

# Tableaux de bord adaptifs pour le Spatial OLAP

Ali Hassan, Sandro Bimonte

Irstea, TSCF  
9, av. Blaise Pascale, 63178 Aubière, France  
{ali.hassan, sandro.bimonte}@irstea.fr

**Résumé.** Les entrepôts des données spatiales (EDS) et OLAP Spatial (SOLAP) intègrent les outils d'analyse spatiale et de géovisualisation offerts par les Systèmes d'Information Géographique (SIG) aux fonctionnalités OLAP. Peu de travaux étudient les problèmes de géovisualisation dans les systèmes SOLAP et aucun travail ne propose d'outils pour des affichages cartographiques SOLAP lisibles. Par conséquent, les décideurs font manuellement la configuration de la visualisation cartographique dans les outils SOLAP existants. Donc, nous présentons dans cet article une nouvelle méthodologie de géovisualisation pour les résultats des requêtes SOLAP qui donne des cartes lisibles.

## 1 Introduction

Un système SOLAP a été défini comme "*Une plateforme visuelle spécialement conçue pour supporter l'analyse et l'exploration spatio-temporelles rapides et faciles des données multidimensionnelles composées de plusieurs niveaux d'agrégation à l'aide d'affichages cartographiques aussi bien qu'à l'aide de tableaux et diagrammes statistiques*" (Andrienko et Andrienko, 1999). Les systèmes SOLAP permettent d'analyser des grands volumes de données géoréférencées par des opérateurs d'exploration de données interactives et simples.

Les règles de sémiologie permettent une bonne lisibilité des informations (spatiales et alphanumériques) affichées sur une carte. Ces règles dépendent de plusieurs facteurs, tels que le nombre et le type (par exemple, numérique, ordinal, etc.) des variables (c'est-à-dire les éléments d'information représentés) et le type de géométrie (points, lignes, etc.) (Harrie et al., 2015).

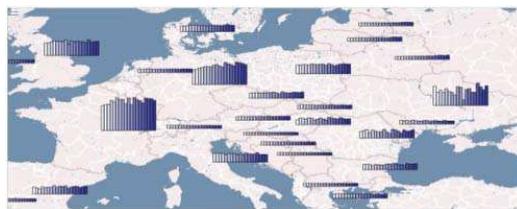
Contrairement aux SIG, les affichages cartographiques SOLAP sont représentés par des cartes interactives créées en ligne à l'aide des opérateurs SOLAP. Le choix de la variable visuelle correcte est effectué manuellement par les décideurs au cours du processus d'analyse en utilisant des assistants (wizards). Ces assistants ne permettent pas la spécification de configurations basée sur les dimensions (par exemple, l'utilisation des cartes animées sur la dimension temporelle), et leur utilisation est assez longue et fastidieuse.

Afin de surmonter ces limitations, nous présentons dans cet article un outil générique pour la visualisation correcte (lisible) des résultats des requêtes SOLAP.

