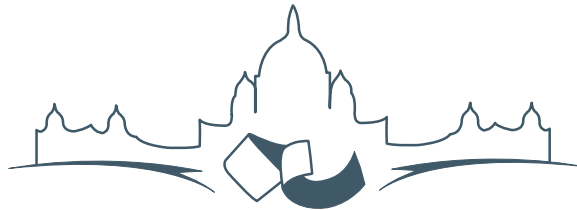

Journal de l'OSGeo

Le journal de la Fondation Open Source Geospatial

Volume 3 / Décembre 2007



2007 FREE AND OPEN SOURCE SOFTWARE
FOR GEOSPATIAL (FOS4G) CONFERENCE
VICTORIA CANADA  SEPTEMBER 24 TO 27, 2007

Compte rendu du FOS4G 2007

Intégration & Développement

- Portable GIS : SIG sur une clé USB
- Génération Automatique d'Applications SIG / Base de données sur Internet
- db4o2D - Extension de Base de données Orientée Objet pour les types géospatiaux 2D
- Google Summer of Code pour la géomatique

Intérêt thématique

- Approche générique pour la gestion de standards de métadonnées
- Vers des services web dédiés à la cartographie thématique
- Interopérabilité pour les données géospatiales 3D
- Un Service Web orienté modèle pour une interopérabilité sémantique améliorée
- Spatial-Yap : un système de base de données spatialement déductif

Études de cas

- Le Projet DIVERT : Développement de Télématiques Inter-Véhicules Fiabiles
 - GRASS et la Modélisation des Risques Naturels
 - Une Base de Données Spatiales pour l'Intégration des Données du Projet de Gestion des Ressources Naturelles du Rondonia
 - GeoSIPAM : Logiciel libre et Open Source appliqué à la protection de l'Amazonie brésilienne
 - Le Système de Suivi de la Déforestation Amazonienne
-

From the Editor...

Comme promis, le voici — les résumés des conférences du plus gros et du meilleur FOSS4G jusqu'ici. C'est une édition spéciale du Journal de l'OSGeo, dédié pour vous apporter un échantillon de la centaine de conférences présentés lors de l'événement. Pour plus d'information, résumés des conférences et des discussions lisez les différents blogs et rapport disponible sur la [page de presse](http://page.de.presse) du site du foss4g2007.org.

Le comité organisateur local est déjà prêt pour préparer l'événement de l'année prochaine au Cap, Afrique du Sud. Pour plus d'information rendez-vous sur le site foss4g2008.org. Pendant que vous lirez ceci, les candidats pour le FOSS4G 2009 écrivent rapidement leur offre pour héberger l'événement — bon courage à eux !

La prochaine sortie du Journal redeviendra normale. Ceci est une invitation à écrire un article, des exemples de code, des cours, études de cas ou sur d'autres thématiques. Si vous êtes intéressé pour soumettre des articles, ajoutez vous s'il vous plait à la page wiki [Volume 4](http://Volume4)¹ et un éditeur vous contactera. Contactez moi directement si vous vous demandez comment votre article pourrait convenir au Journal.

Come toujours merci à l'équipe Éditoriale, relecteurs et aux **37 contributeurs** de ce volume. Bonne lecture !

Tyler Mitchell

Rédacteur en Chef

<http://www.osgeo.org>

[tmitchell AT osgeo.org](mailto:tmitchell@osgeo.org)



Table des matières

From the Editor...	1
Sponsors & Supporters du FOSS4G 2007	2
Portable GIS : SIG sur une clé USB	3
Génération Automatique d'Applications SIG / Base de données sur Internet	6
db4o2D - Extension de Base de données Orientée Objet ...	20
Google Summer of Code pour la géomatique	24
Approche générique pour la gestion de standards de métadonnées	27
Vers des services web dédiés à la cartographie thématique	34
Interopérabilité pour les données géospaciales 3D	39

Un Service Web orienté modèle pour une interopérabilité sémantique améliorée	43
Spatial-Yap : un système de base de données spatiales déductive	51
Le Projet DIVERT : Développement de Télématiques Inter-Véhicules Fiables	56
GRASS et la Modélisation des Risques Naturels	62
Une Base de Données Spatiales pour l'Intégration des Données du Projet de Gestion des Ressources Naturelles du Rondonia	70
GeoSIPAM	75
Le Système de Suivi de la Déforestation Amazonienne	82

¹http://wiki.osgeo.org/index.php/Journal_Volume_4

Sponsors & Supporters du FOSS4G 2007

Réflexions de deux supporters

La conférence autour de la Géomatique Libre et Open Source (The Free and Open Source Software for Geospatial Conference - FOSS4G), avec la participation de la Fondation Géospatiale Open Source (OS-Geo), est devenu l'événement majeur pour la communauté du Géospatiale Open Source. Cette année, on m'a demandé de faire un résumé pour la session de clôture. Voici les thèmes que j'ai présenté, qui sont ordonnés selon leurs maturités.

Les expériences de ces dernières années (développement de projets, création de l'OSGeo, et bien d'autres choses) ont donné à la communauté du FOSS4G une confiance que je n'avais pas sentie auparavant. Du fait que je n'ai pu participer à quelques unes des rencontres de cette série, notamment la dernière, la session de m'est apparue être une véritable conférence. Je vois un changement à la réponse à la question que j'ai posée aux participants de ces dernières années : "Quel logiciel de stockage utilisez-vous?". Ce qui me pousse à dire que la proportion de participants nouveaux dans les SIG ou le géospatial a augmenté. La communauté FOSS4G est "juste comme" le reste de la communauté SIG. Je note que la poursuite des modèles d'entreprises efficaces et du

financement du développement est plus sereine que par le passé. La demande Open Source Géospatiale dans le milieu éducatif est grandissante.

Adena Schutzberg
Executive Editor, Directions Magazine

Le projet de planification ouvert (Open Planning Project) souhaite remercier les organisateurs de la conférence FOSS4G 2007 et plus particulièrement les participants pour en avoir fait un événement magnifique. Elle a été un bon moment pour prendre contact avec des collègues et pour rencontrer les nouveaux utilisateurs de nos technologies collectives.

Nous avons été enthousiasmés par la croissance de la popularité de Geoserver et OpenLayers, et avons apprécié l'opportunité qui nous a été offerte de présenter le développement de nouvelles caractéristiques que nous avons intégrées dans ces projets. Une nouvelle bonne année s'annonce pour les avancées dans le stockage de données géographiques et nous reviendrons avec encore plus de développements enthousiasmants à la prochaine conférence.

Chris Holmes, The Open Planning Project



Merci à tous les sponsors du FOSS4G 2007 pour leur soutiens !

Intégration & Développement

Portable GIS : SIG sur une clé USB

Jo Cook, traduit par Ludovic Granjon

Résumé

Il s'agit d'une suite de logiciels SIG qui peut être lancée à partir d'une clé USB (et par conséquent totalement portable) sur des ordinateurs tournant sous Microsoft Windows. Cela a conduit à un pack qui peut être rapidement et facilement déployé par des personnes sans expérience préalable ; dans un éventail d'environnements différents ; et fournit une méthode simple et économe en énergie pour les entreprises ou personnes afin d'évaluer ces packs en comparaison des alternatives propriétaires coûteuses.

Introduction

Le terme "SIG Portable" se réfère à une suite complète de logiciel de SIG qui peut être lancée à partir d'une clé USB. Cela permet d'être utilisé sur n'importe quel PC sous Windows, sans nécessiter installation ou configuration. Quelques modifications, qui étaient nécessaires pour permettre aux logiciels de fonctionner dans ce système, ont été apportées au code existant, tel que retirer la nécessité d'une lettre fixe pour le lecteur. De plus les packs individuels ont été intégrés dans une seule "suite", avec un système de menus pour les exploiter et une documentation complète.





Les logiciels open source sont souvent difficiles à installer comparés aux alternatives propriétaires de Windows. Les développeurs semblent souvent supposer un meilleur niveau préalable de connaissance que le nouvel utilisateur moyen, et les options pour obtenir de l'aide, tel que les listes de diffusion et les groupes de discussions en ligne peuvent être intimidants. Je crois, par expérience personnelle et par témoignage, que beaucoup d'utilisateurs sont rebutés par ce qu'ils n'arrivent pas à l'installer, encore moins à l'utiliser."

Les CDS Live sont souvent utilisés dans ces situations mais l'avantage de l'approche "Portable GIS" est qu'elle travaille dans le même système opérationnel que l'utilisateur, avec un accès vers leurs fichiers systèmes (et données) plutôt que dans un environnement indépendant.

Il y a deux autres avantages de l'approche "Portable GIS". Le fait qu'elle tourne sur clé USB implique qu'elle peut être déployée rapidement et facilement dans de nombreuses situations, et le fait qu'elle ne requiert pas d'utilisateur pour l'installation et la configuration fait d'elle une proposition attractive pour les entreprises et les personnes souhaitant évaluer les SIG open source.

Produits inclus sur la clé

Les programmes peuvent être séparés en deux modules, bureautique et basé sur le web. Tous les aspects de la suite sont open source, incluant la documentation et le système de menus. Les programmes sont les suivants :

Bureautiques : GRASS, QGIS, FWTools

Basés sur le web : XAMPPlite (full apache/MySQL/PHP web server stack), PostgreSQL avec PostGIS, MapServer, OpenLayers, Tilecache.

La documentation a été écrite sous la forme d'un petit wiki. Il s'agit d'une seule page html wiki, donc elle peut être éditée par les utilisateurs pour ajouter leurs propres notes. Le système de menus est basé sur le "daily cup of tech" menu, construit par Tim

Fehlman. Il est construit en utilisant AutoIt, et fonctionne à partir d'un système de plateau. Il fournit un lien vers l'installation des programmes (lorsque nécessaire), et vers le démarrage et l'arrêt des exécutables pour chaque pack. Il permet également à l'utilisateur de naviguer vers le lecteur de fichier système de manière normale.

Problèmes et courbes d'apprentissage

Alors que quelques programmes, tel que XAMPPLite, sont conçus pour être lancés de cette manière, et d'autres, tel que QGIS sont connus pour fonctionner, d'autres ont besoin de modifications pour fonctionner sans la nécessité d'une lettre de lecteur fixe. Dans beaucoup de cas, ceci est obtenu en éditant le fichier batch qui installe l'environnement de travail du programme, ou en installant de manière traditionnelle sur un PC sous Windows et en copiant le répertoire du programme dessus (l'approche de test et d'erreurs).

De plus, l'installation native de GRASS sous Windows est complexe. La compréhension de la syntaxe du fichier batch, et en particulier comment les programmes comme GRASS l'utilise, était le principal problème qui a été surmonté dans la construction de la suite "Portable GIS".

Dans d'autres cas, tel que PostgreSQL, où traditionnellement, un utilisateur particulier est requis pour lancer le programme, la situation semblait désespérée jusqu'à ce qu'un post arrivé par hasard sur la liste annonce que ce n'était plus le cas. PostgreSQL peut maintenant être installé et lancé par l'utilisateur connecté sur le PC, et ne nécessite plus d'être lancé comme un service. Cela a rendu le processus d'installation sur une clé USB relativement simple.

Quelques problèmes demeurent. Les machines windows en travail ou les environnements standards sont souvent bloqués pour prévenir le lancement de services ou le réglage des variables d'environnement. Dans ces cas, beaucoup d'aspects de la suite ne fonctionneront pas. Actuellement, il est difficile de voir la manière de contourner la situation, autrement que par porter cela à l'attention des utilisateurs finaux !

la route à suivre

En plus de la rationalisation de la suite, il y a plusieurs améliorations à faire. Plus de programmes peuvent être ajoutés, lorsque cela est nécessaire. Une façon pour cela sera d'augmenter la taille du package (actuellement 800 MB).

Une procédure simple et transparente pour mettre à jour les paquets individuels doit être identifiée. Une documentation détaillée pour mettre à jour ces paquets doit être fournie, incluant ces fichiers qui ne doivent pas être mis à jour (ceux qui ont été modifiés pour permettre au programme de fonctionner de façon portable par exemple) ou l'auteur doit prendre ces responsabilités pour produire une nouvelle suite lorsque les programmes sont mis à jour.

Il est possible d'améliorer l'installation en retirant les bibliothèques dupliquées et donc réduire la taille globale du pack. Par exemple, mapserver est inclus comme un programme web et également dans l'application FWTools, et GRASS est inclus de fa-

çon propre et comme une partie de l'installation de QGIS. Cependant, ceci empêcherait les utilisateurs de mettre à jour les programmes de manière individuelle tel que cela a été souligné plus haut.

