibles d’être utilisés pour déterminer le voisinage d’une localisation choisie par le décideur, puis d’estimer sa valeur (figure 5.14-c).

Ainsi, chaque niveau « MembreAgrégé » de la dimension "Cluster" (figure 5.13) est de type "RegularGridPoint" et dispose d’une fonction d’interpolation pour assurer sa continuité.

Dans le cube original, une requête impliquant une localisation non-échantillonnée à un niveau agrégé (ex. valeur à la localisation *p(x, y)* pour l’année 2012), est calculée par une agrégation classique Map Algebra des voisins de cette localisation, avant qu’une interpolation ne soit effectuée sur les valeurs obtenues (cf. section 4.3).

Par contre, dans le ClusterCube&Field, les requêtes impliquant une localisation non-échantillonnée à un niveau agrégé, sont calculées par l’agrégation des voisins de cette localisation selon les centroids adéquats (*Agrégation Approximée*). Puis, la méthode

d'interpolation *InterpolatePoint* de FieldMDX utilise les valeurs obtenues après l'agrégation du voisinnage pour estimer une valeur agrégée à la localisation voulue (*Agrégation + continuité*).

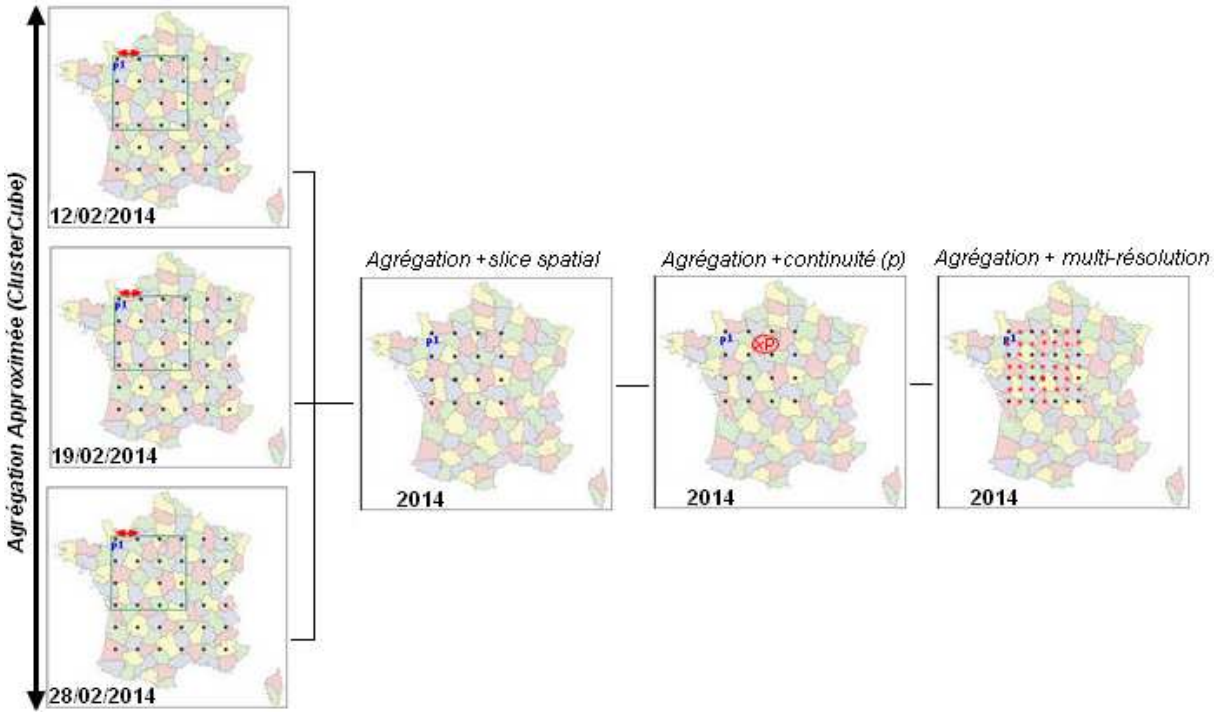


Fig 5. 14 a) Agrégation approximée, b) Agrégation approximée et slice spatial, c) Agrégation approximée et continuité et d) Agrégation approximée et multi-résolution

La multi-résolution est gérée dans le ClusterCube&Field de la même manière que la continuité. C'est-à-dire, en remplaçant la méthode d'agrégation classique utilisée (MapAlgebra), par le processus d'agrégation approximée du ClusterCube (figure 5.14-d).

Le ClusterCube&Field offre une optimisation spatiale et non-spatiale des analyses en calculant à la volée les faits liés aux points des niveaux de résolutions élevés grâce à l'interpolation spatiale (FISS) et en calculant les agrégations sur les dimensions non-spatiales en utilisant l'échantillonnage (optimisation non-spatiale).

Ainsi, comme le montre le modèle multidimensionnel présenté dans la figure 5.13, les deux niveaux de résolution "GrilleAgrégée100" et "GrilleAgrégée200" partagent la relation de voisinnage "NeighborhoodRelationship" qui permet d'affecter à chaque membre du niveau "GrilleAgrégée200" ses membres voisins au niveau "GrilleAgrégée100" (cf. section 5.5.1).

Lorsque la requête effectuée par le décideur requiert des valeurs agrégées de la grille régulière, à un niveau de résolution élevé (*Agrégation + multi-résolution*), le

ClusterCube&Field utilise la "NeighborhoodRelationship" pour récupérer les voisins de chaque membre de cette résolution depuis le niveau de résolution le moins fin (ex. "GrilleAgrégée100"). Puis, il effectue leur agrégation en utilisant les centroids et leurs valeurs. Le résultat obtenu pour chaque voisin est utilisé dans le processus d'interpolation pour estimer les valeurs à la nouvelle résolution ("GrilleAgrégée200").

Cette méthode permet d'effectuer une réduction du nombre de faits correspondants aux membres des résolutions élevées par l'interpolation, ainsi qu'une réduction des faits correspondants aux membres des dimensions non-spatiales par l'échantillonnage.