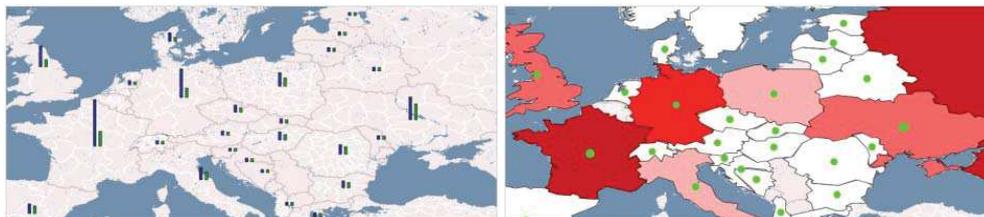
## 2 État de l'art

Les systèmes SOLAP académiques et industriels existants proposent l'utilisation de méthodes de visualisation géographique simples, telles que les cartes choroplèthes, les cartes thématiques et les multicartes (Bimonte, 2014), (Golfarelli et al., 2013), (Malinowski, 2014). En effet, seulement quelques travaux proposent des méthodes de géovisualisation particulières (un état de l'art détaillé se trouve dans (Bimonte, 2014)). Par exemple, (Leonardi et al., 2014) ont étudié une nouvelle méthode de géovisualisation pour les entrepôts des données des trajectoires. (Bédard et al., 2006) ont ajouté des éléments multimédias, tels que des photos, des vidéos, etc., aux données d'entrepôt de données spatiales. Enfin, (Johany et Bimonte, 2016) ont étudié l'utilisation des chorèmes pour enrichir les variables visuelles d'affichage SOLAP. À notre connaissance, seuls (Silva et al., 2012) ont étudié la lisibilité des cartes SOLAP, ce qui a été accompli en fournissant des méthodes de visualisation SOLAP basées sur le clustering pour éviter la surcharge visuelle.

Plusieurs travaux de SIG fournissent des frameworks pour la création de cartes thématiques lisibles (Andrienko et Andrienko, 1999), (Bertin, 1983). Cependant, en dehors de quelques règles simples basées sur "une ou plusieurs variables", les systèmes SOLAP existants ne mettent pas en œuvre ces frameworks. Donc, ils fournissent toujours aux décideurs des cartes illisibles et laissent la responsabilité de sélectionner la configuration de variable visuelle (c'est-à-dire l'association de mesures avec des variables visuelles) pour chaque requête SOLAP.



(a) R1 : une mesure avec 15 ans



(b) R2 : deux mesures avec un an

FIG. 1 – Cartes SOLAP.

## 3 Motivation

À l'aide d'un EDS de FAO (Food and Agriculture Organization of the United Nations) développé dans (Johany et Bimonte, 2016), qui permet d'analyser la surface cultivée et la production agricole par année, pays et culture, il est possible de répondre à des requêtes telles que

"Quelle était la production de blé par pays et par année au cours des 15 dernières années ?" (**R1**). Le résultat compte 15 valeurs (la production par an) par pays. Par conséquent, une carte classique thématique de graphique à barres comme le montre la Figure 1 (a) semble illisible car elle transmet trop d'informations (Bertin, 1983). Donc, une autre méthode de géovisualisation, comme une carte dynamique qui permet d'afficher un ensemble de cartes comme une série temporelle (Harrower et Fabrikant, 2008), devrait être utilisée. Ce problème existe car le nombre de variables visuelles qui peuvent être affichées dans une carte est inférieur au nombre d'informations lisibles affichées dans une table croisée.

Considérons la requête suivante : "Quelles étaient la production de blé et la surface cultivée par pays en 1990 ?" (**R2**). Les résultats de cette requête peuvent être affichés à l'aide d'un graphique à barres avec deux barres (côté gauche de la Figure 1 (b)). Bien que la carte soit lisible en termes de nombre de variables visuelles, elle n'est pas adéquate pour représenter deux mesures différentes ayant des domaines numériques différents (hectares et tonnes). L'utilisation de variables visuelles différentes pour différentes mesures est donc recommandée (Johany et Bimonte, 2016), comme illustré sur le côté droit de la Figure 1 (b).

## 4 Framework pour la visualisation SOLAP

Afin d'éviter la configuration manuelle des affichages cartographiques SOLAP et d'éviter les problèmes d'illisibilité liés au nombre de variables visuelles ( $nb$ ), nous proposons un nouveau framework pour la géovisualisation basé sur des "règles d'affichage". Tout d'abord, il est nécessaire de définir le nombre d'informations maximal à afficher pour un seul élément spatial ( $nb_{max}$ ). Autrement dit, le nombre de variables visuelles qui peuvent être présentées sur la carte dans une seule zone spatiale. Ensuite, un ensemble de règles peut être spécifié. Pour chaque règle, nous devons définir :

**Préférence** : détermine quelle règle doit être utilisée si plusieurs règles sont applicables.

**Conditions** : détermine quand la règle doit être utilisée en fonction du nombre d'information à afficher. Plusieurs conditions pourraient être spécifiées :

- Rang de nombres totaux d'informations à afficher dans une zone spatiale :  $nb = [x_{min}, x_{max}]$ ,
- Rang de nombres de mesures utilisées dans la requête SOLAP :  $n_{Measure} = [x_{min_M}, x_{max_M}]$ ,
- Rang de nombres de membres de chaque dimension (sauf la dimension spatiale) auquel la règle sera appliquée :  $n_{di} = [x_{min_{di}}, x_{max_{di}}]$ .

