

Une architecture orientée services pour l'OLAP Spatial

Sandro Bimonte*, Ali Hassan*, Philippe Beaune**

*Irstea, TSCF

24, av. Blaise Pascale, 63172 Aubière, France

{sandro.bimonte, ali.hassan}@irstea.fr

**Agaetis

10 Rue Evariste Galois, 63000 Clermont-Ferrand, France

pbeaune@agaetis.fr

Résumé. Un système OLAP Spatial (SOLAP) vise à analyser interactivement les données géoréférencées. Il permet aux décideurs d'explorer et de visualiser les entrepôts de données spatiales en utilisant des tables multidimensionnelles « pivot table » et d'affichage cartographique des faits sur des cartes interactives. Dans cet article, nous présentons un prototype pour la visualisation cartographique des requêtes SOLAP. Ce prototype se base sur une architecture orientée services (SOA). Dans ce prototype, le client SOLAP est totalement basé sur des représentations de données standard et des services web de visualisation cartographique.

1 Introduction

Étant donné que les systèmes OLAP ne permettent pas d'intégrer les données spatiales dans le processus d'analyse et d'exploration multidimensionnelles, les systèmes Spatial OLAP (SOLAP) ont été introduits (Bédard et al., 2006). Ils intègrent les fonctionnalités OLAP et des Systèmes d'Information Géographique (SIGs) dans un cadre unique afin de profiter des capacités d'analyse associées aux données spatiales. Les attributs géométriques des données géographiques permettent une représentation cartographique (en plus de la table multidimensionnelle) des résultats des requêtes SOLAP. Les cartes interactives sont les principaux instruments utilisés pour l'analyse de données spatio-temporelles (Andrienko et al., 2003). Les clients SOLAP permettent d'effectuer les opérations OLAP par de simples interactions de l'utilisateur avec les cartes (Bimonte, 2010). Les représentations tabulaire et cartographique sont synchronisées (Rivest et al., 2005). Autrement dit, une action (par exemple, un drill-down ou un roll-up) sur un composant se propage aux autres composants.

Les entrepôts de données spatiales (EDSs) et les systèmes SOLAP sont généralement implémentés en utilisant une architecture relationnelle à plusieurs niveaux. Le premier niveau est l'EDS, qui est généralement mis en œuvre en utilisant un SGBD spatial pour représenter les mesures et les dimensions (Kimball, 1996). Le deuxième niveau est le serveur SOLAP qui met en œuvre le modèle spatio-multidimensionnel et les opérateurs SOLAP. Le troisième niveau est le client SOLAP qui permet de visualiser les requêtes SOLAP.

Dans cette architecture complexe, les services web jouent un rôle crucial. Une architecture orientée services (SOA) est un modèle de conception où chaque niveau est défini par un ensemble de services web (Bell, 2010). Un des avantages les plus importants de la SOA est l'interopérabilité entre les différents outils sans programmation supplémentaire. Dans le contexte des systèmes OLAP, XMLA « XML for Analysis » permet l'utilisation d'outils OLAP dans un environnement SOA en représentant les tables multidimensionnelles résultant des requêtes OLAP, ce qui permet d'échanger les données entre le serveur OLAP et le client OLAP.

Dans cet article, nous présentons un prototype pour mettre en œuvre une solution SOA afin de visualiser les résultats des requêtes SOLAP. Comme le SOLAP est une extension spatiale d'OLAP, nous proposons donc de suivre la même stratégie utilisée dans le contexte OLAP :

- l'utilisation d'une architecture relationnelle à plusieurs niveaux ;
- l'utilisation des modèles de données standard, tels que XMLA pour représenter le résultat d'une requête comme un tableau et GML et SLD pour représenter les données spatiales et leurs styles d'affichage lors de l'utilisation des services web.

Cet article est organisé comme suit. La section 2 présente l'état de l'art. La section 3 présente notre cas d'étude. Puis nous présentons l'architecture de notre prototype (section 4).