Des demandes ont été faites pour fournir des versions valables pour différentes plateformes. C'est un challenge intéressant, mais dans une certaine limite, il est supposé que les utilisateurs de Linux sont plus familiers avec l'installation de logiciels open source et la communauté autour d'eux !

A cet instant, la suite n'est pas disponible publiquement à cause sa taille importante et de la bande passante nécessaire pour l'hébergement. Bien sûr si elle doit être utilisée, alors elle doit être hébergée. Différentes options pour cela sont disponibles et sont explorées en ce moment. Pour plus d'informations, allez voir la discussion sur le blog Archaeogeek.²

Jo Cook

Oxford Archaeology North

<http://www.oxfordarchaeology.co.uk>

[j.cook AT oxfordarch.co.uk](mailto:j.cook@oxfordarch.co.uk)

²La discussion sur le blog Archaeogeek : <http://www.archaeogeek.com/blog/portable-gis/>

Génération Automatique d'Applications SIG / Base de données sur Internet

Nirut Chalainanont, Junya Sano and Toshimi Minoura,
traduit par Marie Silvestre

Résumé

Nous avons développé des applications Internet SIG / base de données (WebGD) qui permettent aux utilisateurs d'ajouter, interroger, mettre à jour, et supprimer des entités géographiques ainsi que leurs données attributaires à partir d'un navigateur Internet standard. Le code que partagent ces applications est organisé comme un framework WebGD. Le comportement de l'interface cartographique d'une application WebGD est défini par des fichiers de configuration. Nous avons également un générateur d'application WebGD (WebGD-Gen) qui produit automatiquement ces fichiers de configuration à partir des métadonnées de la base de données, incluant celles des tables *geometry_columns* et *spatial_ref_sys*. WebGD-Gen permet également de générer des scripts Web qui interagissent avec l'interface cartographique et la base de données. Le framework WebGD et WebGD-Gen permettent donc de réduire significativement les temps de développement et les coûts de maintenance d'une application Internet SIG / base de données complexe.

Introduction

Internet est devenu le lieu de prédilection pour le partage de l'information. Une application de cartographie dynamique sur Internet permet aux utilisateurs, sans avoir recours à un logiciel de SIG bureautique, d'effectuer des requêtes spatiales et de produire des cartes sur un navigateur Internet standard.

Cependant, une application SIG sur Internet typique permet simplement de récupérer des cartes et des données relatives à ces cartes. Un serveur web fournit des informations au client mais le client ne peut pas renvoyer des informations au serveur (3). Ce flux d'informations unidirectionnel est un problème majeur des applications cartographiques sur Internet actuelles. De plus, créer une application Internet interactive avec une interface cartographique prend du temps. Les serveurs cartographiques propriétaires et les systèmes de gestion de base de données géographiques coûtent cher. Cette situation

freine l'utilisation d'applications SIG Internet destinées à rassembler des données, à l'analyse et à la prise de décision.

Nous avons développé une série d'outils qui permettent de réduire significativement le coût des développements (14; 16; 9; 15). Le code partagé par les applications SIG Internet interactives est organisé comme un framework Internet SIG / base de données (WebGD). La plupart des fonctions complexes qui produisent des fonctions SIG sur Internet sont inclus dans ce framework. Avec le framework WebGD, en utilisant uniquement des fichiers de configuration, on peut créer une interface cartographique permettant aux utilisateurs d'ajouter, interroger, et supprimer des entités géographiques ainsi que leurs données attributaires

Une fonctionnalité importante d'une application WebGD est qu'elle intègre complètement les données spatiales et leur données attributaires, ce qui permet d'effectuer des analyses spatiales sur les données (13). Ces fonctions sont à la disposition des utilisateurs sur des zones géographiques différentes, pour des questions de gestion économique et rapide des données.

Il y a quelques années, nous avons développé un générateur de formulaires web qui génère automatiquement des scripts pour des formulaires traditionnels à partir du schéma d'une base de données relationnelle (4). Cette fonctionnalité a été récemment étendue en générateur d'applications WebGD (WebGD-Gen). Ce nouveau générateur d'applications peut produire la plupart du code pour une application WebGD à partir du schéma relationnel d'une base de données et les informations concernant les *geometry columns*. Avec l'interface cartographique et les scripts web générés automatiquement, les entités géographiques (par exemple des points, des lignes et des polygones) peuvent être ajoutés, interrogés, et supprimés. Ainsi, si un schéma relationnel et des métadonnées sur les couches sont disponibles, une application non-personnalisée peut rapidement être mise en place grâce au framework WebGD et au générateur d'applications WebGD-Gen.

La génération automatique d'interfaces cartographiques et de scripts pour formulaires web rend possible le développement *incrémental* et *itératif* d'applications SIG Internet. Lorsqu'une couche géographique est ajoutée ou modifiée, il suffit de créer ou

de mettre à jour le fichier de configuration concernant cette couche puis de régénérer le formulaire web pour cette couche. Lorsqu'un schéma de base de données est modifié, il est possible de régénérer intégralement le set de scripts web en quelques minutes. Ainsi, même avec un jeu de couches géographiques et schéma de base de donnée partiel, il est possible de générer un prototype fonctionnel pour de premiers tests. De plus, la plupart des bugs peuvent être corrigés en modifiant simplement le code partagé dans le framework WebGD ou dans WebGD-Gen puis en régénérant les scripts web pour chaque application.

Les applications WebGD utilisent les composants *open-source* suivants. PostgreSQL, un système de gestion de base de données *relationnel-objet*, et PostGIS permettent de gérer les données spatiales. PostGIS est une extension de PostgreSQL pour les applications SIG (11). MapServer (12) génère les cartes qui sont visualisées par un navigateur web à partir des données spatiales fournies par PostGIS. Les pages web, dont celle qui affiche les cartes, sont générées par des scripts PHP côté serveur. Le module PHP MapScript interagit avec MapServer (6; 2). Lorsqu'une requête d'ajout ou de suppression d'une entité géographique est reçue par un script PHP, celui-ci accède directement à la base de données PostgreSQL en utilisant l'extension PostGIS. Ce genre d'applications tournent sur un PC sans aucun logiciel propriétaire.

Nous détaillons les fonctionnalités du framework WebGD dans la partie **Applications WebGD**. Le processus de génération des fichiers de configuration des couches géographiques est détaillé dans la partie **Framework WebGD** et celui de la génération des scripts web dans la partie 4. Le processus de génération automatique des fichiers de configuration des couches géographiques est expliqué dans la partie **Génération automatique des fichiers de configuration des couches géographiques**. La partie **Historique du Développement de WebGD** se consacre à un bref historique du développement du framework WebGD et de WebGD-Gen. La partie **Conclusions et travaux à venir** conclue cet article.

Applications WebGD

la figure 1 montre l'interface web d'une des applications WebGD, le "Natural Heritage Information System" de Caroline du Nord (NHIS). Cette application fournit une interface cartographique pour une copie de la base de données Biotics 4.0 maintenue par le "North Carolina Natural Heritage Program".

Biotics 4.0 est une application SIG bureautique basée sur la base de données développée par NatureServe. Les éléments clés de cette base sont les *occurrences d'éléments* (EOs) qui correspondent aux zones terrestres et/ou aquatiques dans lesquelles les espèces sont présentes ou ont été présentes (7). Les enregistrements d'EO sont composés à la fois de données *spatiales* et de données *tabulaires*, et la base de données contient environ 700 tables relationnelles citeFogelsong :2002. L'extension Biotics Mapper implémentée dans ArcView par NatureServe fournit une interface cartographique qui permet de représenter, créer, mettre à jour et supprimer les EOs et les données attributaires (8). Dans notre implémentation, nous pouvons réaliser toutes ces opérations à partir d'un simple navigateur Internet. De plus, des formulaires web, environ 3500 au total, sont générés pour toutes les tables de la base de données.

L'application NHIS permet des transferts *bi-directionnels* de données spatiales aussi bien que de données ordinaires. Les scientifiques ou autres personnes disposant des droits nécessaires peuvent *ajouter, interroger et supprimer* des entités géographiques telles que des polygones EO, des lignes, des points ainsi que des données attributaires. Des requêtes peuvent être effectuées en *sélectionnant des zones* sur la carte ou en utilisant un formulaire web traditionnel. De plus, des images *orthorectifiées* ou des photos aériennes dont la résolution est d'un mètre sont disponibles en tant que couche. Lorsque ces images sont combinées avec d'autres couches cartographiques telles que les autoroutes, les limites de comtés, le réseau hydrographique et les rues, des lieux peuvent être facilement localisés grâce à la superposition des couches (16).

Les opérations principales proposées par l'interface cartographique d'une application WebGD sont les suivantes :

1. Pour récupérer des informations concernant des entités géographiques d'une zone d'intérêt, l'utilisateur peut zoomer ou dézoomer sur cette zone en utilisant les outils de navigation cartographique. Si l'utilisateur zoome suffisamment, les images aériennes d'un mètre de résolution deviennent visibles. L'utilisateur peut également se déplacer vers une nouvelle zone en la sélectionnant dans la liste des **raccourcis**.
2. Pour obtenir des informations sur une entité géographique, l'utilisateur peut sélectionner la couche dans la légende et le bouton **Information** dans la barre d'outils puis cliquer sur les limites de cette entité.
3. La fonction **Insert** permet d'ajouter une entité

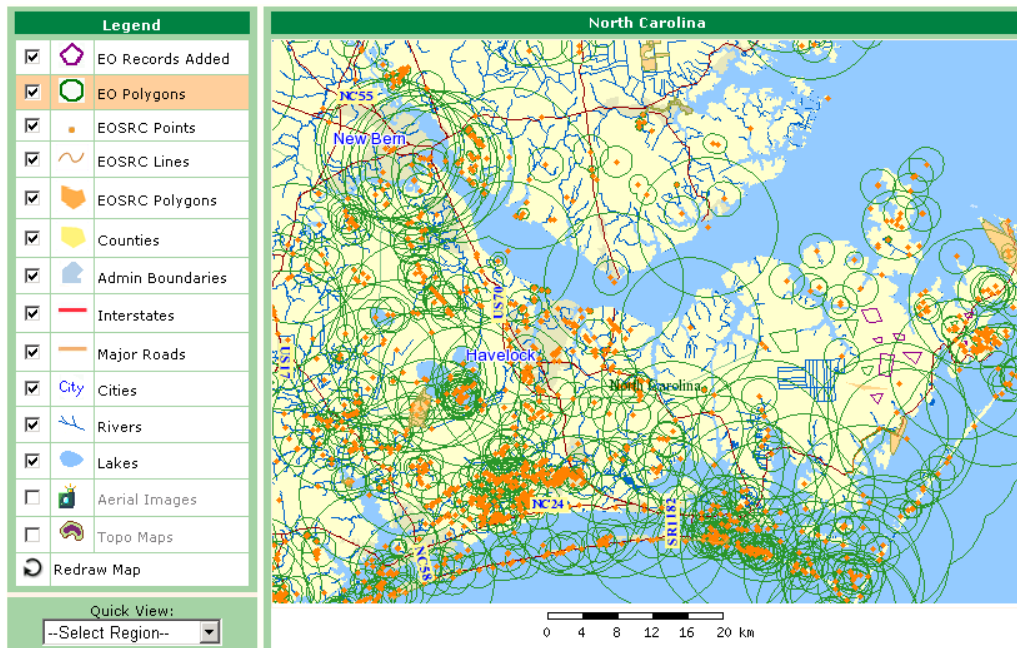


FIG. 1: Interface de l'application NHIS de Caroline du Nord

géographique par clic sur la carte. Le bouton **Done** doit être pressé une fois que tous les points ont été rentrés.

4. La fonction **Search by Area** permet aux utilisateurs de récupérer une liste d'entités comprises dans un *rectangle d'emprise* spécifié sur la carte et qui satisfont les critères de recherche. Ces entités sont mises en *surbrillance* sur la carte. De plus, l'utilisateur peut sélectionner des entités dans la liste en cochant les cases qui leur sont associées. Ensuite, la carte est actualisée et les entités sélectionnées sont mises en surbrillance.
5. L'interface d'administration des données peut être activée en cliquant sur **Database** dans la barre de menu sous la bannière. On peut cliquer sur un icône pour afficher une *arborescence* que l'on peut parcourir. L'arborescence pour les **taxons** est la principale. Comme certaines données sur les espèces menacées sont confidentielles, l'utilisateur doit s'authentifier avec un mot de passe pour accéder aux données de cette application.