**Actions** : détermine comment les informations seront visualisées sur la carte (choroplèthe, graphique à barres, carte dynamique, etc.) si toutes les conditions sont atteintes.

**Exemple** : Dans le contexte de l'exemple FAO, afin d'éviter les méthodes de géovisualisation illisibles (voir Figure 1 (a)) ou inadéquates (voir le côté gauche de la figure 1 (b)), nous pouvons définir les règles du Tableau 1. R1 impose l'utilisation d'une carte choroplèthe lorsqu'une seule information doit être affichée dans chaque zone spatiale ( $nb = 1$ ). S'il y a deux informations, chacune pour une mesure différente, alors R2 impose l'affichage de la mesure "Production" par une variable choroplèthe et la mesure "Surface" par des cercles, ce qui correspond au côté droit de la Figure 1 (b). Si  $nb$  est entre deux et quatre, les informations peuvent être affichées par des barres (R3). Pour empêcher la méthode de géovisualisation du côté gauche

## Tableaux de bord adaptifs pour SOLAP

de la Figure 1 (b), R2 a une valeur de préférence deux donc cette règle est préférée à R3 qui a une valeur de préférence trois. Selon R4, si une mesure doit être affichée pour une culture sur plusieurs années, une carte dynamique est utilisée (R4). Si le nombre d'information à afficher sur plusieurs années est supérieur ou égale à cinq, alors plusieurs cartes (multicartes) doivent être utilisées (R5). L'utilisation de "\*" dans la condition signifie qu'il n'y a pas de limite.

Notons que ces règles ont été définies par des experts en EDS en collaboration avec les décideurs.

R1	R2	R3	R4	R5
<b>Préférence = 1</b>	<b>Préférence = 2</b>	<b>Préférence = 3</b>	<b>Préférence = 2</b>	<b>Préférence = 4</b>
<b>Conditions :</b> $nb = [1]$	<b>Conditions :</b> $nb = [2]$	<b>Conditions :</b> $nb = [2, 4]$	<b>Conditions :</b> $nb = [2, *]$	<b>Conditions :</b> $nb = [5, *]$
<b>Actions :</b> Cloropeth map	<b>Actions :</b> $n_{Measure} = [2]$ Production : cloropeth Surface : circles	<b>Actions :</b> Bars	<b>Actions :</b> $n_{Measure} = [1]$ $n_{Time} = [2, *]$ $n_{Crops} = [1]$ Dynamic map	<b>Actions :</b> Multi maps

TAB. 1 – Règles d'affichage pour l'exemple FAO.

Une fois que les règles ont été définies, en tenant compte du nombre de variables visuelles à afficher, le système doit garantir qu'un affichage visuel existe pour chaque table croisée possible. En d'autres termes, nous devons définir un moyen de vérifier que les règles définies permettent d'afficher toutes les tables croisées possibles. Par conséquent, par exemple, considérant que le nombre maximal d'informations à afficher ( $nb_{max}$ ) de 15 est choisi, les règles d'affichage devraient couvrir toutes les analyses possibles qui correspondent à ce nombre.