2 État de l'art

Peu de travaux étudient l'utilisation des services web pour les systèmes SOLAP. Un état de l'art détaillé est présenté dans (Bimonte, 2014). (Dubé et al., 2009) proposent un modèle XML pour l'échange des cubes de données SOLAP. Ce format XML peut être utilisé dans le développement de services web adaptés à diverses situations. (Silva et al., 2005) présente un framework de services web pour le traitement des données géographique et multidimensionnel. L'objectif principal de ce travail est d'utiliser des technologies standardisées tels que les services web, Java et XML. (Golfarelli et al., 2013) ont présenté une architecture SOA basée sur Map Server et Oracle, mais la présentation cartographique des résultats des requêtes SOLAP est fournie par une API Java ad hoc. Pour conclure, les travaux existants adoptent partiellement l'approche SOA car la visualisation cartographique pour les résultats des requêtes SOLAP n'est pas implémentée en utilisant des services web.

3 Cas d'étude

Nous utilisons un exemple d'EDS chargé en utilisant des données ouvertes de la FAO (Food and Agriculture Organization of the United Nations) (FAO, 2016). La Figure 1 illustre le schéma spatio-multidimensionnel de cet exemple en utilisant le profil UML présenté dans (Boulil et al., 2015). Ce modèle comprend une dimension temporelle « Time », une dimension spatiale « Location », une dimension représentant les cultures « Crops » et deux mesures numériques représentant la surface cultivée « surface » et la production agricole « production ». Les dimensions spatiale et temporelle ont des hiérarchies regroupant les pays dans des zones, et les années par décennies.

Notre prototype doit prendre en charge les besoins de visualisation cartographique différents pour répondre à plusieurs types de requêtes SOLAP, comme par exemple : *la surface culti-*

vée pour chaque zone et ses pays en 2015. Le prototype se base sur un modèle de données générique pour la visualisation cartographique des requêtes SOLAP (Bimonte, 2014).

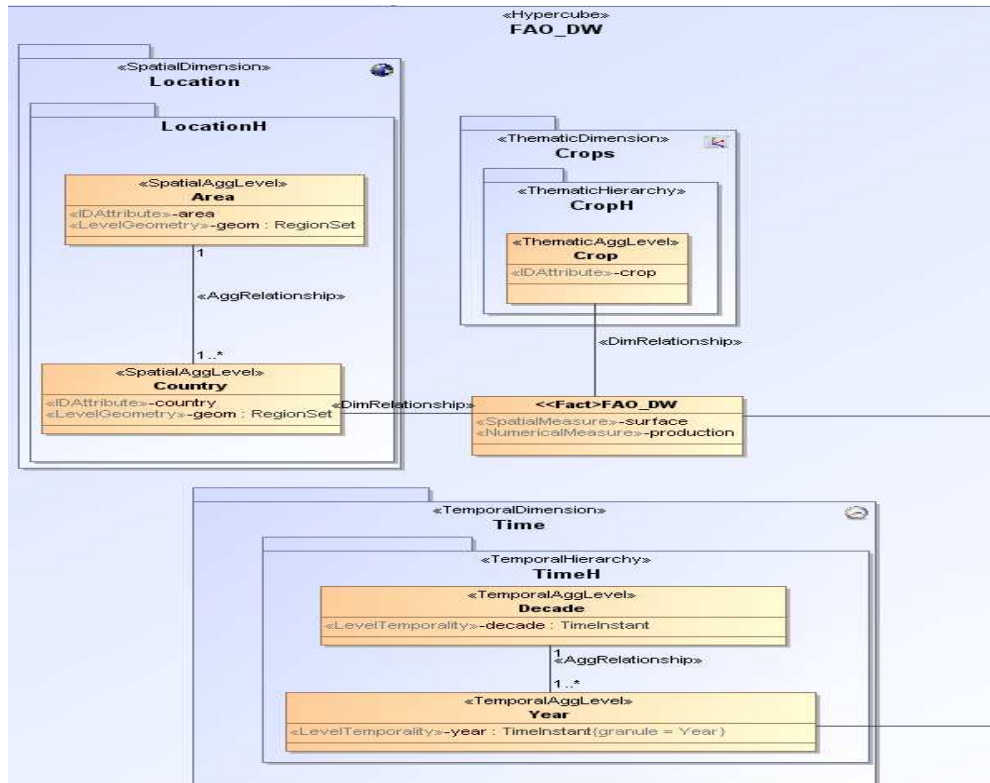


FIG. 1 – Entrepôt de données spatiales de FAO (FAO, 2016).