La figure 5.14-d représente une requête qui consiste à générer des valeurs agrégée pour l'année 2014 à haute résolution (7*7) à partir des valeurs à faible résolution (4*4) obtenues après le slice spatial.

Pour résumer, dans le ClusterCube&Field, les opérateurs Map Algebra utilisés classiquement dans le processus d'agrégation sont remplacés par la méthode d'échantillonnage qui optimise l'agrégation en l'approximant. Puis, les méthodes d'interpolation sont utilisées sur la grille de points agrégée obtenue, pour générer les valeurs aux résolutions élevées.

5.6 Conclusion

Nous avons proposé dans ce chapitre une méthodologie pour optimiser les performances des requêtes d'agrégation sur un grand volume de donnée, au sein d'une architecture SOLAP relationnelle. Pour cela nous proposons d'estimer les résultats réels en utilisant les méthodes d'échantillonnage. Ainsi, nous définissons le méta-modèle du ClusterCube, qui nous a permis de proposer le modèle spatio-multidimensionnel pour l'analyse approximée de l'odeur. Nous proposons aussi une mise en œuvre de notre méthodologie dans une architecture SOLAP, suivie de quelques expérimentations.

A noter que l'utilisateur, après avoir vu les résultats estimés, peut souhaiter visualiser les résultats réels correspondants. Dans ce cas le « Switcher » pourra toujours facilement passer d'un protocole de calcul à un autre selon la demande. Un avantage de notre technique est que les résultats exacts peuvent toujours être calculés si nous le souhaitons (puisque les données de base sont conservées). Un autre avantage de notre approche est que la mise à jour des données ne présente pas de complications et ne change pas la taille du ClusterCube, seuls les coefficients de pondérations sont recalculés. Enfin, notre méthode ne nécessite pas de ré-implémenter une partie du système de gestion de BD ou d'étendre le système d'analyse

multidimensionnel SOLAP. Nous avons aussi démontré la possibilité de combiner la méthodologie proposée dans ce chapitre, avec l'approche d'intégration de l'analyse spatiale continue dans le SOLAP, proposée au chapitre 4. Le ClusterCube&Field représente une fusion des deux approches, ce qui permet de bénéficier d'une optimisation spatiale et non-spatiale des analyses sur les champs continus incomplets représentés par des grilles régulières de points.

Pour les perspectives futures, nous proposons d'effectuer une étude plus précise sur la précision des résultats obtenus en menant des expérimentations plus avancées, impliquant de nouvelles méthodes d'échantillonnage. En effet, la méthodologie que nous proposons est orientée vers l'intégration de la technique l'échantillonnage dans le calcul des requêtes d'agrégation dans l'OLAP. Cependant, l'étude de la précision des résultats dépend de la méthode d'échantillonnage utilisée, de la taille des données impliquées dans les requêtes et de l'homogénéité des échantillons choisis.

Une deuxième perspective consiste à définir une méthode d'estimation du temps d'exécution des requêtes, afin de permettre au module « Switcher » d'évaluer si la requête doit être redirigée vers le ClusterCube ou non. Cette méthode devrait être capable de générer un modèle de coût qui regroupe les temps d'exécution des requêtes les plus fréquentes.

Le module « Switcher » devrait aussi mettre à disposition de l'utilisateur une vue de paramétrage, pour qu'il puisse contrôler l'exploration du cube classique et du ClusterCube indépendamment du modèle de coûts et de la contrainte de temps d'exécution formulée lors de la conception.

Partie III : Conclusion et Perspectives

Chapitre 6. Conclusions et Perspectives

6.1 Rappel de la problématique

Les besoins d'analyse spatio-multidimensionnelle des phénomènes continus (les grilles régulières de points) à différentes résolutions spatiales dans le SOLAP sont les suivants :

(1) Permettre d'utiliser les opérateurs OLAP classiques (ex. Roll-up, slice),

(2) Permettre d'utiliser les opérateurs SOLAP (ex. opérateurs topologiques, Map Algebra),

(3) Fournir une vue continue des données pendant l'analyse spatiale,

(4) Permettre d'explorer les données spatiales continues à différents niveaux de résolution,

(5) Conserver des performances acceptables pour une analyse en ligne multidimensionnelle.

Nous nous sommes basés dans nos recherches sur un cas d'étude réel fourni par la société Agaetis (« Agaetis » 2014). Ce cas d'étude consiste en l'analyse de la pollution (odeur) sur le territoire français selon plusieurs dimensions : temps, localisation du capteur (*grille régulière de points*), source (origine) de l'odeur, type du polluant.

6.2 Contributions

Le principal axe de nos travaux est basé sur l'intégration des grilles régulières de points dans l'architecture SOLAP relationnelle. Nos propositions sont présentées dans les chapitres 4 et 5. Ces propositions s'étalent sur toutes les composantes de l'architecture du système SOLAP, de manière à proposer une approche générique, basée sur des standards tel que SQL et MDX, pour l'exploitation des grilles régulières de points dans une structure spatio-multidimensionnelle.

6.2.1 Intégration des champs continus représentés par des grilles régulières de points dans le SOLAP : de la modélisation conceptuelle à l'implémentation.

L'intégration des champs continus représentés par des grilles régulières de points dans le SOLAP a été effectuée à différentes étapes de l'architecture relationnelle.

- **Modélisation conceptuelle**

Nous proposons une extension du profil UML « CI SOLAP » (Boulil 2012) pour la définition des champs continus incomplets à multi-résolution, leurs attributs et leurs opérateurs. Nous proposons un modèle conceptuel qui définit le champ continu comme un niveau de dimension spatiale qui dispose d'une fonction d'interpolation et d'un attribut qui indique sa résolution. Une relation de voisinage « NeighborhoodRelationship » relie les membres des différents niveaux de résolution du champ continu. Cette relation entre les différentes résolutions permet d'améliorer le niveau de détails des requêtes sur les phénomènes géo-référencés, en utilisant des fonctions d'interpolation.