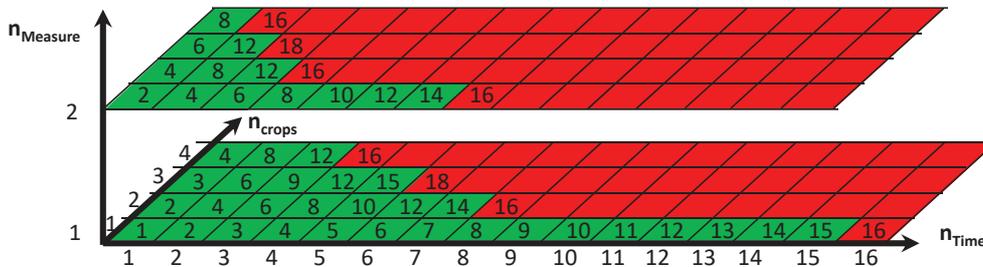


FIG. 2 – Matrice multidimensionnelle (T) avec le nombre d'informations ( $nb$ ).

En utilisant une matrice multidimensionnelle (Figure 2), nous pouvons présenter le nombre d'informations à afficher ( $nb$ ) pour chaque analyse en fonction du nombre de mesures et des membres de chaque dimension non spatiale. Dans notre exemple :

$$nb = n_{Measure} \times n_{Time} \times n_{Crops}$$

Dans la Figure 2, les cellules rouges correspondent au nombre d'information supérieur à  $nb_{max}$ , donc les règles d'affichage ne concernent pas ces cas d'analyse. Par contre, elles devraient couvrir toutes les cellules vertes qui ont un nombre inférieur à  $nb_{max}$ . Pour vérifier cela, nous utilisons l'algorithme ci-dessous.

**Algorithm Vérification**

**Input :** Règles d’affichage,  
matrice multidimensionnelle(T) avec le nombre d’informations

**Output :** result,  
matrice multidimensionnelle (T1) avec les règles applicables

**Begin**

```

1 : result = true
2 : for each Cellule verte  $C_i$  in T :
3 :   found_rule = false
4 :   for each Règle  $R_j$  in Règles d’affichage :
5 :     if ( $C_i.nb < nb_{max}$  and  $C_i.nb \in R_j.nb$ ) then
6 :       applicable_rule = true ;
7 :       for each dimension  $D_k$  of T :
8 :         if ( $C_i.n_{D_k} \notin R_j.n_{D_k}$ ) then
9 :           applicable_rule = false ;
10 :        found_rule = (found_rule OR applicable_rule)
11 :        if applicable_rule then
12 :           $C_i.add(R_j)$  ;
13 : result = (result and found_rule) ;

```

**End**

Cet algorithme prend en entrée les règles d’affichage (Tableau 1) et la matrice multidimensionnelle (Figure 2). Il donne comme résultats :

- Une valeur booléenne indiquant si les règles couvrent tous les cas requis ou non ;
- La matrice multidimensionnelle avec les règles d’affichage applicables (Figure 3).

Pour se faire, l’algorithme vérifie si chaque cellule verte (ligne 2) atteint toutes les conditions (de la ligne 5 à 9) pour chaque règle (ligne 4). Si c’est le cas, la règle est ajoutée à la matrice multidimensionnelle (lignes 11-12). Si l’algorithme ne trouve aucune règle à appliquer pour une cellule, la valeur booléenne résultante devient "false" (ligne 13), indiquant que les règles ne couvrent pas tous les cas requis.

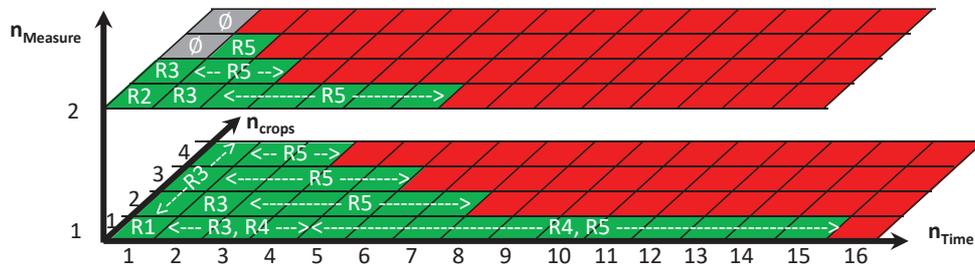


FIG. 3 – Matrice multidimensionnelle (T1) avec les règles d’affichage applicables.