Plusieurs applications WebGD créées avec le framework WebGD et WebGD-Gen sont accessibles sur cette page.³

³Exemples d'applications WebGD : http://yukon.een.orst.edu/index_webgd.html Vous pouvez y ajouter/interroger/mettre à jour/supprimer des données. Cependant, NE DISTRIBUEZ PAS ces données. Si un login est demandé, utilisez le nom d'utilisateur cs540 et le mot de passe CSxyz540. Il est possible que vous ayez à vous authentifier deux fois, une fois pour le serveur puis une fois pour une application.

La première application proposée est le "Natural Heritage Information System". Bien que cette application couvre tout les USA ou le monde, les données actuellement disponibles ne couvrent que la Caroline du Nord. La deuxième application fournit une interface cartographique pour une application qui suit les pratiques de conservation sur des parcelles de territoire. La troisième est une application cartographique sur Internet pour la collection de plantes germplasm maintenue à la "Western Regional Plant Introduction Station" (USDA-ARS). La quatrième application permet de récupérer les informations sur le sol à l'endroit sélectionné par clic sur l'interface cartographique.

Une caractéristique remarquable du framework WebGF actuel est la possibilité de *changer dynamiquement de système de référence spatiale*. Typiquement, chaque région géographique ou localité a des *projections recommandées* de manière à éviter les distorsions sur les cartes créées (1). Le framework permet de couvrir le monde entier avec des cartes différentes à différentes échelles, par exemple la carte du monde, celle d'un continent ou d'une région. L'interface cartographique sélectionne automatiquement la projection la mieux adaptée à la zone géographique

affichée. Par exemple, pour le monde on peut utiliser un *système de coordonnées géographiques*, pour les USA la *projection équivalente Albers* ou pour l'Oregon la *projection conique conforme Lambert*. Les analyses spatiales peuvent donc se faire en utilisant la projection la mieux adaptée pour une région particulière. Le *changement dynamique de référence spatiale*, le *fichier de configuration de la carte*, la *légende* et les *raccourcis* supportés par le framework WebGD permettent de couvrir n'importe quelle partie du monde ayant sa propre échelle et son propre système de référence spatiale, y compris les régions couvertes par des images aériennes d'un mètre de résolution. Il s'agit d'une fonctionnalité très importante d'autant plus qu'aujourd'hui le coût de stockage d'images aériennes sur les Etats-Unis entiers devient abordable (les 10 téraoctets nécessaires pour stocker des images aériennes sur tous les USA coûtent actuellement environ \$10,000). De plus, beaucoup d'Etats reversent leurs images aériennes dans le domaine public.

Framework WebGD

Le framework WebGD propose les fonctionnalités de base de l'interface cartographique d'une application WebGD telles que le zoom, le déplacement sur la carte et les opérations d'ajout/interrogation/mise à jour/suppression sur les entités géographiques.

Une opération sur la carte qui ne manipule pas la géométrie des entités, par exemple, le *zoom* ou le *déplacement sur la carte* est traitée de la manière suivante :

1. L'action de l'utilisateur sur la carte est transmis par le navigateur web au serveur web par une requête HTTP GET.
2. Le serveur web active le script PHP qui gère l'action de l'utilisateur.
3. Dans le script PHP, plusieurs méthodes PHP MapScript sont appelées pour préparer les nouveaux paramètres de la carte tels la projection cartographique et l'étendue de la carte ainsi que les noms des couches à afficher. La méthode de dessin de la carte dans PHP MapScript est ensuite appelée pour demander la création de l'image de la carte à MapServer.
4. MapServer demande les données relatives aux couches géographiques stockées dans la base de données PostgreSQL.
5. Les données pour les couches sont renvoyées.
6. L'image de la carte est renvoyée au script PHP.
7. La page HTML générée par le script PHP est renvoyée au serveur web.

8. Le serveur web transmet la page HTML au navigateur web en incluant l'image de la nouvelle carte .

Une opération cartographique qui accède ou manipule la géométrie d'entités, par exemple la *recherche par zone*, l'*ajout* ou *déplacer un point*, est traitée de la même manière que ci-dessus, excepté qu'entre les étapes 2 et 3, les opérations suivantes sont réalisées :

- a Le script PHP se connecte à la base de données PostgreSQL pour réaliser une opération spatiale via une requête SQL adaptée pour PostGIS.
- b Le résultat de l'opération spatiale est renvoyée au script PHP par requête SQL.

Le framework WebGD fait beaucoup appel à des *fichiers de configuration* pour la personnalisation d'une application. Les fichiers de configuration principaux sont les *fichiers de configuration des régions*, les *fichiers de configuration des couches géographiques* et un *fichier de configuration des raccourcis*.

Fichiers de Configuration de Région

Afin de limiter les distorsions, différentes zones géographiques peuvent être affichées en utilisant différents systèmes de référence spatiale. La *plus petite zone* qui contient l'étendue de la carte à générer détermine la *région actuelle*. La *référence spatiale* est ensuite automatiquement changée pour correspondre à celle de la nouvelle région. Enfin la légende des couches, la liste des commandes de navigation sur la carte et de manipulation des données ainsi que la liste des *raccourcis* sont reconfigurés pour cette nouvelle région. Toutes ces reconfigurations sont nécessaires car il se peut que différentes régions nécessitent différents jeux de couches géographiques, commandes et listes des raccourcis.

Pour chaque région, le *fichier de configuration de région* inclut les éléments suivants :

1. le système référence spatiale spécifique à la région,
2. le *nom de la région* qui est affiché sur l'interface cartographique,
3. *l'unité de mesure des distances*,
4. le *nom du fichier de configuration des couches géographiques* de la région,
5. le *nom du fichier de configuration des raccourcis*.

Par exemple le fichier de configuration pour le monde correspond à :

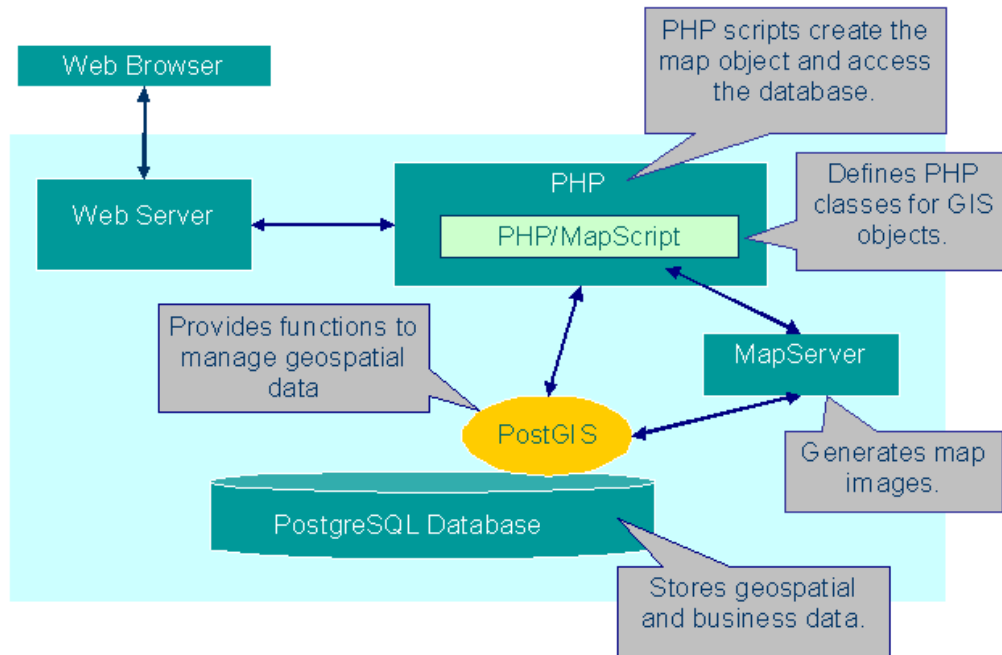


FIG. 2: Organisation d'une Application WebGD

```
$region = array(
    "gid" => 1,
    "name" => "world",
    "display_name" => "World",
    "srid" => 4326,
    "rank" => 1,
    "units" => "MS_DD",
    "mapfile" => "gmap75_world.map",
    "quickview" => "world_gview.php",
    "legend" => "world_maplayers.php",
    "proj4text" => "+proj=longlat +ellps=WGS84
    +datum=WGS84"
);
```

Le options suivantes peuvent être spécifiées dans un fichier de configuration :

- gid** – l'identifiant unique associé à chaque région, il s'agit de la clé primaire de chaque enregistrement représentant une région dans la table regions de la base de données.
- name** – le nom de la région pour le programme.
- display_name** – le nom de la région utilisé sur l'interface cartographique. Ce nom peut inclure des capitales, des espaces et autres signes de ponctuation.
- srid** – l'identifiant de la référence spatiale pour la projection cartographique associée à la région.
- rank** – le chiffre représentant la priorité, utilisé pour déterminer la région à afficher sur la carte. La région avec le chiffre le plus élevé est sélectionnée parmi les régions dont l'étendue est com-

- plètement comprise dans celle de la carte à afficher. Les régions avec les rangs les plus élevés recouvrent celles dont les rangs sont inférieurs.
- units** – l'unité de mesure associée à la région.
- mapfile** – le fichier de configuration de la carte à charger lorsque la région est sélectionnée.
- quickview** – le nom du fichier de configuration du raccourci pour la région.
- legend** – le nom du fichier de configuration des couches géographiques pour la région.
- proj4text** – la chaîne de caractères de définition de la projection utilisée pour la région.

Dès que la zone cartographique change, suite à une action de l'utilisateur sur l'interface cartographique, le fichier de configuration de la nouvelle région est chargé automatiquement. Ensuite l'interface cartographique est modifiée selon le fichier de configuration de cette nouvelle région.

Par exemple, le fichier de configuration pour l'Oregon correspond à :

```
$region = array(
    "gid" => 150137,
    "name" => "oregon",
    "display_name" => "Oregon",
    "srid" => 6010,
    "rank" => 150137,
    "units" => "MS_FEET",
    "mapfile" => "gmap75_oregon.map",
    "quickview" => "oregon_gview.php",
```

```

"legend" => "oregon_maplayers.php",
"proj4text" => "+proj=lcc +lat_1=43.0
+lat_2=45.5 +lat_0=41.75
+lon_0=-120.5
+x_0=400000.00000 +y_0=0.0"
);

```

Fichier de Configuration des Couches Géographiques

Le fichier de configuration des couches géographiques pour une région spécifie les couches à inclure dans la légende ainsi que leurs caractéristiques. Plusieurs régions peuvent partager un même fichier de configuration des couches géographiques.

Par exemple, le fichier de configuration des couches géographiques pour la région de l'Oregon est défini comme suit :

```

$layer_groups = array (
  'grp_eo_py' => array(
    'geom_type' => 'polygon',
    'table' => 'eo_py',
    'layer_selectable' => true,
    'gid_column' => 'gid',
    'geom_col' => 'the_geom',
    'legend_label' => 'EO Polygons',
    'search_script' =>
      'forms/eo/eo_py_eo_search.phtml',
    'select_script' =>
      'forms/eo/eo_py_eo_select.phtml',
    'edit_script' =>
      'forms/eo/eo_py_edit.phtml',
    'normal_layer' => 'eo_py',
    'searched_layer' => 'eo_py_searched',
    'checked_layer' => 'eo_py_checked',
    'selected_layer' => 'eo_py_selected',
    'img_src' => 'images/eo_poly.png',
    'img_width' => 26,
    'img_height' => 26,
    'onclick' => 'activate_layer("grp_eo_py")',
    'data_srid' => 32119
  ),
  'grp_eosrc_pt' => array(
    'geom_type' => 'point',
    'table' => 'eosrc_pt',
    'layer_selectable' => true,
    'gid_column' => 'gid',
    'geom_col' => 'the_geom',
    'legend_label' => 'EOSRC Points',
    'search_script' =>
      'forms/eo/eosrc_pt_search.phtml',
    'select_script' =>
      'forms/eo/eosrc_pt_select.phtml',
    'edit_script' =>
      'forms/eo/eosrc_pt_edit.phtml',
    'normal_layer' => 'eosrc_pt',
    'searched_layer' => 'eosrc_pt_searched',
    'checked_layer' => 'eosrc_pt_checked',
    'selected_layer' => 'eosrc_pt_selected',
    'img_width' => 5,
    'img_height' => 5,
    'onclick' => 'activate_layer("grp_eosrc_pt")',
    'data_srid' => 32119
  )
);

```

```

),
...