- **Modélisation logique**

Afin de pouvoir exploiter le modèle conceptuel que nous proposons, nous introduisons deux modèles logiques pour l'exploitation des grilles régulières à multi-résolution dans l'EDS. Tout d'abord l'approche *FASS*, qui consiste en un modèle logique basé sur le schéma en étoile dont la table de faits est reliée classiquement au niveau le plus détaillé de chaque dimension. Puis, l'approche *FISS*, qui consiste en un modèle logique en étoile dont la particularité est que la table de faits est reliée au niveau le moins détaillé de la dimension spatiale continue.
La différence entre l'approche FISS que nous proposons et l'approche FASS est que FISS ne stocke dans la table de faits que les valeurs à basse résolution spatiale (la moins fine), puis utilise la fonction d'interpolation qui correspond à chaque niveau de résolution pour estimer les valeurs de ses points. Ceci, permet d'améliorer considérablement les performances de stockage.

- **Extension du langage de requête MDX**

Nous proposons FieldMDX comme une extension de MDX. De la même manière que GeoMDX étend les opérateurs MDX avec des opérateurs spatiaux (ex. intersection, inclusion, …), nous proposons deux opérateurs pour l'analyse continue et à multi-résolution des grilles régulières : *InterpolatePoint* et *InterpolateResolution*.
InterpolatePoint (continuité) permet de retrouver la valeur de la requête demandée à chaque localisation dans l'espace d'étude. Elle récupère les valeurs des voisins requis par la fonction d'interpolation puis estime la valeur au point voulu.
InterpolateResolution (multi-résolution) permet, lors de l'utilisation de l'approche *FISS*, de calculer les valeurs des requêtes sur les grilles régulières de points à des niveaux de détails plus élevés grâce à la relation de voisinage « NeighborhoodRelationship ».

- **Implémentation et des expérimentations**

Nous avons implémenté les opérateurs FieldMDX dans une architecture ROLAP basée sur la solution OLAP-SIG hybride. Puis, nous avons proposés des expérimentations basées sur notre cas d'étude, afin de montrer une comparaison entre les performances de calcul des approches *FASS* et *FISS*. Nous avons aussi montré une comparaison entre les performances de la méthode FISS, implémentée avec la méthode d'interpolation bilinéaire, puis avec la méthode d'interpolation bicubique, afin de démontrer que le choix de la méthode d'interpolation peut influencer sur les performances obtenues.

6.2.2 Approximation des opérations d'agrégation des grilles régulières de points par l'échantillonnage dans le SOLAP.

La méthodologie que nous proposons pour l'approximation des opérations d'agrégation en utilisant les méthodes d'échantillonnage, a été définie dans ce travail à différents niveaux de l'architecture relationnelle SOLAP.

- **Définition d'un méta-modèle**

Nous proposons un méta-modèle qui permet de définir les composants du ClusterCube. Nous définissons le ClusterCube comme un cube SOLAP réduit qui permet d'optimiser les requêtes d'agrégation, en utilisant un échantillon de faits prélevé du cube original.
Le modèle que nous proposons introduit les concepts du ClusterCube :
- *La dimension « Cluster »* regroupe les membres des requêtes d'agrégation jugées couteuses dans un niveau appelé « MembreAgrégé » et les centroids qui représentent les clusters des échantillons, dans un niveau appelé « Centroid ».
- *La table de faits* est composée des faits qui correspondent aux centroids et qui sont utilisés pour calculer les valeurs des requêtes du niveau « MembreAgrégé » après avoir été pondérés.
Un modèle de mapping a aussi été proposé afin de permettre, en prétraitement, la transformation des requêtes coûteuses du cube original, en membres du niveau « MembreAgrégé » et les échantillons obtenus après le processus d'échantillonnage, en membres du niveau « Centroid ».

- **Modélisation logique**

Nous avons présenté un modèle logique basé sur le schéma en étoile qui permet l'exploitation des échantillons de données au niveau du SGBD et du SOLAP. Ce modèle représente le ClusterCube, avec la dimension « Cluster » et la table de faits, dans la structure de l'EDS. Il permet d'utiliser les mesures « ValeurPondérée » et « ValeurApproximée » qui permettent, respectivement, de pondérer les valeurs des centroids, puis d'estimer la valeur de la requête d'agrégation demandée par l'utilisateur.

- **Modélisation multidimensionnel**

Nous présentons le modèle multidimensionnel du « ClusterCube&Field », qui combine les capacités de l'analyse des champs continus incomplets et celles du ClusterCube, pour une analyse spatiale continue et optimisée, des grilles régulières de points à multi-résolution.
Ce modèle permet de calculer les résultats des requêtes d'agrégation sur les grilles régulières de points à multi-résolution en utilisant des échantillons de données. Il permet d'utiliser les opérateurs SOLAP (slice spatial, opérateurs topologiques, …), ainsi que les opérateurs FieldMDX (continuité, multi-résolution,…) dans le même cube d'analyse.

- **Implémentation et des expérimentations**

Enfin, nous proposons une implémentation de notre méthodologie dans une architecture ROLAP basée sur la solution OLAP-SIG hybride. Cette implémentation représente une preuve de faisabilité de notre approche. Elle a été réalisée en utilisant un cas d'étude concernant la pollution et montre la différence entre les performances obtenues avec notre approche et celles obtenues avec l'agrégation classique basée sur la Map Algebra.

6.3 Discussion et Perspectives

Dans cette dernière partie, nous abordons les limites des approches proposées et nous présentons des perspectives de recherche directement liées à la thèse (qui pourraient être traitées à court termes) ainsi que d'autres pistes plus prospectives.

6.3.1 Les limites et perspectives à court termes

L'utilisation du FieldMDX permet d'utiliser des méthodes d'interpolation pour estimer les valeurs manquantes et ainsi générer une continuité spatiale des données. Ce besoin est

formulé dans le profil UML SOLAP (CI SOLAP), mais n'a été instancié dans l'architecture SOLAP, que pour l'utilisation des interpolations bilinéaire et bicubique qui utilisent le principe des plus proches voisins. L'utilisation de méthodes d'interpolations supplémentaires plus complexes, telles que krigeage, permettra d'identifier de nouveaux défis liés à :

- La nature de la relation de voisinage entre les différents niveaux continus de la dimension spatiale,
- L'extension des opérateurs de continuité et de multi-résolution FieldMDX pour intégrer de nouvelles fonctions d'interpolation,
- La réévaluation des performances de calcul du FieldMDX,
- La mise en place d'une méthode qui permettra de choisir la fonction d'interpolation à utiliser en fonction des besoins.