Le résultat de l'application de cet algorithme à notre exemple (c'est-à-dire aux règles d'affichage du Tableau 1 et à la matrice multidimensionnelle de la Figure 2) est présenté dans la Figure 3. Ce résultat montre qu'il existe deux cas d'analyse (cellules grises) avec  $n_{Measure} = 2$ ,  $n_{Time} = 1$  et  $n_{Crops} = 3$  et 4 qui ne sont pas couverts par les règles d'affichage. Cela nécessite soit l'ajout d'une nouvelle règle pour couvrir ces cas, soit une modification des conditions d'une règle existante pour les couvrir.

## 5 Implémentation

Parce que nos travaux précédents (Bimonte, 2014), (Bimonte et al., 2016) ne supportent pas les règles d'affichage, nous les étendons en ajoutant une représentation XML de ces règles (Figure 4). Cette représentation comprend deux balises principales `<Maps>`, `<Display_rule>`.



FIG. 4 – Implémentation des règles d'affichage.

**Balise `<Maps>`** : elle contient plusieurs balises différentes (les noms des balises sont différents `<Map_1>`, `<Map_2>`, ...). Chacune de ces balises définit un type d'affichage et elle définit également ses différents paramétrages nécessaires. Par exemple, la balise `<Map_2>` dans la figure 4 (a) définit un type d'affichage par des cercles et elle définit :

- `<count_levels>` : le nombre de variations de diamètre utilisées avec une distribution uniforme selon les valeurs des mesures,
- `<size_min>` : le diamètre qui correspond à la valeur minimale de mesure,

— <size\_max> : le diamètre qui correspond à la valeur maximale de mesure.

**Balise <Display\_rule>** : elle spécifie les différents règles d’affichage (préférence, conditions et actions) qui précisent dans quel cas d’analyse il faut utiliser quel type d’affichage (prédéfini dans la première balise <Maps>) et quelle couleur pour chaque mesure. La figure 4 (b) spécifie la règle (R2) de Tableau 1. Cette règle d’affichage relie la mesure "Production" au type d’affichage par choroplèthe (Map\_3) en utilisant la couleur "#991055" et la mesure "Surface" au type d’affichage par des cercles (Map\_2) en utilisant la couleur "#109955". Les couleurs sont définies selon le système de codage (RGB).

Une fois que les règles ont été définies par les décideurs et les experts en EDS, notre outil OLAP2MAP garantit la bonne visualisation pour chaque requête SOLAP. Sur la base de nos travaux précédents (Bimonte, 2014), (Bimonte et al., 2016), l’architecture de notre prototype est composée de trois niveaux : l’entrepôt de données spatiales (EDS), le serveur SOLAP et le client SOLAP (Figure 5).

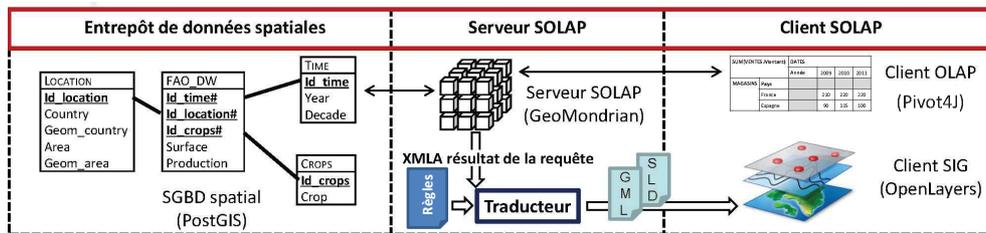


FIG. 5 – l’architecture du prototype.

L’EDS est implémenté à l’aide de "PostGIS" qui est un système de gestion de base de données spatiales. Ce niveau est responsable du stockage des données multidimensionnelles alphanumériques et spatiales.