```

La légende de la carte sur la figure 3 a été produite avec l'exemple de fichier de configuration présenté ci-dessus.

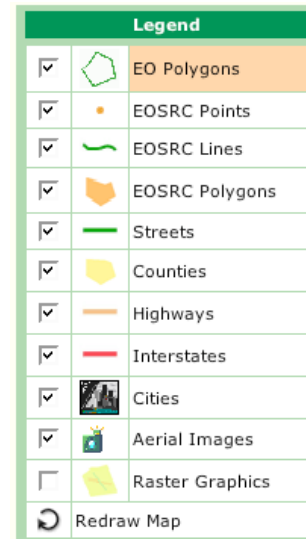


FIG. 3: Légende de la carte de l'application WedGD NHIS

Nous allons détailler chaque option utilisée dans ce fichier de configuration :

geom_type – le type de géométrie des entités contenues dans la couche. Il peut s'agir de polygon (polygone), multipolygon (multi-polygone), linestring (ligne), multilinestring (poly-ligne), point (point) ou multipoint (multi-point).

table – le nom de la table dans la base de données qui contient le champ géométrie de la couche.

layer_selectable – une valeur booléenne qui détermine si des opérations spatiales peuvent être effectuées sur la couche ou pas. Certaines couches, telles que les comtés et les routes principales de l'exemple ci-dessus, sont statiques et ne sont pas sélectionnables pour les opérations spatiales.

gid_column – le nom du champ qui contient les identifiants des entités géométriques de la couche.

geom_col – le nom du champ géométrie pour la géométrie des entités contenues dans la couche. Le nom par défaut est the_geom.

legend_label – le nom de la couche dans la légende.

search_script – le nom et la localisation du formulaire de recherche associé à la couche.

select_script – le nom et la localisation du formulaire de sélection associé à la couche.

edit_script – le nom et la localisation du formulaire d'édition associé à la couche.

normal_layer – le nom de la couche affichée lorsqu'elle n'est pas en surbrillance.

searched_layer, checked_layer, selected_layer – les noms des couches utilisées pour mettre en surbrillance les entités géométriques respectivement *cherchées, cochées* ou *sélectionnées*.

img_src, img_width, img_height – le nom du fichier, la largeur et la hauteur de l'icône dans la légende.

onclick – le nom du gestionnaire d'événements javascript activé lorsque l'utilisateur sélectionne la couche dans la légende.

data_srid – l'identifiant de la référence spatiale (*srid*) qui correspond à la projection cartographique utilisée pour les entités géométriques de la couche. Si le *srid* est différent de celle de la région, les entités géométriques de la couche sont reprojctées avant d'être affichées sur la carte.

Fichier de Configuration des Raccourcis

Le mécanisme de *raccourcis* permet à l'utilisateur de sélectionner une zone d'intérêt. C'est-à-dire que l'utilisateur peut rapidement passer à une nouvelle zone en la sélectionnant dans la liste des raccourcis.

Chaque entrée dans un *fichier de configuration des raccourcis* décrit le nom de la zone représentée par cette entrée, la projection cartographique utilisée sur l'étendue de la zone et l'étendue. Par exemple, le fichier de configuration du raccourci pour la région de l'Oregon présentée sur la figure 4 pourrait être comme suit :

```
$qview = array(
  array(
    'name' => 'World',
    'srid' => 4326,
    'extent' => '-180,-90,180,90'
  ),
  array(
    'name' => 'United States',
    'srid' => 4326,
    'extent' => '-125,13,-65,53'
  ),
  array(
    'name' => 'United States, East',
    'srid' => 4326,
    'extent' => '-102,22,-60,50'
  ),
  array(
    'name' => 'United States, West',
    'srid' => 4326,
    'extent' => '-135,30,-105,50'
  ),
)
```

```
array(
  'name' => 'Whole Oregon',
  'srid' => 6010,
  'extent' => '46461.662375,-43912.968464,
    2487069.754184,1785176.331408',
),
...
```

Voici la liste des raccourcis des régions telle qu'elle apparaît sur l'interface cartographique correspondant au fichier de configuration présenté ci-dessus.

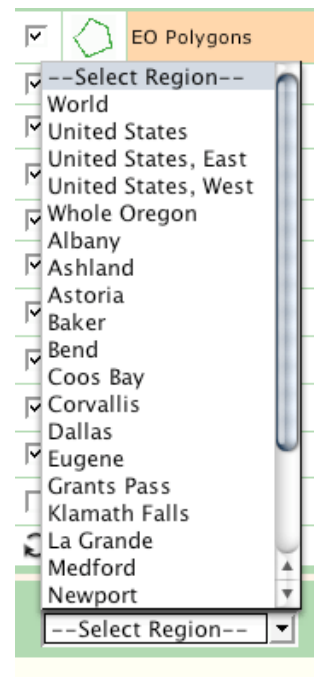


FIG. 4: Choix des raccourcis pour l'application WebGD NHIS

Pour chaque entrée dans le fichier de configuration des raccourcis, les options suivantes peuvent être spécifiées :

name – le nom de la zone.

srid – l'identifiant de la référence spatiale pour l'étendue détaillée ci-dessous.

extent – les coordonnées *xmin, ymin, xmax* et *ymax* décrivant la position des coins inférieur gauche et supérieur droit de la zone géographique.

Génération automatique de scripts de formulaires web

Plusieurs outils ont été développés pour enrichir le framework WebGD et simplifier le développement d'une application. Le générateur de site In-

Internet WebGD (WebGD-Gen) peut créer une application WebGD entière, incluant une interface cartographique web. WebGD-Gen génère *automatiquement* un jeu de scripts de formulaires web cohérents à partir des *fichiers de configuration* qui sont également générés automatiquement à partir du schéma relationnel d'une base de données. La génération de formulaires étant automatique, les coûts de développement d'une application en sont grandement réduits. Pour une base de données telle que Biotics qui contient environ 700 tables, programmer l'ensemble des 3 500⁴ formulaires nécessaires manuellement serait extrêmement coûteux voir infaisable.

WebGD-Gen correspond à une collection de *templates*. Chaque template, combiné avec le *fichier de configuration* correspondant, génère un des six types de scripts web : les scripts *recherche*, *sélection*, *édition*, *informations*, *action* et *arborescence*. Les templates et les fichiers de configuration sont écrits en PHP. Les scripts web qu'ils génèrent sont également en PHP. Ces scripts générés sont exécutés par un serveur web via un interpréteur PHP. Chaque script, excepté le script action, crée un formulaire web qui est affiché par l'ordinateur client via un navigateur Internet. La figure 5 illustre les interactions entre les scripts web et les formulaires.

De plus, si les lignes suivantes données en exemple sont ajoutées à un fichier de configuration, WebGD-Gen permet de générer automatiquement les requêtes pour l'ajout, la recherche et la suppression d'*entités géographiques* :

```
// type of geographical features
$web_gd = 'MULTIPOLYGON';
// layer group in legend
$layer_name = 'grp_eo_py';
// geometry column containing shapes
$geometry_column = 'the_geom';
//geographical feature IDs
$gid_column = 'gid';
// epsg spatial reference
$db_table_srid = 32119;
```

Les formulaires générés pour les entités géographiques ont les fonctionnalités supplémentaires suivantes, comparé aux formulaires des tables ordinaires d'une base de données :

1. Un formulaire de recherche peut être activé à partir de l'interface cartographique. Dans ce cas, l'étendue du rectangle de recherche spécifié sur la carte est passé en tant que paramètre additionnel de recherche.
2. Un formulaire de sélection comprend du code Javascript supplémentaire pour mettre en sur-

brillance les entités géographiques récupérées ou sélectionnées par l'utilisateur.

3. Un formulaire d'édition peut insérer un enregistrement pour une entité géographique, après transformation des coordonnées depuis le système de référence spatiale de l'interface cartographique actuelle vers celui utilisé par le champ géométrie de l'enregistrement.

Les formulaires reliés sont automatiquement associés les uns aux autres. Par exemple, la figure 6 montre le formulaire d'*édition* d'une table Etudiant. Depuis ce formulaire d'édition, l'utilisateur peut ouvrir les formulaires du département et des cours associés à l'étudiant. Les informations nécessaires pour créer ces liens sont extraites des *relations clé primaire-clé étrangère* entre les tables dans la base de données.

Afin de générer le formulaire d'édition d'un Etudiant de la figure 6, les paramètres suivants doivent être rentrés dans le fichier de configuration.

```
$edit_fields=array(
  array("column"=>"student_id",
    "label"=>"Student Id", "type"=>"numeric",
    "maxlen"=>"40", "size"=>"40"),
  array("column"=>"first_name",
    "label"=>"First Name", "type"=>"text",
    "maxlen"=>"40", "size"=>"40"),
  array("column"=>"age", "label"=>"Age",
    "type"=>"text", "maxlen"=>"40", "size"=>"40"),
  array("column"=>"gender", "label"=>"Gender",
    "type"=>"text", "maxlen"=>"40", "size"=>"40"),
  array("column"=>"department_id",
    "label"=>"Department ID", "type"=>"to_one",
    "linked_table"=>"departments", "maxlen"=>"40",
    "size"=>"40"),
  array("label"=>"Courses Taken", "type"=>"to_many",
    "linked_table"=>"courses", "maxlen"=>"40",
    "size"=>"40"),
  array("column"=>"other_info",
    "label"=>"Other Information",
    "type"=>"textarea", "rows"=>"4",
    "cols"=>"32",
  );
```

Chaque élément du tableau \$edit_fields représente un champ dans un formulaire d'édition avec les options column (champ), label (étiquette), table (table), maxlen (longueur maximale), size (taille), et type (type). L'option type peut être numeric, text, time, date, email, phone, textarea, to_one ou to_many.

textarea : il s'agit d'une chaîne de caractère affichée dans une zone de texte.

to_one : Un champ de type to_one permet à l'utilisateur de visualiser ou de modifier un enregistrement lié à l'actuel via une relation du type

⁴700 x 50

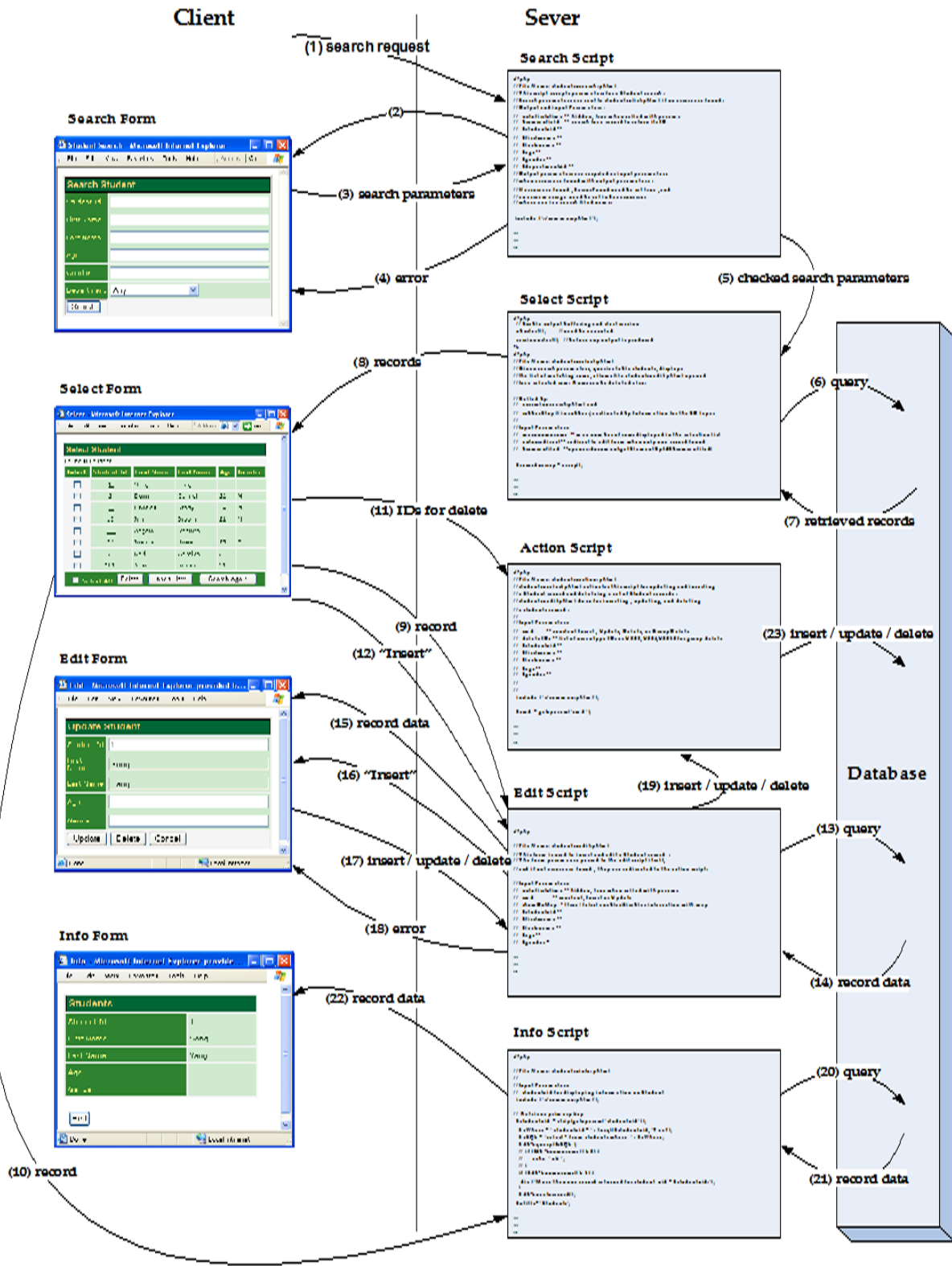


FIG. 5: Interactions entre les scripts web et les formulaires

Department Select Form

Select	Department ID	Department Name	Description
<input type="checkbox"/>	1	Computer Science	Department of Computer Science
<input type="checkbox"/>	2	Mathematics	Department of Mathematics