En ce qui concerne l'utilisation du ClusterCube, celui-ci propose un cube de données réduit issue des échantillons prélevés sur le cube originale. Le ClusterCube offre une approche basée sur l'échantillonnage (clustering), pour estimer les résultats des requêtes d'agrégation volumineuses. Cependant, bien que l'approche, que nous proposons, permette de conserver des performances de calcul adaptées à l'analyse spatiale en ligne, l'étude de la précision des résultats obtenus n'a pas été réalisée. En fait, notre premier objectif était déjà d'expérimenter l'intégration des techniques d'échantillonnage dans le SOLAP, pour optimiser les performances des requêtes d'agrégation sur les grilles régulières de points. Nous avons montré que les performances obtenues varient selon le nombre de clusters générés en utilisant une similarité stricte (cf. section 5.4.1). Cependant, l'étude approfondie de la précision des résultats obtenus selon la méthode d'échantillonnage utilisée et la nature des données analysée permettra d'identifier des seuils de « performances-précision » optimales à proposer aux décideurs dans leurs analyses. Cette étude pourrait permettre de définir un modèle de coût des requêtes effectuées dans le cube original et dans le ClusterCube, afin que le module Switcher (cf. section 5.2.3) puisse rediriger les requêtes vers le cube qui offre le meilleur compromis « performances-précision ».

Ainsi, afin d'améliorer l'approche présentée, nous proposons les perspectives suivantes :

- Une étude comparative des performances de calcul et de précision de différentes méthodes d'échantillonnage dans le SOLAP,
- Une définition d'un modèle de coût des requêtes d'agrégation dans le SOLAP,

- Une implémentation du module Switcher pour permettre l'accès aux données originales et rediriger les requêtes vers le ClusterCube.

6.3.2 Les perspectives plus lointaines

Ces perspectives portent sur la généralisation de notre approche à d'autres champs continus. Par exemple, pour étendre les possibilités d'analyse, nous avons besoin d'effectuer la même démarche que celle proposée dans cette thèse (cf. section 1.3), afin d'intégrer au modèle SOLAP les besoins d'analyse requis pour différents autres types de champs continus (raster, TIN, diagrammes de voronoi, …). Il serait intéressant d'étendre l'architecture SOLAP pour l'analyse continue en conceptualisant chaque type de champs continus dans le SOLAP, puis en proposant des opérateurs d'analyse pour la gestion de leur continuité et leur multi-résolution.

Ceci contribuera à élargir le choix de représentations spatiales mise à disposition des décideurs. En effet, pour que les décideurs puissent exprimer leurs besoins d'analyse spatiale dans le SOLAP, ils doivent pouvoir choisir le type de représentations à utiliser selon l'application à réaliser.

Références

ADMS. 2014. « ADMS 5 model ». Consulté le septembre 29. http://www.cerc.co.uk/environmental-software/ADMS-model.html.

« Agaetis ». 2014. http://www.agaetis.fr/.

Ahmed Taher Omran et Abdullatif Mihdi Buras. 2009. « CSOLAP (Continuous Spatial On-Line Analytical Processing) ». *World Academy of Science, Engineering & Technology*. Issue 30, p256

Ahmed Taher Omran et Maryvonne Miquel. 2005. « Multidimensional Structures Dedicated to Continuous Spatiotemporal Phenomena ». In *Proceedings of the 22Nd British National Conference on Databases: Enterprise, Skills and Innovation*. BNCOD'05. Berlin, Heidelberg: Springer-Verlag, pages 29‑40. doi:10.1007/11511854_3.

ArcGis. 2012. « Pyramides raster ». *ArcGis*. http://help.arcgis.com/fr/arcgisdesktop/10.0/help/index.html#/na/009t00000019000000/

Bartholdi John J. et Paul Goldsman. 2004. « Multiresolution Indexing of Triangulated Irregular Networks ». *IEEE Transactions on Visualization and Computer Graphics, volume* 10, issue 4, pages 484‑495. doi:10.1109/TVCG.2004.14

Bédard Yvan. 1997. « Spatial OLAP ». Forum annuel sur la R-D, Géomatique VI: Un monde accessible, 13-14 novembre

Bédard Yvan, Jiawey HAN et Tim Merrett. 2001. « Fundamentals of Spatial Data Warehousing for Geographic Knowledge Discovery ». In *Geographic Data Mining and Knowledge Discovery*, Research Monographs in GIS, Taylor & Francis, pages 53‑73

Bédard Yvan, Marie-Josée Proulx. 2002. « MODELING MULTIPLE REPRESENTATIONS INTO SPATIAL DATA WAREHOUSES: A UML-BASED APPROACH ». Symposium sur la théorie, les traitements et les applications des données géospatiales, Ottawa

Bédard Yvan, Marie-Josée Proulx et Sonia Rivest. 2005. « Enrichissement du OLAP pour l'analyse géographique: exemples de réalisation et différentes possibilités technologiques. » In *EDA*, 1‑20

Bédard Yvan, Sonia Rivest et Marie-Josée Proulx. 2007. « Spatial. Online Analytical. Processing (SOLAP): Concepts, Architectures, and Solutions ». *Data Warehouses and OLAP: Concepts, Architectures, and Solutions, Idea Group Inc*, pages 298‑319

Bedient Harold A. et James G. Howcroft. 1978. « Data Compression Applied to Nmc Grid-Point Data Fields ». Washington, D. C.