Le **Serveur SOLAP** est responsable de la mise en œuvre des opérateurs SOLAP pour calculer et naviguer au sein des cubes de données spatiales. Dans notre prototype, le serveur SOLAP utilisé est "GeoMondrian" qui implémente XMLA (XML for Analysis).

Le **client SOLAP** se compose d’un client OLAP et un client SIG :

- Le client OLAP "Pivot4J" est responsable de la visualisation tabulaire de données. Il échange les données avec le serveur SOLAP en utilisant XMLA ;
- Le client SIG "OpenLayers" est responsable de la visualisation cartographique. Pour l’affichage cartographique d’une requête SOLAP basée sur les règles d’affichage, nous fournissons un traducteur automatique de XMLA du résultat de la requête en GML (Geography Markup Language) et SLD (Styled Layer Descriptor), plus de détails dans (Bimonte, 2014). GML et SLD sont des représentations (dérivées du XML) des données spatiales et leur apparence respectivement.

Les règles d’affichage ont été intégrées dans le client SOLAP et sont automatiquement déclenchées lors de chaque requête SOLAP.

Dans la Figure 6, les décideurs visualisent leur table croisée dynamique et leur carte thématique de graphique à barres associée. Ensuite, en modifiant la table croisée pour afficher le résultat d’une requête SOLAP ayant deux mesures et trois ans (flèche 1), la visualisation cartographique est automatiquement adaptée à l’aide de la règle R5 comme décrit précédemment (flèche 2).

Une vidéo d’un exemple de l’utilisation de l’outil apparaît à <https://youtu.be/sxji8pU4WhE>

## Tableaux de bord adaptatifs pour SOLAP

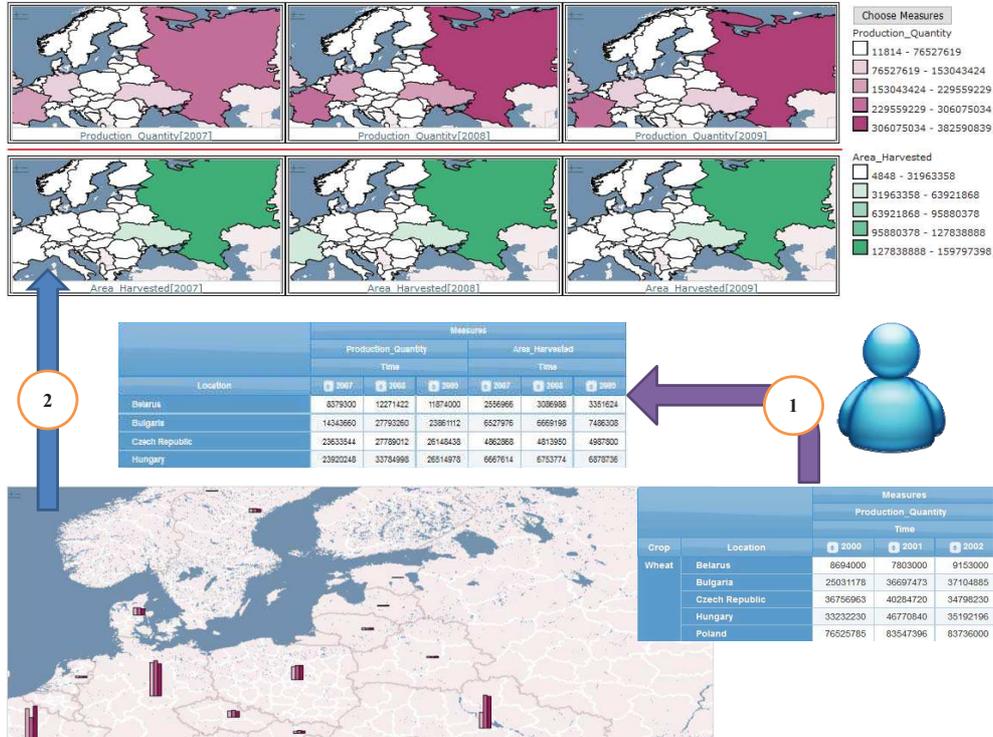


FIG. 6 – Exemple de géovisualisation utilisant des règles d’affichage.