4 L'architecture du prototype

L'idée principale de l'implémentation de notre prototype est d'utiliser une architecture totalement basée sur les standards SLD et GML (qui sont utilisés par les services web cartographiques standard), et XMLA (représentant le résultat d'une requête dans la table multidimensionnelle). C'est pourquoi nous fournissons un traducteur automatique de XMLA en SLD-GML.

Un « Styled Layer Descriptor » (SLD) est un schéma XML qui décrit le style des couches de carte, tandis que « Geography Markup Language » (GML) permet d'exprimer des caractéristiques géographiques. GML permet de représenter tous les types spatiaux (e.g. point, ligne, polygone, etc.). Une utilisation typique de SLD est destinée aux WMS « Web Map Service » pour que ces derniers puissent interpréter efficacement une couche de donnée spécifique. Donc, nous utilisons donc GML pour représenter les données spatiales, et SLD pour son apparence.

SOA pour SOLAP

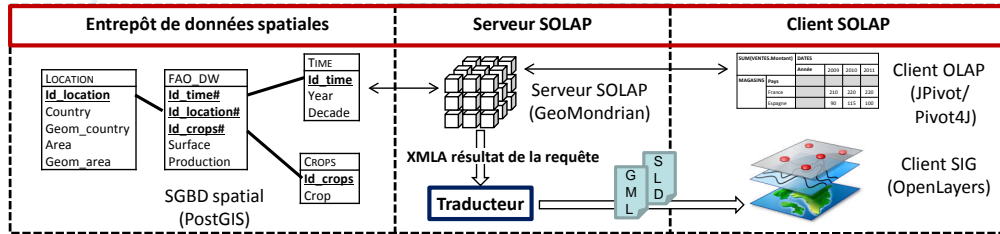


FIG. 2 – l'architecture du prototype.

L'idée principale de notre traducteur de XMLA en SLD-GML (Figure 2) consiste à analyser le fichier XMLA représentant le résultat de la requête SOLAP et générer un fichier xml composé des mesures requises dans la requête. Ce fichier xml contient, pour chaque mesure, son nom, sa valeur et les données spatiales correspondantes. Une fois le fichier des mesures spatiales généré, il est utilisé pour générer à la fois les fichiers GML et SLD, qui contiennent les données spatiales et le style qui sera appliqué respectivement.



FIG. 3 – Extrait des fichiers GML et SLD.

Le fichier GML contient plusieurs balises qui sont des attributs des données spatiales (id, nom, valeur, coordonnées, etc.) (Figure 3 (a)). Le fichier SLD contient des règles qui précisent dans quel cas, un style spécifique doit être appliqué à des données spatiales (Figure 3 (b)). Par exemple, l'utilisation de la couleur rouge lorsqu'une valeur est supérieure à seuil précis. Dans ce travail, nous avons mis en place de cartes choroplèthes. Toute autre implémentation

de visualisation peut être facilement définie en utilisant un nouveau modèle SLD sans aucun impact sur les systèmes SOLAP et SIG utilisés.

L'architecture de notre prototype est composée de trois niveaux : l'entrepôt de données spatiales, le serveur SOLAP et le client SOLAP (Figure 2)

L'entrepôt de données spatiales (EDS) est réalisé en utilisant PostGIS qui est un SGBD spatial. Ce niveau est le responsable du stockage des données multidimensionnelles alphanumériques et spatiales. Ici, les données sont modélisées selon la méthode R-OLAP classique (Kimball, 1996) où les faits et les dimensions sont traduits au niveau logique sous la forme de relations. La Figure 2 (à gauche) représente le modèle logique basé sur le modèle conceptuel de la Figure 1. Ce modèle relationnel est une solution standard pour tous les serveurs OLAP, ce qui signifie que tous les SGBDs spatiaux pourraient éventuellement être utilisés.

Le Serveur SOLAP est le responsable de la mise en œuvre des opérateurs SOLAP pour calculer et naviguer au sein des cubes de données spatiales. Ces cubes de données sont représentés par un schéma SOLAP qui définit une correspondance entre les concepts de cube de données spatiales (dimensions, hiérarchies, mesures, etc.) et le schéma relationnel (tables et colonnes utilisées pour représenter les faits et les dimensions). Dans notre prototype, le serveur SOLAP utilisé est GeoMondrian. GeoMondrian implémente le XMLA. Il est important de souligner que tous les serveurs SOLAP compatibles avec XMLA peuvent être utilisés sans aucun effort de programmation supplémentaire.