Bimonte Sandro. 2007. « Intégration de l'information géographique dans les entrepôts de données et l'analyse en ligne: de la modélisation à la visualisation ». Lyon: Informatique et Information pour la Société

Bimonte Sandro et Myoung-Ah Kang. 2010. « Towards a Model for the Multidimensional Analysis of Field Data ». In *Proceedings of the 14th East European Conference on Advances in Databases and Information Systems*, ADBIS'10. Berlin, Heidelberg: Springer-Verlag. Pages 58‑72

Bimonte Sandro, Myoung-Ah Kang, Luca Paolino, Monica Sebillo, Mehdi Zaamoune et Giuliana Vitiello. 2014. « OLAPing Field Data: A Theoretical and Implementation Framework ». *Journal Fundamenta Informaticae - Warehousing and OLAPing*

Complex, Spatial and Spatio-Temporal Data, Volume 132 Issue 2, April 2014, Pages 267-290. doi:10.3233/FI-2014-1043.

Bimonte Sandro, Anne Tchounikine et Maryvonne Miquel. 2007. « Spatial OLAP: Open Issues and a Web Based Prototype ». In the *10th AGILE International Conference on Geographic Information Science*, Aalborg University, Denmark. Pages 1-11.

Bimonte Sandro, Anne Tchounikine et Maryvonne Miquel. 2005. « Towards a Spatial Multidimensional Model ». In *Proceedings of the 8th ACM International Workshop on Data Warehousing and OLAP*. DOLAP '05. New York, NY, USA: ACM, pages 39‑46, doi:10.1145/1097002.1097009.

Blakeley Jose A., Per-Ake Larson et Frank Wm Tompa. 1986. « Efficiently Updating Materialized Views ». In *Proceedings of the 1986 ACM SIGMOD International Conference on Management of Data*. SIGMOD '86. New York, NY, USA: ACM, pages 61‑71. doi:10.1145/16894.16861.

Blaschka Markus, Carsten Sapia, G. Hofling et Barbara Dinter. 1998. « Finding your way through multidimensional data models ». In *9th International Workshop on Database and Expert Systems Applications,* pages 198‑203. doi:10.1109/DEXA.1998.707403.

Boulil Kamal. 2012. « Une approche automatisée basée sur des contraintes d'intégrité définies en UML et OCL pour la vérification de la cohérence logique dans les systèmes SOLAP: Applications dans le domaine agri-environnemental ». Thèse de doctorat, France: Laboratoire d'Informatique, de Modélisation et d'Optimisation des Systèmes.

Buzydlowski Jan W., Il-Yeol Song et Lewis Hassell. 1998. « A Framework for Object-Oriented on-Line Analytic Processing ». Proceeding DOLAP '98 Proceedings of the 1st ACM international workshop on Data warehousing and OLAP, New York, NY, USA, pages 10-15. doi:10.1145/294260.294264.

Cao Phuong Thao. 2013. « Approximation of OLAP queries on data warehouses ». Thèse de doctorat, Université Paris Sud - Paris XI.

Chaudhry Omair Z. 2009. « Modelling Geographic Phenomena at Multiple Levels of Detail: A Model Generalisation Approach Based on Aggregation ». Saarbrücken: VDM Verlag, pages 164.

Chaudhuri Surajit, Ravi Krishnamurthy, Spyros Potamianos et Kyuseok Shim. 1995. « Optimizing queries with materialized views ». In *2013 IEEE 29th International Conference on Data Engineering (ICDE)*, IEEE Computer Society, pages 190‑190.

Codd Edgar F., Codd Sally B. et Salley Clynch T. 1993. « *Providing OLAP (On-Line Analytical Processing) to User-Analysts: An IT Mandate* ». Codd & Associates, pages 26.

Helen Couclelis. 1992. « People Manipulate Objects (but Cultivate Fields): Beyond the Raster-Vector Debate in GIS ». Proceedings of the International Conference GIS - From Space to Territory: Theories and Methods of Spatio-Temporal Reasoning on Theories and Methods of Spatio-Temporal Reasoning in Geographic Space, Springer-Verlag London, UK, pages 65-77

Das Gautam. 2009. « *Sampling Methods in Approximate Query Answering Systems* ». The University of Texas at Arlington, USA, pages 6. Doi: 10.4018/978-1-59140-557-3.ch186

De Cola Lee et Nalan Montagne. 1993. « The pyramid system for multiscale raster analysis ». *Computers & Geosciences* 19 (10), pages 1393‑1404. doi:10.1016/0098-3004(93)90057-C.

Elzbieta Malinowski, Esteban Zimányi. 2004. « OLAP Hierarchies: A Conceptual Perspective. ». Advanced Information Systems Engineering. Lecture Notes in Computer Science Volume 3084, pages 477-491. Doi:10.1007/978-3-540-25975-6_34.

Feng Yu et Shan Wang. 2002. « Compressed Data Cube for Approximate OLAP Query Processing ». *Journal of Computer Science and Technology* 17 (5), pages 625‑35. doi:10.1007/BF02948830.

Fidalgo Robson N., Valéria C. Times, Joel Silva et Fernando F. Souza. 2004. « GeoDWFrame: A Framework for Guiding the Design of Geographical Dimensional Schemas ». In *Data Warehousing and Knowledge Discovery*, édité par Yahiko Kambayashi, Mukesh Mohania et Wolfram Wöß, pages 26-37. Lecture Notes in Computer Science 3181. Springer Berlin Heidelberg

Follin Jean-Michel et Alain Bouju. 2007. « Cartographie multi-résolution dans un contexte mobile ». *Revue internationale de géomatique* 17 (2), pages 227‑245. doi:10.3166/rig.17.

Gascueña Concepción M. et Rafael Guadalupe. 2009. « A multidimensional methodology with support for spatio-temporal multigranularity in the conceptual and logical phases ». *IGI Global*. pages 194-230.