Department Search Form

Student Edit Form

Update Student

Student ID: 123

First Name: Joanna

Last Name: Howell

Age: 25

Gender: F

Department ID: 1

Courses taken: EHQW

Other Information:

Department Edit Form

Update Department

Department ID: 1

Department Name: Computer Science

Description: Department of Computer Science

Course Select Form

Select	Student ID	Course ID	Course Name	Grade
<input type="checkbox"/>	123	1	Databases	A
<input type="checkbox"/>	123	2	Programming	C
<input type="checkbox"/>	123	3	Operating System	B
<input type="checkbox"/>	123	4	Software Engineering	C

FIG. 6: Formulaire d'édition d'un Etudiant

un à un ou *plusieurs à un*. Dans notre exemple, le champ `department_id` de la table `students` est spécifié comme étant de type `to_one`. Lorsque l'on clique sur le bouton `View`, le formulaire d'édition montrant l'enregistrement correspondant au département de l'étudiant est affiché. De plus, un utilisateur peut chercher et sélectionner un nouveau département à lier. L'option `linked_table` désigne le nom de la table qui stocke l'enregistrement associé. L'option `child_column` indique le champ à utiliser pour lier dans la table associée. Si cette option est omise le nom du champ utilisé pour le lien est identique à celui du champ clé étrangère de l'enregistrement actuel.

to_many : L'objectif de ce type est de lister les enregistrements liés à l'actuel via une relation du type *un à plusieurs* ou *plusieurs à plusieurs*. Dans notre exemple, le champ étiqueté `Cours` suivi est de type `to_many`. Lorsque l'on clique sur le bouton `Show`, tous les cours pris par l'étudiant sont affichés dans le formulaire de sélection des cours. Ce type nécessite la présence des options `linked_table`, `parent_column` et `child_column`. Les options `linked_table` et `child_column` peuvent être omises lorsque le nom du champ de la clé étrangère de la table associée est identique à celui du champ de la clé primaire de l'enregistrement actuel. L'option `parent_column` indique le champ de la clé étrangère dans la table actuelle. Si cette option est omise, le champ de la clé étrangère correspond au champ de la clé primaire de la table actuelle.

Génération automatique des fichiers de configuration des couches géographiques

On peut créer un fichier de configuration des couches géographiques de manière semi-automatique en utilisant les informations de la table `geometry_columns` qui stocke les métadonnées liées au SIG et le nom de la couche dans le fichier de configuration de la carte.

1. Le *méta-fichier de configuration des couches géographiques* utilisé par défaut et qui liste tous les groupes de couches, `xxx_maplayers.mconfig`, peut être créé avec **map_mconfgen**. Les noms des groupes de couches sont générés à partir de des noms des tables stockés dans la table

`geometry_columns` et à partir des noms des groupes de couches définis dans le fichier de configuration de la carte.

2. On peut modifier le fichier de méta-configuration des couches utilisé par défaut et dont le format est identique à un fichier de configuration des couches, en sélectionnant et en changeant l'ordre des groupes de couches. On peut également fournir des valeurs personnalisées pour les paramètres de chaque groupe de couches. Le fichier de méta-configuration a uniquement besoin de garder les définitions des paramètres personnalisés.
3. Enfin, le fichier de configuration des couches `xxx_maplayers.config` est généré par **map_mconfgen**. Les informations nécessaires à cette étape sont récupérées du méta-fichier de configuration, de la table `geometry_columns` et du fichier de configuration de la carte. Les valeurs par défaut des paramètres, déterminées par la table `geometry_columns` et du fichier de configuration de la carte, sont écrasées si des valeurs personnalisées sont présentes dans le fichier de méta-configuration.

Le fichier de méta-configuration des couches permet de définir n'importe quel paramètre accepté dans le fichier de configuration des couches. Cependant, le fichier que **map_mconfgen** génère par défaut contient uniquement, pour chaque groupe de couches, les définitions des paramètres suivants :

layer_selectable – Une option booléenne qui détermine si la couche peut être sélectionnée pour des opérations spatiales ou pas.

legend_label – Le nom de la couche à afficher sur la légende de la carte.

Voici un exemple de fichier de méta-configuration généré par défaut :

```
$layer_groups = array(
  'grp_eo_py' => array(
    'layer_selectable' => true,
    'legend_label' => 'Eo Py',
  ),
  'grp_eosrc' => array(
    'layer_selectable' => true,
    'legend_label' => 'Eosrc Pt',
  ),
  ...
)
```

L'utilisateur peut modifier les valeurs de ces définitions et en ajouter d'autres. Par exemple le fichier de méta-configuration peut être modifié de la manière suivante :

```
$layer_groups = array(
```

```

'grp_eo_py' => array(
  'layer_selectable' => true,
  'legend_label' => 'EO Polygons',
  'search_script' => 'forms/eo/eo_py_eo_search.phtml',
  'select_script' => 'forms/eo/eo_py_eo_select.phtml',
  'edit_script' => 'forms/eo/eo_py_edit.phtml',
  'img_src' => 'images/eo_poly.png',
),
'grp_eosrc_pt' => array(
  'layer_selectable' => true,
  'legend_label' => 'EOSRC Points',
  'search_script' => 'forms/eo/eosrc_pt_search.phtml',
  'select_script' => 'forms/eo/eosrc_pt_select.phtml',
  'edit_script' => 'forms/eo/eosrc_pt_edit.phtml',
  'img_src' => 'images/eosrc_pt.png',
),
...

```

Les modifications suivantes ont été effectuées :

1. legend_label a été changé pour le nom souhaité de la couche dans la légende de la carte.
2. Les noms et la localisation des formulaires de recherche, de sélection et d'édition sont spécifiés respectivement dans search_script, select_script et edit_script.
3. Le nom du fichier source de l'icône dans la légende de la carte a été spécifié dans img_src.

L'ordre des groupes de couches peut également être changé pour correspondre à celui souhaité dans la légende de la carte.

Ensuite le fichier de configuration des couches montré précédemment peut être généré par `map_confgen`.

Historique du Développement de WebGD

Le framework WebGD et WebGD-Gen ont été développés de manière *incrémentielle* et *itérative* au cours des quatre dernières années. En 2000, nous avons tout d'abord implémenté une application qui permettait d'ajouter des entités de type ponctuel sur une carte en utilisant ASP avec ArcIMS et ArcSDE. En 2001, nous avons re-développé cette application avec ASP.NET, vu qu'ASP.NET propose des contrôles web qui constituent de meilleures briques pour la construction de pages web. Sur la base de cette application, la première version du framework WebGD a vu le jour en 2002, afin de supporter de multiples applications.⁽¹⁵⁾

Début 2003, nous avons re-développé une application appelée Motels Oregon avec MapServer, PostGIS et PostgreSQL⁽⁹⁾. Cette version Linux était non seulement plus fiable et rapide que la précédente

mais également basée sur des logiciels libres. En développant l'application MapServer suivante, un système de gestion de la ressource en germplasm (GEM-GIS), la première version du framework WebGD pour MapServer a été créée. Ce framework a ensuite été amélioré pour qu'il intègre la gestion des entités de type polygone aussi bien que celles de type ponctuel.

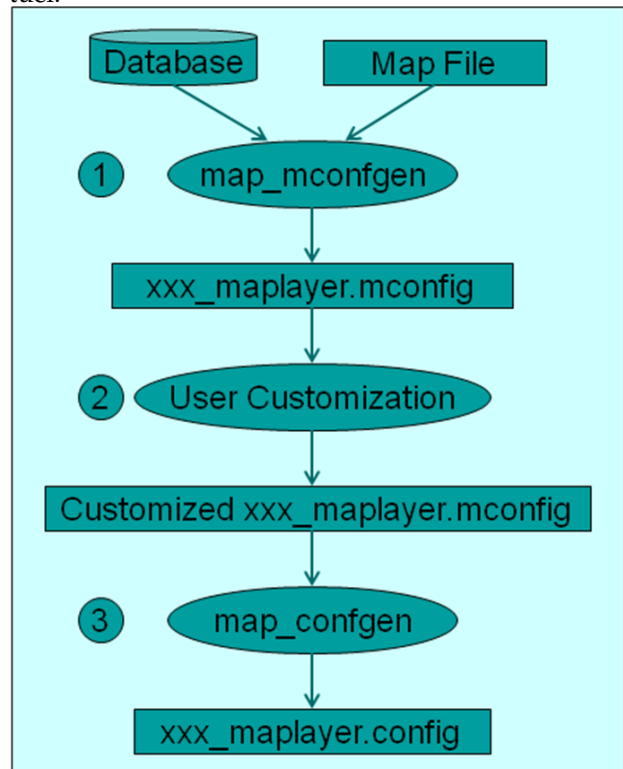


FIG. 7: Processus de création d'un fichier de configuration des couches

Deux améliorations majeures ont été faites en 2004 sur le framework WebGD : le *changement dynamique* de référence spatiale pour différentes régions et la *génération automatique* de formulaires web utilisés pour les ajouts, les interrogations et les suppressions d'entités géographiques. Comparé à une application qui affiche simplement des entités géographiques telles que des points sur une carte, le framework actuel est environ 20 fois plus complexe en terme de temps passé à développer les fonctionnalités nécessaires.

Conclusions et travaux à venir

Nous avons développé le framework WebGD et le générateur d'applications WebGD-Gen pour développer rapidement des applications SIG/base de

données Internet.

1. Les entités géographiques, telles que les habitats de plantes ou d'animaux, les zones de chantiers routiers ou des tronçons hydrographiques, peuvent être insérées, interrogées et supprimées grâce à l'interface cartographique et les formulaires web qui s'affichent sur un navigateur web standard.
2. Une application peut être créée sans programmer une seule ligne de code. L'interface cartographique et les scripts web pour l'accès aux données peuvent être générés automatiquement à partir de fichiers de configuration et ces fichiers de configuration peuvent eux-même être générés à partir du schéma de base de données et les métadonnées sur les données SIG, par exemple, les informations stockées dans la table `geometry_columns`. La génération automatique d'une application SIG Internet ne réduit pas seulement les coûts de développement de manière significative mais facilite également le développement incrémentiel et itératif de l'application.
3. Le changement dynamique de système de référence spatial permet à une application de couvrir différentes régions ayant des fichiers de configuration de la carte, des projections, des légendes de carte et des listes de raccourcis différents. Il s'agit d'une fonctionnalité importante pour une application qui couvrirait par exemple l'ensemble des Etats-Unis ou du monde.
4. Nous avons créé le framework WebGD en utilisant uniquement des logiciels libres et open source. Les outils logiciels que nous utilisons, tels que MapServer de l'Université du Minnesota, le SGBD PostgreSQL, PostGIS, Apache et PHP sont tous disponibles gratuitement. Les données SIG utilisées provenant de l'USGS, de TIGER/LINE et de Digital Chart of the World (DCW) font également partie du domaine public. Le framework est disponible à tous en utilisation libre.
5. Les coûts pour faire tourner nos applications sont extrêmement bas. Nous pouvons mettre des copies de grosses bases de données telles que Biotics, les données sur les sols de SSURGO2 et une partie du Système d'Information National sur la Ressource en Germplasm sur un PC à \$800.