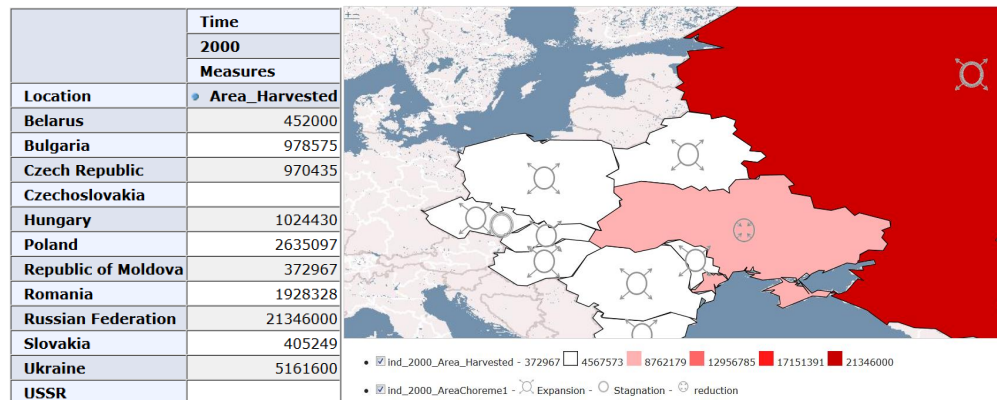


FIG. 4 – Le client SOLAP (avec le client OLAP JPivot).

Le client SOLAP est composé d'un client OLAP et un client SIG en utilisant des services web standards.

- Le client OLAP est le responsable de la visualisation tabulaire de données en utilisant XMLA. Dans notre prototype, pour montrer l'interopérabilité entre les différents outils, nous utilisons deux clients OLAP (JPivot ou Pivot4J) qui permettent de déclencher les opérateurs OLAP par une simple interaction avec la table multidimensionnelle. JPivot et Pivot4J supportent MDX et XMLA ;
- Le client SIG est le responsable de la visualisation cartographique. Cette visualisation est implémentée dans notre prototype par le client web OpenLayers. Puisque les fi-

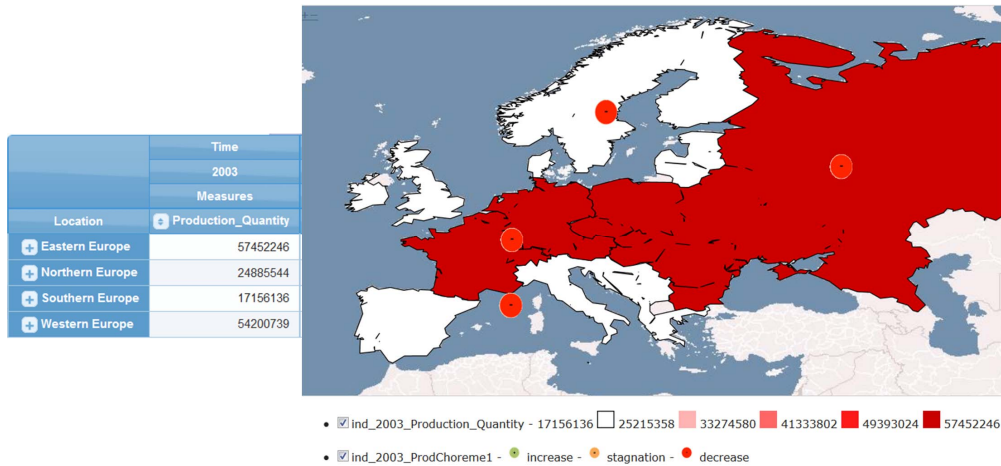


FIG. 5 – Le client SOLAP (avec le client OLAP Pivot4J).

chiers SLD et GML générés après l'exécution d'une requête sont envoyés directement à OpenLayers, le résultat de la requête peut être affiché en même temps en utilisant JPivot/Pivot4J et OpenLayers (Figure 4 et Figure 5).