Glorio Octavio et Juan Trujillo. 2008. « An MDA Approach for the Development of Spatial Data Warehouses ». In *Data Warehousing and Knowledge Discovery*, édité par Il-Yeol Song, Johann Eder et Tho Manh Nguyen. Lecture Notes in Computer Science 5182. Springer Berlin Heidelberg, pages 23‑32

Gómez Leticia I., Silvia A. Gómez et Alejandro A. Vaisman. 2012. « A Generic Data Model and Query Language for Spatiotemporal OLAP Cube Analysis ». In *Proceedings of the 15th International Conference on Extending Database Technology*. EDBT '12. New York, NY, USA: ACM, pages 300‑311, doi:10.1145/2247596.2247632

Gómez Leticia, Alejandro Vaisman et Esteban Zimányi. 2010. « Physical Design and Implementation of Spatial Data Warehouses Supporting Continuous Fields ». In *Proceedings of the 12th International Conference on Data Warehousing and Knowledge Discovery*. DaWaK'10. Berlin, Heidelberg: Springer-Verlag, pages 25‑39

Goodchild Michael F. 1993. « The state of GIS for environmental problem solving ». *In Environmental Modeling with GIS*, Goodchild , M. F , Parks , B. O and Steyaert , L. T . New York: Oxford University Press, pages 8–15

Gordillo Silvia. 2001. « Modélisation et manipulation de phénomènes continus spatio-temporels ». Thèse de Doctorat "Informatique, Information pour la Société" de Lyon. Université Claude Bernard Lyon I

Gratton Yves. 2002. « Le krigeage: La méthode optimale d'interpolation spatiale ». *Les articles de l'Institut d'Analyse Géographique*, 1

Han Jiawei. 1997. « OLAP mining: An integration of OLAP with data mining ». In Proceedings of the 7th IFIP 2.6 Working Conference on Database Semantics (DS-7), pages 1-9

Han Jiawei, Nebojsa Stefanovic et Krzysztof Koperski. 1998. « Selective Materialization: An Efficient Method for Spatial Data Cube Construction ». In *Research and Development in Knowledge Discovery and Data Mining*, édité par Xindong Wu, Ramamohanarao Kotagiri et Kevin B. Korb. Lecture Notes in Computer Science 1394. Springer Berlin Heidelberg, pages 144‑58

Hellerstein Joseph M., Peter J. Haas et Helen J. Wang. 1997. « Online Aggregation ». In *Proceedings of the 1997 ACM SIGMOD International Conference on Management of*

Data. SIGMOD '97. New York, NY, USA: ACM, pages 171‑182. doi:10.1145/253260.253291.

Inmon William H. 2005. « Building the Data Warehouse ». Édition : 4th Edition. Indianapolis, Ind: John Wiley & Sons.

JAI. « Java Advanced Imaging (JAI) API ». http://www.oracle.com/technetwork/java/javase/tech/jai-142803.html.

Jermaine Christopher. 2003. « Robust Estimation with Sampling and Approximate Pre-aggregation ». In *Proceedings of the 29th International Conference on Very Large Data Bases - Volume 29*, pages 886‑97. VLDB '03. Berlin, Germany: VLDB Endowment.

Jolion Jean-Michel et Rosenfeld Azriel. 1994. « A Pyramid Framework for Early Vision - Multiresolutional Computer Vision ». The Springer International Series in Engineering and Computer Science, Vol. 251.

Ruoming Jin, Glimcher Leo, Jermaine Chris et Agrawal Gagan. 2006. « New Sampling-Based Estimators for OLAP Queries ». In *Proceedings of the 22nd International Conference on Data Engineering, 2006. ICDE '06*, pages 18‑18. doi:10.1109/ICDE.2006.106

Kang Myoung-Ah, Zaamoune Mehdi, Pinet François, Bimonte Sandro et Beaune Philippe. 2013. « Optimisation des performances des opérations d'agrégation au sein des entrepôts de grilles spatialisées ». SAGEO 2013 Conférence Internationale de Géomatique et d'analyse spatiale, Brest, France

Kimball Ralph. 1996. « The Data Warehouse Toolkit: Practical Techniques for Building Dimensional Data Warehouses ». John Wiley & Sons, Inc. New York, NY, USA.

Kobbelt Leif, Jens Vorsatz et Hans-Peter Seidel. 1999. « Multiresolution hierarchies on unstructured triangle meshes ». *Computational Geometry* 14 (1–3), pages 5‑24. doi:10.1016/S0925-7721(99)00032-2

Latecki Longin Jan. 1998. « Discrete Representation of Spatial Objects in Computer Vision ». Computational Imaging and Vision, Vol. 11, Norwell, MA, USA: Kluwer Academic Publishers, pages 216

Laurini Robert, Gordillo Silvia. 2000. « Field Orientation for Continuous Spatio-temporal Phenomena ». International Workshop on Emerging Technologies for Geo-based Applicatons, Ascona, Switzerland. Published by the Swiss Federal Institute of Technology at Lausanne, pages 77-101

Laurini Robert, Pariente Dion. 1996. « Towards a Field-oriented Language: First Specifications ». In Geographic Objects with Indeterminate Boundaries, Edited by Burrough and Frank, Taylor and Francis, pages 225-236

Ledoux Hugo. 2008. « FieldGML: An Alternative Representation For Fields ». In *Headway in Spatial Data Handling*, édité par Anne Ruas et Christopher Gold. Lecture Notes in Geoinformation and Cartography. Springer Berlin Heidelberg, pages 385‑400

Li Jiyuan, Lingkui Meng, Frank Z. Wang, Wen Zhang et Yang Cai. 2014. « A Map-Reduce-Enabled SOLAP Cube for Large-Scale Remotely Sensed Data Aggregation ». *Computers & Geosciences* 70, pages 110‑19. doi:10.1016/j.cageo.2014.05.008

Lin Dekang. 1998. « An Information-Theoretic Definition of Similarity ». In *Proceedings of the Fifteenth International Conference on Machine Learning*. ICML '98. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc, pages 296‑304

Malinowski Elzbieta. 2008. « Advanced Data Warehouse Design: From Conventional to Spatial and Temporal Applications ». Data-Centric Systems and Applications. Berlin: Springer, pages 435