La génération automatique du code d'une application WebGD permettra d'économiser beaucoup d'efforts dans le développement de systèmes spatiaux

d'aide à la décision. Bien que quelques personnalisations sont nécessaires, le temps à y accorder peut être diminué à des semaines ou des mois comparé aux années nécessaires pour développer un *système spatial d'aide à la décision* depuis le début.

Le framework WebGD et WebGD-Gen sont actuellement disponibles sur demande. Nous cherchons des collaborateurs pour le rendre publique en tant que logiciel libre et open source. Nous sommes également en train de re-développer l'interface cartographique en utilisant OpenLayers afin de supporter les déplacements fluides sur la carte.

Bibliographie

- [1] P. H. Dana Map Projection Overview http://www.colorado.edu/geography/gcraft/notes/mapproj/mapproj_f.html
- [2] DM Solutions Group Inc. PHP MapScript <http://www.maptools.org>
- [3] R. Kingston (1998) Web-based GIS for public participation decision making. In *Procs of NCGIA PPGIS Meeting*, Santa Barbara, California. Retrieved Map 2003 from <http://www.ncgia.ucsb.edu/varenius/ppgis/papers/kingston/kingston.html>
- [4] Eum D. and Minoura T. (June 2003) Web-based database application generator. *IEICE Transactions on Information and Systems*, Vol. E86-D, No. 6.
- [5] Fogelson, C. (2002) Biotics 4.0 data model version 1.0. Retrieved January 5, 2004, from <http://whiteoak.natureserve.org/hdms/HDMS-DataModel.shtml>
- [6] McKenna, Jeff MapServer PHP/MapScript Class Reference - Versions 3.6, 4.0 & 4.2 DM Solutions Group Inc.
- [7] NatureServe (February 2002) Element Occurrence Data Standard. Retrieved January 4, 2004, from <http://whiteoak.natureserve.org/eodraft/all.pdf>
- [8] NatureServe (December 2003) Biotics 4.0 Getting Started Guide. Retrieved January 5, 2004, from <http://whiteoak.natureserve.org/eodraft/all.pdf>

- naturereserve.org/hdms/biotics-learn-more.shtml (now obsolete).
- [9] Sano J., Wanalertlak N., Maki A., and Minoura T. (July 2003) Benefits of web-based GIS/database applications. In *Proc. of 2nd Annual Public Participation GIS Conference* Portland, Oregon.
- [10] USDA-ARS Western Regional Plant Introduction Station, USDA - Agricultural Research Service, Pullman, Washington <http://www.ars-grin.gov/ars/PacWest/Pullman/>
- [11] Ramsey, Paul PostGIS Manual Refractions Research Inc.
- [12] University of Minnesota (2003) MapServer <http://mapserver.gis.umn.edu>
- [13] Sharma, A. (December 2003) Web-based analysis module for a germplasm collection. Master of Science report, School of Electrical Engineering and Computer Science, Oregon State University
- [14] Wangmutitakul P, Li L, and Minoura T. (March 2003) User Participatory Web-Based GIS/Database Application. *Proc. of Geotec Event Conference*
- [15] Wangmutitakul Paphun et al. (2004) Framework for Web-based GIS/database Applications *Journal of Object Technology* 3, 209-225
- [16] Wuttiwat T., Minoura T. and Steiner J. (May 2003) Using Digital Orthographic Aerial Images as User Interfaces *Proc. of ASPRS Annual Conference*, Anchorage, Alaska

Toshimi Minoura, Nirut Chalainanont, Junya Sano
Oregon State University
<http://enr.oregonstate.edu/~minoura/research/creeda/>
minoura AT eecs.orst.edu

db4o2D - Extension de Base de données Orientée Objet ...

... pour les types géospatiaux 2D

Stefan Keller, traduit par Guillaume Sueur

db4o signifie "database for objects", c'est-à-dire Base de données Orientée Objet. C'est un système de gestion de base de données orienté objet (OODBMS ou SGDBOO en français) écrit en Java et en .NET et de ce fait destiné à ces deux plateformes. Ce logiciel a été initialement publié en 2001 par db4Objects, Inc., et depuis lors s'est taillé une part majeure du marché de ce qu'on peut appeler les bases de données objet de deuxième génération. Il est disponible sous deux licences ("dual licensing model"), une licence OpenSource de type GPL pour l'utilisation personnelle et non-commerciale mais aussi une licence commerciale.

Dans cet article, nous présenterons tout d'abord db4o. Dans le chapitre *OODBMS et db4o*, nous en présenterons les avantages, limites et différences d'un point de vue conceptuel mais aussi selon une approche de développeur. Puis nous ferons une présentation de la thèse d'un étudiant nommée db4o2D. Enfin nous concluerons sur les perspectives du projet db4o2D.

Premiers pas avec db4o

Commençons avec un peu de code source afin d'avoir une idée de la facilité avec laquelle on peut créer des objets persistants. Notre premier exemple met en oeuvre les quatre célèbres opérations de base en matière de bases de données : Créer, Lire, Mettre à jour, Supprimer (CRUD en anglais, pour Create, Read, Update and Delete). Cet exemple se base sur un objet *Personne* doté des attributs nom, prénom et date de naissance.

```
// 1: Initialisation d'un objet conteneur
ObjectContainer db= null;
try {
    // 2: Ouverture de la base db4o en mode integre
    db= Db4o.openFile("addressbook.yap");

    // 3: Creation et stockage de quelques 'Kellers'
    db.set(new Person("Brian", "Keller", 1960));
    db.set(new Person("Clara", "Keller"));
    db.set(new Person("Test"));
}
```

```
db.commit();

// 4: Recherche et lecture de tous les Keller
// Avec une "Requete par l'exemple"
ObjectSet result= db.get( \
    new Person(null, "Keller", 0));
while (result.hasNext())
    System.out.println( \
        (Person) result.next());

// 5: Mise a jour de l'age de Clara
result= db.get(new Person("Clara", "Keller"));
Person found= (Person) result.next();
found.setYearOfBirth(1970);
db.set(found);

// 6: Suppression de la donnee de Test
result= db.get(new Person("Test"));
while (result.hasNext())
    db.delete(result.next());

// 7: Enregistrement des modifications
db.commit();
} finally {
    if (db != null)
        db.close();
}
```

Fig. 1. Morceau de code montrant les opérations CRUD sur des objets personnes stockés dans un carnet d'adresses.

Dans cet exemple, nous parcourons les étapes suivantes qui illustrent quelques opérations CRUD :

1. Cette instruction initialise un conteneur dans lequel db4o manipule l'ensemble des objets persistants de manière transactionnelle.
2. La deuxième instruction ouvre un fichier de base de données db4o. Nous proposons l'utilisation de l'extension .yap mais ce n'est pas standardisé. Il y a une explication officielle qui indique qu'il s'agit de l'abréviation de 'yet another protocol' ('encore un nouveau protocole' NdT). De manière non officielle, cette extension se réfère à Yap, une petite île de Micronésie. Comme indiqué, nous choisissons d'utiliser db4o en mode intégré, mais il existe aussi un mode client-serveur.
3. Avec la méthode `set()` d'un `ObjectContainer` trois personnes nouvellement créées sont stockées. Avec le `commit()` tous les objets créés,

modifiés ou supprimés sont alors synchronisés avec le fichier de base de données.

4. Dans cette étape nous parcourons tous les objets de la base de données dont le nom est égal à 'Keller', et bouclons ensuite sur le résultat. Dans cet exemple, la 'requête par l'exemple' est utilisée parmi les trois langages d'interrogation disponibles dans db4o.
5. Lors de la création de "Clara Keller" à l'étape 3 nous avons oublié la date de naissance lors de l'appel du constructeur (car il est poli de ne pas révéler à l'avance l'âge d'une dame). Maintenant nous désirons vraiment indiquer cette valeur, et donc nous devons trouver les objets tels que 'Clara Keller' et mettre à jour le premier que renverra la requête. Nous utilisons la méthode `set()` à nouveau et présumons qu'il existe une méthode `setYearOfBirth()` dans la définition de la classe `Personne`.
6. A des fins de démonstration, nous avons préalablement inséré des données de test qui vont maintenant être supprimées. Cette fois, tous les objets récupérés par la requête seront détruits par la méthode `delete()` de l'objet `ObjectContainer`.
7. Enfin nous terminons avec la méthode `commit()` et fermons la base de données.

Les seules définitions absentes de ce bout de code sont les instructions d'importation ainsi que celles de la classe `Personne` qui consiste en trois constructeurs, la méthode `setYearOfBirth()` et de manière optionnelle, une méthode `toString()` pour permettre d'écrire l'objet `Personne` proprement.

Cela suffit à démontrer la simplicité d'une application utilisant db4o. Maintenant nous allons présenter quelques exemples d'utilisation de SGBDOO, quelques fonctionnalités supplémentaires de db4o avant une présentation plus large de l'extension qui incorpore les types géométriques dans db4o.

SGBDOO et db4o

Exemples d'utilisation et fonctionnalités

Quand il n'y a qu'une seule application à la fois tournant sur une base de donnée et si c'est une application dédiée à la mobilité comme on en trouve dans les LBS (Location Based Systems, ou systèmes basés sur la localisation) ou dans les logiciels embarqués, cela peut valoir la peine d'essayer un SGBDOO de deuxième génération. Du fait de leur facilité à gérer les relations entre objets, les SGBDOO sont intrin-

sèquement bien adaptés lorsque des modèles objet complexes, des structures objet à plat ou de grandes arborescences objet sont de mise – ce qui est de fait souvent le cas dans les Systèmes d'Information Géographique (SIG) et les LBS.

Voici quelques-uns des fonctionnalités techniques de db4o :

- Mode intégré et mode client-serveur.
- Pas d'administration serveur lors de l'exécution. Les propriétés de la base de données sont contrôlées hors de l'application hôte.
- Besoin de peu d'espace disque pour les bibliothèques du programme, de même qu'en mémoire lors de l'exécution.
- Facilité d'utilisation : db4o utilise des interfaces de programmation d'application (API) réflexives à partir de Java et .NET. Il n'y a donc aucune annotation supplémentaire, aucun pré ou post-traitement (ingénierie de byte code), aucune sous-classe ou implémentation d'interface à effectuer.
- Méthodes de contrôle du lazy loading (NdT : technique qui consiste à ne charger que le strict nécessaire pour l'exécution du programme) des relations des objets incorporés.
- Outils de réplication en modules complémentaires.

Comme on pourrait s'y attendre de la part d'une base de données, db4o implémente des propriétés ACID (Atomicité, Consistance, Isolation et Durabilité) qui garantissent des transactions sûres : une transaction démarre quand on ouvre ou interroge la base, et se termine avec les méthodes `commit()` ou `rollback()`. Trois approches sont implémentées pour les requêtes : la requête par l'Exemple, les requêtes SODA (Simple Object Database Access, ou Objet Simple d'accès à la base de données) et les 'Requêtes natives'. La première a été mise en pratique plus haut. Les requêtes SODA sont plutôt comparables à celles utilisées dans les environnements ORM (Object Relational Mapping, ou correspondance Objet-Relationnel). Une partie de la théorie mise en oeuvre derrière les Requêtes natives provient d'un projet de Microsoft (LINQ 2007).

Avantages

Parler d'environnement ORM nous amène à comparer les SGBDOO avec cette approche. db4o est très rapide si on en croit les bancs d'essai (Polepos 2007). Un argument conceptuel est qu'il n'y a pas d'erreur de correspondance objet/relationnel : aucun ajustement des types ou tailles des données (à moins de

le demander), aucune création d'un schéma relationnel séparé, pas de langage SQL et pas de requête textuelle SQL à gérer.

Il est donc important de mentionner que db4o propose une intéressante prise en charge des techniques de développement agile : c'est parce que les requêtes sont écrites dans le langage de l'application (Java, .NET) et sont donc sûres pour ce qui est du typage des données. Il y a aussi des fonctionnalités sympathiques d'évolution des schémas, et pas de SQL. Les ingénieurs logiciels ont la vie plus facile qu'avant car ils évoluent dans un monde objet à la différence des professionnels des bases de données.