## 6 Conclusion

Motivés par l’importance des outils de géovisualisation dans l’analyse SOLAP, nous avons présenté dans cet article une nouvelle méthodologie pour l’affichage cartographique des résultats de requêtes SOLAP. Cette méthodologie permet l’adaptation automatique de l’affichage cartographique avec les résultats des requêtes SOLAP. Par ailleurs, comme les décideurs ne sont pas toujours des experts SIG, nous envisageons de développer notre outil pour générer automatiquement les cartes et les règles d’affichage à partir du schéma d’EDS. Nous envisageons également d’enrichir notre modèle de règles d’affichage pour qu’il puisse prendre en compte une hiérarchisation de préférences.

## Remerciement

Ce travail a été financé par le projet Captiven d’Irstea dans le cadre de l’ANR 11- CNRT-0003 programme "investissements d’Avenir" valorisation PME.

## Références

- Andrienko, G. L. et N. V. Andrienko (1999). Interactive maps for visual data exploration. *International Journal of Geographical Information Science* 13(4), 355–374.
- Bédard, Y., M.-J. Proulx, S. Rivest, et T. Badard (2006). *Merging Hypermedia GIS with Spatial On-Line Analytical Processing : Towards Hypermedia SOLAP*, Chapter Geographic Hypermedia : Concepts and Systems, pp. 167–187. Springer Berlin Heidelberg.
- Bertin, J. (1983). *Semiology of Graphics*. University of Wisconsin Press.
- Bimonte, S. (2014). A generic geovisualization model for spatial OLAP and its implementation in a standards-based architecture. *Ingénierie des Systèmes d'Information* 19(5), 97–118.
- Bimonte, S., A. Hassan, et P. Beaune (2016). Une architecture orientée services pour l'olap spatial. *Revue des Nouvelles Technologies de l'Information XIIe journées francophones sur les Entrepôts de Données et l'Analyse en Ligne, RNTI-B-12*, 17–24.
- Golfarelli, M., M. Mantovani, F. Ravaldi, et S. Rizzi (2013). Lily : A geo-enhanced library for location intelligence. *DaWaK 8057*, 72–83.
- Harrie, L., H. Stigmar, et M. Djordjevic (2015). Analytical estimation of map readability. *ISPRS International Journal of Geo-Information* 4(2), 418–446.
- Harrower, M. et S. I. Fabrikant (2008). The role of map animation in geographic visualization. In M. Dodge (Ed.), *Geographic Visualization : Concepts, Tools and Applications*, pp. 49–65. Chichester, UK : Wiley.
- Johany, F. et S. Bimonte (2016). *A Framework for Spatio-Multidimensional Analysis Improved by Chorems : Application to Agricultural Data*, pp. 59–80. Cham : Springer International Publishing.
- Leonardi, L., S. Orlando, A. Raffaetà, A. Roncato, C. Silvestri, G. Andrienko, et N. Andrienko (2014). A general framework for trajectory data warehousing and visual olap. *GeoInformatica* 18(2), 273–312.
- Malinowski, E. (2014). Geobi architecture based on free software. In *Geographical Information Systems Trends and Technologies*. Elaheh Pourabbas CRC Press.
- Silva, R., J. a. Moura-Pires, et M. Y. Santos (2012). Spatial clustering in solap systems to enhance map visualization. *International Journal of Data Warehousing and Mining (IJDWM)* 8(2), 23–43.

## Summary

Spatial Data Warehouse (SDW) and Spatial OLAP (SOLAP) systems integrate spatial analysis and geovisualization tools offered by Geographic Information Systems to OLAP functionalities. Few works investigate geovisualization issues in SOLAP systems, and no works investigate tools for readable SOLAP cartographic displays. Therefore, decision-makers manually make the configuration of the cartographic visualization in existing SOLAP tools. Therefore, we present in this paper a new geovisualization methodology for SOLAP queries that yields readable maps.