Malinowski Elzbieta et Esteban Zimányi. 2005. « Spatial Hierarchies and Topological Relationships in the Spatial MultiDimER Model ». In *Database: Enterprise, Skills and Innovation*, édité par Mike Jackson, David Nelson et Sue Stirk, 17‑28. Lecture Notes in Computer Science 3567. Springer Berlin Heidelberg. http://link.springer.com/chapter/10.1007/11511854_2

———. 2007. « Logical Representation of a Conceptual Model for Spatial Data Warehouses ». *Geoinformatica* 11 (4), pages 431‑57. doi:10.1007/s10707-007-0022-3

Markl Volker, Ramsak Franck et Bayer Rudolf. 1999. « Improving OLAP performance by multidimensional hierarchical clustering ». In *Database Engineering and Applications, 1999. IDEAS '99. International Symposium Proceedings*, pages 165‑77. doi:10.1109/IDEAS.1999.787265

Maryvonne Miquel, Bédard Yvan, Brisebois Alexandre, Pouliot Jacynthe, Marchand Pierre, Brodeur Jean. 2002. « Modeling Multidimensional Spatio-Temporal Data Warehouses in a Context Of Evolving Specifications ». Symposium sur la théorie, les traitements et les applications des données géospatiales, Ottawa.

McHugh Rosemarie. 2008. « INTÉGRATION DE LA STRUCTURE MATRICIELLE DANS LES CUBES SPATIAUX free ebook download ». Thèse de doctorat, Université Laval, Canada-Quebec.

Mennis Jeremy, Roland Viger et C. Dana Tomlin. 2005. « Cubic Map Algebra Functions for Spatio-Temporal Analysis ». *Cartography and Geographic Information Science* 32 (1): pages 17

Michel Arnaud et Emery Xavier. 2000. « Estimation et interpolation spatiale : méthodes déterministes et méthodes géostatistiques ». Paris: Hermes

Mondrian. « Pentaho Mondrian Documentation ». http://mondrian.pentaho.com/documentation/olap.php.

Mozgeris Gintautas. 2009. « The continuous field view of representing forest geographically: from cartographic representation towards improved management planning ». *SAPI EN. S. Surveys and Perspectives Integrating Environment and Society*, n° 2.2

O'Sullivan David et Unwin David. 2010. « Geographic Information Analysis ». John Wiley & Sons; Édition : 2nd Edition, pages 432

Paolino Luca, Monica Sebillo, Genoveffa Tortora et Giuliana Vitiello. 2010. « Integrating Discrete and Continuous Data in an OpenGeospatial-Compliant Specification ». *Transactions in GIS* 14 (6), pages 731‑53. doi:10.1111/j.1467-9671.2010.01231.x

Parent Christine, Stefano Spaccapietra et Esteban Zimányi. 2006. « Conceptual Modeling for Traditional and Spatio-Temporal Applications: The MADS Approach ». Secaucus, NJ, USA: Springer-Verlag New York, Inc, pages 465

Parker J. Anthony, Robert V. Kenyon et Donald E. Troxel. 1983. « Comparison of Interpolating Methods for Image Resampling ». *IEEE Transactions on Medical Imaging* 2 (1), pages 31‑39. doi:10.1109/TMI.1983.4307610.

Patil G. P. et C. Taillie. 2001. « A Multiscale Hierarchical Markov Transition Matrix Model for Generating and Analyzing Thematic Raster Maps ». University Park, PA., USA: Center for Statistical Ecology and Environmental Statistics, Department of Statistics, Penn State University.

Pariente Dillon. 1994. « Estimation, modelisation et langage de declaration et de manipulation de champs spatiaux continus ». Thèse INSA Lyon.

Pariente Dillon, Servigne Sylvie, Laurini Robert. 1994. « A neural method for geographical continuous field estimation ». Neural Processing Letters 1(2), pages 28-31.

Pedersen Torben Bach et Nektaria Tryfona. 2001. « Pre-Aggregation in Spatial Data Warehouses ». In *Advances in Spatial and Temporal Databases*, édité par Christian S. Jensen, Markus Schneider, Bernhard Seeger et Vassilis J. Tsotras. Lecture Notes in Computer Science 2121. Springer Berlin Heidelberg, pages 460‑478

Pedrini Hélio. 2008. « Multiresolution terrain modeling based on triangulated irregular networks ». *Brazilian Journal of Geology* 31 (2), pages 117‑122.

Pinet François et Michel Schneider. 2010. « Precise Design of Environmental Data Warehouses ». *Operational Research* 10 (3), pages 349‑369. doi:10.1007/s12351-009-0069-z.

Rafanelli Maurizio. 2003. « Multidimensional Databases: Problems and Solutions ». Idea Group Inc (IGI). 340 pages. DOI: 10.4018/978-1-59140-053-0

Rivest Sonia, Bédard Yvan, Proulx Marie-Josée et Nadeau Martin. 2003. « SOLAP: a new type of user interface to support spatio-temporal multidimensional data exploration and analysis ». In *Proceedings of the ISPRS Joint Workshop on Spatial, Temporal and Multi-Dimensional Data Modelling and Analysis, Quebec, Canada*, 2‑3.

Rivest Sonia, Bédard Yvan et Marchand Pierre. 2001. « Toward better support for spatial decision making: defining the characteristics of spatial on-line analytical processing (SOLAP) ». *GEOMATICA-OTTAWA*- 55 (4), pages 539‑55

Rivest Sonia, Bédard Yvan, Proulx Marie-Josée, Nadeau Martin, Hubert Frederic et Pastor Julien. 2005. « SOLAP technology: Merging business intelligence with geospatial technology for interactive spatio-temporal exploration and analysis of data ». *ISPRS Journal of Photogrammetry and Remote Sensing*, Including Theme Section: Advances in Spatio-temporal Analysis and Representation, 60 (1), pages 17‑33. doi:10.1016/j.isprsjprs.2005.10.002

Robinson Arthur H. 1971. « The Genealogy of the Isopleth ». *Cartographic Journal 8(1) , pages 4953*. doi:10.1179/caj.1971.8.1.49.