Limites

db4o n'est probablement pas adapté pour l'utilisation dans de grands entrepôts de données ou le data mining. Il est particulièrement non recommandé quand plusieurs applications accèdent à la base de données avec de nombreuses vues. Le fait que différents types de langages soient disponibles pour les requêtes montre qu'aucun langage standard et mature n'est disponible en comparaison au prédominant SQL du modèle relationnel. Les contraintes, telles que l'intégrité référentielle ne sont pas (encore) intégrées dans ces langages, sauf dans les Requêtes Natives qui implémentent simplement des fonctions callback. Pour finir, le manque de standardisation actuel a été reconnu. Des actions du Groupe de Management Objet (OMG) sont en cours de manière à avancer vers une nouvelle version de standard ODMG version 4.

Projet db4o2D

PlaceLab (Intel 2006) — un projet Intel — propose des applications mobiles ou de bureau concernant le positionnement WLAN (Wifi). La base de données intégrée à ce logiciel est une base relationnelle open source embarquée. Un composant sensible ici est la gestion des points d'accès avec leur position respective. Cela nous sert de démonstration de base de données objet qui remplace la base relationnelle existante et stocke les géométries en tant qu'objets de première importance.

Dans un projet de thèse (db4o2D 2007), il a été décidé d'utiliser db4o et d'adapter la très utilisée Java Topology Suite (JTS 2007). La JTS respecte le standard 'Simple Features (NdT : Objets simples)' (OGC 2006) qui définit quatre types d'attributs géométriques 2d (2.5D) : le point, la ligne, le polygones

ainsi que des manipulations sûres sur ceux-ci.

Ainsi, dans la portion de code suivante (c.f. Fig 2.), il est montré comment les points d'accès sans fil sont créés (2), stockés (3) et relus (4). Un point contient une paire de valeur de coordonnées et comporte un système de références spatial et des unités de mesure par défaut.

```
// 1: Ouverture de la base de donnees db4o (mode integre)
db= Db4o.openFile("poidb.yap");

// 2: Preparation de deux points JTS
GeometryFactory factory= new GeometryFactory();
double latitude1= 47.225571, longitude1= 8.822271;
Point pt1= factory.createPoint(new Coordinate(
    longitude1, latitude1));
double latitude2= 47.225582, longitude2= 8.822282;
Point pt2= factory.createPoint(new Coordinate(
    longitude2, latitude2));

// 3: Creation et stockage des points d'accès
db.set(new AccessPoint( \
    1001, "802.11g", "wep", 7, pt1));
db.set(new AccessPoint( \
    1002, "802.11b", "open", 11, pt2));
db.commit();

// 4: Parcours de tous les points d'accès
ObjectSet result= db.get(new AccessPoint());
while (result.hasNext()) {
    System.out.println( \
        (AccessPoint) result.next());
}
```

Fig. 2. Portion de code avec des points d'accès sans fil qui contiennent une géométrie ponctuelle.

Conclusion

Le projet db4o2D deviendra un module complémentaire à db4o et est toujours en chantier. Bien que la couche existante d'accès à la base de données était correctement séparée, il est devenu évident que la maintenance du code était rendue plus courte, plus rapide et meilleure du fait de l'utilisation d'un pur SGBDOO.

Les travaux à venir incluent un index spatial et des tests de charge pour de grandes bases de données, ainsi qu'un mode client-serveur multitâche. Les requêtes SODA complexes sont un autre problème. Mais il faut noter que ce n'est pas un obstacle actuellement car les Requêtes Natives peuvent être utilisées d'ici là.

Dans le but de propager les technologies de base de données objet, un groupement à but non lucratif,

ODBMS.ORG (2007) a été créé. db4o est un des projets phares dans le domaine des bases de données objet et il tente de résoudre les problèmes bien connus dans le monde des logiciels embarqués, du développement logiciel, des LBS et des SIG. Sachant que ces problèmes sont des préoccupations depuis un certain temps, on pourrait dire que les SGBDOO de deuxième génération comme celui-ci sont en quelque sorte un retour vers le futur.

Références

- db4o (2007)** : Page d'accueil de db4o et db4objects, Inc. : www.db4o.com (dernière visite le 22 octobre 2007).
- db4o2D (2007)** : db4o2D project space, <http://developer.db4o.com> (dernière visite le 22 octobre 2007).
- Intel (2006)** : Le projet PlaceLab Project, PlaceLab — un système de localisation respectant la vie privée, www.placelab.org (dernière visite le 22 octobre 2007).
- JavaWPS (2007)** : Un système de positionnement en Java basé sur les signaux WLAN, www.gis.hsr.ch/wiki/JavaWPS (dernière visite le 22 octobre 2007).
- JTS (2007)** : Java Topology Suite, une API de fonctions et prédicats spatiaux 2D, www.vividsolutions.com/jts/JTSHome.htm (dernière visite le 22 octobre 2007).
- LINQ (2007)** : Le Projet LINQ, <http://msdn.microsoft.com/netframework/future/linq> (dernière visite le 22 octobre 2007).
- ODBMS.ORG (2007)** : Un portail indépendant sur les technologies de bases de données objet, www.odbms.org, (dernière visite le 22 octobre 2007). OGC (2006) : Spécifications d'implémentation de l'information géographique de l'OpenGIS - Accès aux Objets Simples - Partie 1 : Architecture commune, www.opengeospatial.org/standards/sfa (dernière visite le 22 octobre 2007).
- PolePosition (2007)** : Un banc d'essai des bases de données OpenSource, www.polepos.org (dernière visite le 22 octobre 2007).

Stefan Keller
Institute for Software and GISpunkt
University of Applied Sciences Rapperswil (UAS HSR)
CH-8640 Rapperswil, Switzerland
www.ifs.hsr.ch
Stefan Keller est un professeur en systèmes d'information enseignant les technologies de bases de données et la programmation.
[stefan.keller AT hsr.ch](mailto:stefan.keller@hsr.ch)

Google Summer of Code pour la géomatique

Jan Ježek, traduit par Sidonie Christophe

Résumé

Ces dernières années, le module de référencement de GeoTools est devenu un des outils les plus puissants de transformation de systèmes de coordonnées de référence dans le monde SIG Java. Le module de référencement, avec le module de "couverture", possède un outil réellement puissant pour les opérations, du genre re-projection et transformation de données matricielles.

Le but de cet article est de décrire une nouvelle fonctionnalité qui a été développée par l'auteur pendant les projets des Google Summer of Code 2006 et 2007. Ses potentialités d'utilisation seront aussi discutées en rapport avec les besoins spécifiques sur les systèmes de coordonnées de référence utilisés en République Tchèque. Il sera aussi question du Google Summer of Code lui-même.

La nouvelle fonctionnalité qui a été développée, et est toujours en développement, concerne les méthodes basées sur l'interpolation. Ces procédures sont généralement utilisées dans les cas où la transformation entre les systèmes de coordonnées n'est pas une relation mathématique exacte (comme la projection cartographique ou la transformation affine par exemple).

Ce sujet est étroitement lié à la rectification des cartes anciennes, ainsi qu'à la transformation des systèmes de coordonnées de référence pour ces référentiels, qui ont été dérivées avant que les techniques GPS soient clairement définies, ce qui rend leur transformation dans les systèmes globaux comme le WGS 84, problématique et beaucoup moins précise que prévu.

Google Summer of Code pour la géomatique

Le Google Summer of Code (GSoC) est un événement bien connu, rassemblant des étudiants intéressés par les logiciels open-source avec des développeurs de projets variés. D'énormes projets comme KDE, Ubuntu Linux, Apache Software Foundation, etc. participent à ce programme chaque année. Le but

du GSoC est de faire en sorte que les étudiants intéressés participent à ce projet. Google joue le rôle de sponsor (entre autres) au sein du programme en entier et offre des bourses aux étudiants qui participent avec succès. Pour plus de détails sur le GSoC, voir (4).

La troisième édition de cet événement a eu lieu à l'été 2007. La communauté géospatiale open source a commencé à participer au GSoC en 2006 quand Refractions est devenu organisateur. Les projets concernés se concentraient sur la librairie GeoTools et la plateforme SIG uDig. Pour plus d'information au sujet du GSoC 2006 encadré par Refractions, voir (5).

L'OSGEO a rejoint le programme cet été 2007 et a permis d'obtenir le parrainage de 12 étudiants qui ont contribué à une large série de projets FOSS4G (GRASS, GDAL, GeoTools, GeoServer et uDig). Pour des informations détaillées, voir (6).

Fonctions additionnelles pour les conversions de systèmes de coordonnées dans GeoTools et uDig

Cette partie décrit le travail réalisé par l'auteur pendant les GSoC 2006 et 2007. Le projet était encadré par Jesse Fischer et Martin Desruisseaux.

Le package GeoTools Referencing présente un des outils les plus puissants permettant de re-projeter et réaliser des transformations dans le monde des SIG Java. Le package respecte les spécifications d'implémentation OGC (3). Il offre aussi un mécanisme de plugin, qui permet aux utilisateurs de se connecter à des bases persistantes de référentiels et de paramètres de projection, comme une base de données EPSG (2).

Le projet sur lequel j'ai travaillé durant 2006 et 2007 concernait de nouveaux algorithmes de conversion de systèmes de coordonnées. Le but était d'implémenter des outils qui aident à résoudre la tâche de transformation inverse : nous connaissons quelques coordonnées dans la source et le système de coordonnées de référence de la cible (coordonnées mappées), et nous cherchons une définition de la transformation.

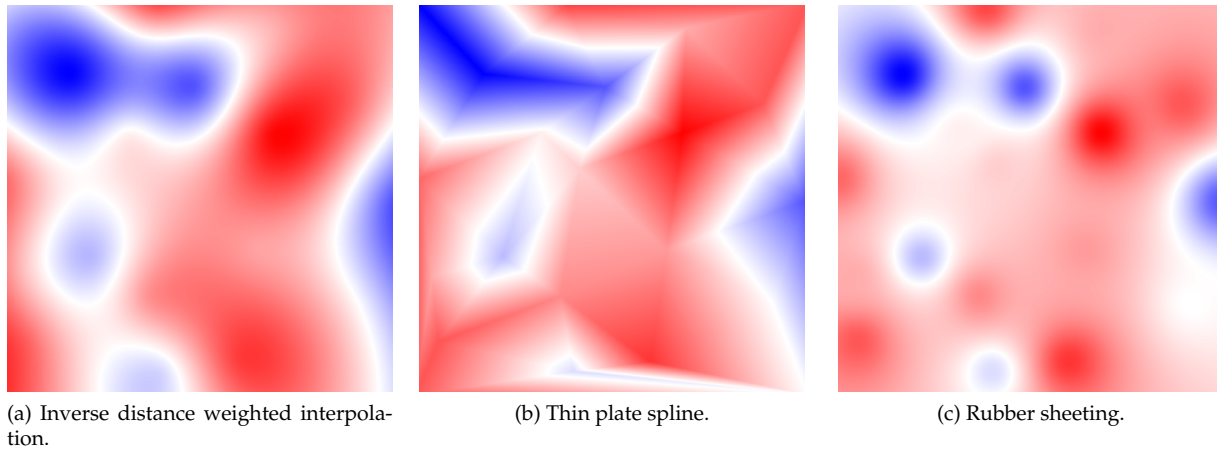


FIG. 1: Output of three interpolation methods

Transformations linéaires

Il existe deux méthodes de transformation, inégalement définies, à partir des coordonnées connues dans la source et le système de coordonnées de référence (CRS). Ces méthodes peuvent être divisées en deux catégories : les transformations linéaires et les méthodes sans résidus.

Les nouveaux outils développés peuvent être utilisés pour calculer les paramètres de transformation pour ces méthodes linéaires :

- Transformation similaire
- Transformation linéaire
- Transformation affine
- Transformation projective
- Transformation des vecteurs-position (Bursa Wolf) (7 paramètres comme en 3D)

Ces méthodes sont inégalement définies par différents nombres de positions cartographiées (une transformation dite similaire en nécessite 2, par exemple). Quand il y a plus de coordonnées que nécessaires, on utilise la méthode des moindres carré pour minimiser le carré de la distance entre la cible et le point source transformé. Actuellement, seule la distance cartésienne est prise en compte.

Algorithmes non résiduels

Un autre ensemble d'algorithmes qui ont été implémentés concerne les possibilités de calculer des transformations qui vont faire correspondre exactement les positions sources et les positions cibles, peu importe combien de positions ont été définies (on appelle ceci une transformation par déformation). Après avoir étudié les possibilités de définir de telles

méthodes grâce aux conventions d'une base de données EPSG, nous avons choisi l'approche suivante.