JPivot/Pivot4J et OpenLayers peuvent être simplement remplacés par d'autres clients OLAP et SIG compatibles avec XMLA et WMS sans aucun effort de programmation supplémentaire.

Des exemples d'utilisation de notre outil avec le client OLAP (JPivot) sont disponibles ici :

<https://www.youtube.com/watch?v=srx0Hm7BPkw> et

https://www.youtube.com/watch?v=iy_rnUkDlr0.

Un autre exemple d'utilisation de notre outil avec le client OLAP (Pivot4J) est disponible ici :

<https://www.youtube.com/watch?v=rxl9ONz0A1I>

5 Conclusion

Dans cet article nous avons présenté un prototype SOLAP qui s'appuie sur une approche standard pour la mise en œuvre d'une solution SOA afin de visualiser les résultats des requêtes SOLAP. Ce prototype se base sur un modèle de données générique pour la visualisation cartographique des requêtes SOLAP (Bimonte, 2014). Dans ce prototype, le client SOLAP repose entièrement sur les représentations de données standard (par exemple, XMLA, SLD, etc.) et les services web de la visualisation cartographiques (XMLA, WMS, etc.). Cette architecture SOA permet l'utilisation de tous les outils d'OLAP et de SIG compatibles avec XMLA et SLD respectivement. En outre, les politiques de visualisation cartographique des résultats SOLAP peuvent être facilement modifiées et adaptées à certains domaines d'application ou décideurs. Notre proposition de SOA est validée dans une architecture OLAP relationnelle classique en utilisant PostGIS, GeoMondrian, JPivot/Pivot4J et OpenLayers.

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, N., G. Andrienko, et P. Gatalsky (2003). Exploratory spatio-temporal visualization : an analytical review. *Journal of Visual Languages & Computing* 14(6), 503–541.
- Bédard, Y., S. Rivest, et M. José Proulx (2006). Spatial on-line analytical processing (solap) : Concepts, architectures, and solutions from a geomatics engineering perspective. *Data Warehouses and OLAP : Concepts, Architecture, and Solutions* 14, 298–319.
- Bell, M. (2010). *SOA Modeling Patterns for Service Oriented Discovery and Analysis*. Wiley Publishing.
- Bimonte, S. (2010). A web-based tool for spatio-multidimensional analysis of geographic and complex data. *IJAEIS* 1(2), 42–67.
- 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.
- Bouilil, K., S. Bimonte, et F. Pinet (2015). Conceptual model for spatial data cubes : A UML profile and its automatic implementation. *Computer Standards & Interfaces* 38, 113–132.
- Dubé, E., T. Badard, et Y. Bédard (2009). XML encoding and web services for spatial OLAP data cube exchange : an SOA approach. *CIT* 17(4), 347–358.
- FAO (2016). <http://data.fao.org/statistics>.
- Golfarelli, M., M. Mantovani, F. Ravaldi, et S. Rizzi (2013). Lily : A geo-enhanced library for location intelligence. *DaWaK* 8057, 72–83.
- Kimball, R. (1996). *The Data Warehouse Toolkit: Practical Techniques for Building Dimensional Data Warehouses*. John Wiley & Sons.
- Rivest, S., Y. Bédard, M.-J. Proulx, M. Nadeau, F. Hubert, et J. Pastor (2005). Solap technology: Merging business intelligence with geospatial technology for interactive spatio-temporal exploration and analysis of data. *ISPRS* 60(1), 17–33.
- Silva, J., V. C. Times, R. N. Fidalgo, et R. S. M. Barros (2005). Providing geographic-multidimensional decision support over the web. In *7th Asia-Pacific Web Conference*, Volume 3399 of *Lecture Notes in Computer Science*, pp. 477–488.

Summary

Spatial OLAP systems aim to interactively analyze geo-referenced data. They allow decision-makers to explore and visualize spatial data warehouse using pivot tables and cartographic representation of facts on interactive maps. In this paper we present a prototype for the cartographic visualization of SOLAP queries. This prototype is based on service oriented architec-

SOA pour SOLAP

ture (SOA). In this prototype, the SOLAP client is totally based on standard data representations and cartographic visualization web services.