Sampaio Marcus Costa, André Gomes de Sousa et Cláudio de Souza Baptista. 2006. « Towards a Logical Multidimensional Model for Spatial Data Warehousing and OLAP ». In *Proceedings of the 9th ACM International Workshop on Data Warehousing and OLAP*. DOLAP '06. New York, NY, USA: ACM, pages 83‑90, doi:10.1145/1183512.1183528.

Siqueira Thiago Luís Lopes, Cristina Dutra de Aguiar Ciferri, Valéria Cesário Times, Anjolina Grisi de Oliveira et Ricardo Rodrigues Ciferri. 2009. « The Impact of Spatial Data Redundancy on SOLAP Query Performance ». *Journal of the Brazilian Computer Society* 15 (2), pages 19‑34. doi:10.1007/BF03194499.

Stefanovic Nebojsa, Jiawei Han et Krzysztof Koperski. 2000. « Object-Based Selective Materialization for Efficient Implementation ». *IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING* 12 (6), pages 938 - 958

Stolte Chris, Diane Tang et Pat Hanrahan. 2002. «Query, Analysis, and Visualization of Hierarchically Structured Data Using Polaris». KDD '02 Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, ACM New York, NY, USA, pages 112-122. Doi : 10.1145/775047.775064

Thalmann Daniel. 2003. « Infographie ». *Cours d'infographie destiné aux étudiants de deuxième cycle de l'EPFL, Ecole polytechnique fédérale de Lausanne*

Tomlin C. Dana. 1990. « Geographic Information Systems and Cartographic Modeling ». Englewood Cliffs, NJ: Prentice Hall

Torlone Riccardo. 2003. « Multidimensional Databases ». Edited by Maurizio Rafanelli, pages 69‑90. Hershey, PA, USA: IGI Global

Vaisman Alejandro et Esteban Zimányi. 2009. « A Multidimensional Model Representing Continuous Fields in Spatial Data Warehouses ». In *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. GIS '09. New York, NY, USA: ACM, pages 168‑177. doi:10.1145/1653771.1653797.

Vckovski Andrej. 1995. « Representation of continuous fields ». In *AUTOCARTO-CONFERENCE*, pages 127‑136. Université de Zurich

Vitter Jeffrey Scott et Min Wang. 1999. « Approximate Computation of Multidimensional Aggregates of Sparse Data Using Wavelets ». In *Proceedings of the 1999 ACM SIGMOD International Conference on Management of Data*. SIGMOD '99. New York, NY, USA: ACM, pages 193‑204. doi:10.1145/304182.304199.

Warmer Jos B. et Anneke G. Kleppe. 1998. « The Object Constraint Language: Precise Modeling With Uml (Addison-Wesley Object Technology Series) ». Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA

Wilkie Michelle, Mary Simmons, Tatyana Petrova, John Graas et Brian Mitchell. 2009. « Using SAS ® OLAP Server for a ROLAP Scenario ». *SAS Global Forum*.

Zaamoune Mehdi, Sandro Bimonte, François Pinet et Philippe Beaune. 2013a. « A New Relational Spatial OLAP Approach for Multi-resolution and Spatio-multidimensional Analysis of Incomplete Field Data ». In *ICEIS 2013 - Proceedings of the 15th International Conference on Enterprise Information Systems, Volume 1, Angers, France, 4-7 July, 2013*, édité par Slimane Hammoudi, Leszek A. Maciaszek, José Cordeiro et Jan L. G. Dietz, pages 145‑152. SciTePress. doi:10.5220/0004434501450152.

———. 2013b. « Intégration des données champs continus incomplets dans l'OLAP : de la modélisation conceptuelle à l'implémentation ». *EDA* RNTI-B-9, pages 34‑44.

Résumé :

Les champs continus sont des types de représentations spatiales utilisées pour modéliser des phénomènes tels que la température, la pollution ou l'altitude. Ils sont définis selon une fonction de mapping f qui affecte une valeur du phénomène étudié à chaque localisation p du domaine d'étude. Par ailleurs, la représentation des champs continus à différentes échelles ou résolutions est souvent essentielle pour une analyse spatiale efficace. L'avantage des champs continus réside dans le niveau de détails généré par la continuité, ainsi que la qualité de l'analyse spatiale fournie par la multi-résolution. L'inconvénient de ce type de représentations dans l'analyse spatio-multidimensionnelle est le coût des performances d'analyse et de stockage.

Par ailleurs, les entrepôts de données spatiaux et les systèmes OLAP spatiaux (EDS et SOLAP) sont des systèmes d'aide à la décision qui permettent l'analyse spatio-multidimensionnelle de grands volumes de données spatiales et non spatiales. L'analyse des champs continus dans l'architecture SOLAP représente un défi de recherche intéressant. Différents travaux se sont intéressés à l'intégration de ce type de représentations dans le système SOLAP. Cependant, celle-ci est toujours au stade embryonnaire.

Cette thèse s'intéresse à l'intégration des champs continus incomplets représentés par une grille régulière de points dans l'analyse spatio-multidimensionnelle. Cette intégration dans le système SOLAP implique que l'analyse des champs continus doit supporter :

(i) les opérateurs OLAP classiques, (ii) la vue continue des données spatiales, (iii) les opérateurs spatiaux (slice spatial) et (iv) l'interrogation des données à différentes résolutions prédéfinies.

Dans cette thèse nous proposons différentes approches pour l'analyse des champs continus dans le SOLAP à différents niveaux de l'architecture relationnelle, de la modélisation conceptuelle à l'optimisation des performances de calcul. Nous proposons un modèle logique FISS qui permet d'optimiser les performances d'analyse à multi-résolution en se basant sur des méthodes d'interpolation. Puis, nous exposons une méthodologie basée sur la méthode d'échantillonnage du Clustering, qui permet d'optimiser les opérations d'agrégation des grilles régulières de points dans l'architecture SOLAP relationnelle en effectuant une estimation des résultats.

Mots clés : OLAP spatial, champs continus incomplets, données géographiques à multi-résolutions, entrepôts de données spatiaux, interpolation spatiale

Keywords : Spatial OLAP, incomplete continuous field, multi-resolution geographic data, spatial data warehousing, spatial interpolation