Pour définir ces transformations, on utilise, comme dans la base EPSG, une méthode basée sur une grille régulière de coordonnées. Dans cette grille, on applique une interpolation bi-linéaire ; donc une fois qu'on connaît les valeurs de grille, on peut appliquer la transformation assez rapidement. La famille des méthodes basées sur des grilles comprend :

- NADCON (EPSG dataset coordinate operation method code 9613) qui est utilisé par le US National Geodetic Survey pour des transformations entre les systèmes US
- NTv2 (EPSG dataset coordinate operation method code 9615) qui a été créé à l'agence nationale de cartographie du Canada et qui a été ensuite adoptée par l'Australie et la Nouvelle Zélande.
- OSTN (EPSG dataset coordinate operation method code 9633) utilisé en Grande Bretagne.

Pour plus d'informations, voir [(2)].

Trois algorithmes ont été implémentés pour permettre aux utilisateurs de calculer cette grille. Ces méthodes sont :

Interpolation par inverse de la distance pondérée :

Les valeurs offset sont calculées selon la distance depuis les positions cartographiées connues

Interpolation par splines de plaques minces (thin plate spline) :

Le nom de cette méthode fait référence à une analogie physique impliquant la flexion d'une mince feuille de métal. En physique, la déflexion est en z , perpendiculaire au plan (1). Les

décalages dans les deux directions (x et y) sont calculés de cette manière.

Méthode de la couverture de caoutchouc (Rubber Sheet)

La surface est divisée en triangles particuliers en utilisant l'algorithme de Delaunay sur le champ des positions cartographiées. Ensuite, une transformation affine est appliquée sur chaque triangle. Cette méthode devrait aussi être appliquée par morceaux, mais le calcul de la grille la rend plus générale, réutilisable et plus rapide.

Détails d'implémentation

Les fonctions décrites ont été conçues afin d'être intégrées dans le module de référencement de GeoTools.

Les algorithmes permettent de générer des fichiers de grilles qui peuvent être réutilisés dans d'autres logiciels — dans tous ceux qui supportent les méthodes EPSG expliquées ci-dessus. La transformation basée sur grille est conçue pour être utilisée avec la transformation par déformation du Java Advance Imaging (JAI) Ceci améliore grandement les performances, spécialement lors de la transformation de jeux de données raster (JAI utilise une librairie native).

D'autres résultats intéressants apparaissent quand on essaie de visualiser les grilles calculées en convertissant les valeurs calculées en images. De cette façon, on peut voir la distribution des décalages et aussi les différences entre les diverses méthodes appliquées. Vous pouvez voir les grilles interpolées par les 3 méthodes utilisant le même ensemble de positions cartographiées sur la figure 1.

Toutes les fonctionnalités décrites sont en ligne.⁵

La migration dans GeoTools 2.5 se fera dans les mois à venir. D'autres documentations et des exemples de codes peuvent être récupérés sur le site GeoTools.⁶

Plugin uDig

Pour finir, nous avons créé un plugin uDig qui permet aux utilisateurs de calculer et d'appliquer les méthodes décrites. Le plugin présente une interface graphique pour accéder à ces fonctions. Un premier brouillon de ce plugin est disponible sur le site de la communauté (see (7)) et vous permet de transformer

des données vecteur en utilisant seulement quelques méthodes (ceci a été développé pendant le GSoC 2006). Un plugin qui inclus toutes les caractéristiques décrites, a été développé seulement pour la version CVS de uDig ; il est toujours en cours de modification et donc la version stable sera opérationnelle pour la version 1.2 de uDig.

Bibliographie

- [1] Wikipedia, the free encyclopedia, Thin plate spline, available at http://en.wikipedia.org/wiki/Thin_plate_spline, [29.10.2007].
- [2] EPSG, Coordinate Conversions and Transformation including Formulas, available at <http://www.epsg.org>, [29.10.2007].
- [3] OGC, Coordinate Transformation Service Implementation Specification, available at <http://www.opengeospatial.org/standards/ct>, [29.10.2007].
- [4] Google, Google Summer of Code, available at <http://code.google.com/soc/>, [29.10.2007].
- [5] Refrations Research, Refrations Research project ideas, available at <http://code.google.com/soc/2006/refract/about.html>, [29.10.2007].
- [6] OSGeo, OSGeo project ideas, available at <http://code.google.com/soc/2007/osgeo/about.html>, [29.10.2007].
- [7] Refrations Research, uDig update site, available at <http://udig.refrations.net/confluence/display/UDIG/uDig+Update+Site>, [29.10.2007].

Jan Ježek

h.jezek AT centrum.cz

⁵GSoC transformations page : <http://svn.geotools.org/geotools/trunk/spike/jan/gsoc-transformations/>

⁶GeoTools website : <http://geotools.codehaus.org/New+Transformation+Algorithms+for+GeoTools+and+uDig>

Intérêt thématique

Approche générique pour la gestion de standards de métadonnées

par Barde Julien et Egdington Duane et Desconnets Jean-Christophe

Introduction

La gestion des *Ressources Informationnelles*⁷ (RI) est une composante essentielle de la gestion des ressources environnementales. En effet, l'amélioration du *traitement des données* et des *prises de décisions* est fortement dépendante de la capacité à *localiser les RI pertinentes*.

Néanmoins, la localisation exhaustive des RI pertinentes est un challenge pour les utilisateurs qui doivent, pour cela, faire face aux contraintes suivantes :

- les RI sont *hétérogènes* (*langage, sémantique, syntaxe/formats, métadonnées, contraintes d'accès dues à leur rareté et à leur coût...*),
- les RI sont *distribuées* dans des *Systèmes d'Information (SI) hétérogènes* dont l'interopérabilité nécessite la résolution de *correspondances sémantiques et syntaxiques*.

L'objectif prioritaire pour améliorer la localisation des RI pertinentes consiste à gérer efficacement

les couples {*élément de métadonnée, valeur*} qui composent les fiches de métadonnées. En effet, une fois au courant des SI à interroger, la priorité pour localiser les RI pertinentes est la gestion des correspondances (syntaxiques) entre les éléments de métadonnée hétérogènes et de leurs valeurs hétérogènes (sémantiques). L'échelle globale du domaine environnemental ainsi que les contextes habituellement pluridisciplinaires des études associées augmentent fortement ces contraintes et le besoin de référentiels sémantiques et spatiaux.

Gestion des métadonnées

L'amélioration de la *gestion actuelle des métadonnées* est donc une priorité. D'une manière générale, on constate une utilisation croissante (bien qu'encore trop faible) de référentiels pour homogénéiser la terminologie des *éléments de métadonnées*, les *relations* qu'ils entretiennent et les *valeurs* qui leur sont associées. En conséquence, la qualité des outils de gestion de métadonnées doit être évaluée autant sur le respect d'une implémentation conforme aux spécifications des standards de référence (syntaxique : les

⁷une *ressource informationnelle* (RI) recouvre l'ensemble des données, informations, connaissances que les acteurs produisent, recherchent et traitent. Indépendamment de leur format (numérique, papier...), selon les acteurs, ce terme peut par exemple représenter un rapport, une carte, une image, une photo, une vidéo, un jeu ou une série de données, une base de données, un modèle...

éléments de métadonnées et leur structure) que sur la capacité à leur associer des valeurs issues de référentiels sémantiques pour éditer des instances.

L'hétérogénéité des jeux d'éléments de métadonnées perturbe l'interopérabilité syntaxique des SI :

- les standards utilisent souvent un noyau d'éléments de métadonnées principaux identique où les éléments font référence aux mêmes concepts en utilisant cependant des termes différents (ces éléments principaux répondent aux questions suivantes : Où? Quoi? Quand? Qui?...),
- l'utilisation de standards dont les portées sont redondantes génère des problèmes de mises en correspondances (étant donné qu'une même sorte de RI peut être potentiellement décrite avec différents standards),
- la mise en place récente de nouveaux standards de métadonnées internationaux s'appuyant sur les standards nationaux/locaux préexistants les rends obsolètes et génère des problèmes d'archivages (stockage des métadonnées en l'état ou mises en correspondances vers les standards de référence),
- l'usage systématique récent de XML Schemas pour standardiser l'implémentation des standards diminue la redondance de leurs portées et prévient des mauvaises interprétations de leur implémentation. En effet, les standards peuvent être utilisés à présent comme des bibliothèques de types (?) (c'est la cas, par exemple, des standards de l'OGC comme la récente mise à jour de l'ISO 19115 par l'ISO 19139, SensorML...).

L'hétérogénéité potentielle des valeurs associées aux éléments de métadonnées interfère avec l'interopérabilité sémantique des SI si on ne peut pas les contrôler (quel que soit le standard de métadonnées choisi) : leur contrôle par l'utilisation complémentaire de référentiels est un problème central pour améliorer la description des RI et donc leur localisation. En particulier, une gestion fine des valeurs associées aux éléments de métadonnées principaux est impérative puisque ces derniers sont utilisés prioritairement pour répondre aux requêtes les plus communes des utilisateurs. La gestion des descriptions suivantes est particulièrement cruciale tant elles sont complexes et ambiguës :

- *description terminologique* avec des vocabulaires contrôlés/référentiels sémantiques (multilingues) pour valuer les éléments de métadonnées de type "keyword". De plus, l'utilisation de tels

référentiels aide à la mise en place de vocabulaires communs dans des contextes pluridisciplinaires,

- *description spatiale* avec des référentiels spatiaux/géographiques présentés avec des GUIs adaptés pour améliorer l'utilisation complexe de l'information géographique (IG) (en particulier lorsqu'il s'agit d'utiliser des formats comme GML, WKT...).

Le besoin d'outils de gestion de métadonnées multi-standards intégrant le contrôle de ces valeurs clés va croissant. Les demandes intra- ou inter-instituts/projets qui gèrent souvent plus d'une sorte de RI et ne peuvent se contenter d'un unique standard vont dans ce sens.

De plus, si on prend en compte les tâches et besoins respectifs des utilisateurs et des développeurs pour la gestion des métadonnées, il apparaît que la plupart d'entre eux sont similaires et indépendants des standards considérés.

Tâches et besoins des utilisateurs

Selon leurs rôles (administrateur...), quel que soit le standard, les tâches des utilisateurs associées à la gestion d'un standard de métadonnées consiste habituellement en :

- la mise en place de profils des standards de métadonnées pour couvrir les besoins de leurs utilisations spécifiques,
- l'édition d'instance de ces standards pour décrire leur RI (en associant des valeurs aux éléments de métadonnées retenus dans les profils),
- la localisation (et l'éventuelle acquisition des RI selon les contraintes de droits d'accès associés) des RI pertinentes pour un travail donné en utilisant un moteur de recherche multi-critères unique qui autorise des requêtes spatiales évoluées,
- l'import/export d'instances de standards de métadonnées (habituellement au format XML),

Les besoins des utilisateurs se concentrent sur la convivialité (actuellement perfectible) des outils proposés pour exécuter ces tâches. La plupart du temps, les utilisateurs expriment le besoin d'un outil centralisé avec des GUIs dont l'ergonomie conviviale sera conservée d'un standard à l'autre puisqu'elle a pour objectif d'aider à réaliser les mêmes tâches. En effet, l'hétérogénéité actuelle des logiciels ou SI, qui implémentent pourtant les mêmes standards, augmente les efforts d'adaptation des utilisateurs alors que les objectifs visés sont paradoxalement les mêmes (étant entendu que l'utilisateur doit passer -voire perdre-

du temps pour se familiariser avec l'ergonomie de chaque outil). Enfin, un maximum d'aide est attendu (*interface cartographique en ligne, vocabulaires contrôlés, calendrier...*) lorsque les valeurs à associer aux éléments de métadonnées sont trop complexes à fournir sous la forme demandée par les standards (GML...). Ces cas d'utilisation sont illustrés par un diagramme UML dans la présentation de la conférence ([ici](#)).

Tâches et besoins des développeurs

De la même manière, les tâches et besoins des développeurs s'avèrent identiques d'un standard à l'autre. *Les tâches des développeurs* consistent à :

- conjuguer la satisfaction des besoins des utilisateurs tout en se conformant aux spécifications des standards,
- la gestion de correspondances entre éléments de métadonnées principaux de différents standards pour répondre aux requêtes de base aussi efficacement que possible,
- intégrer et gérer des référentiels sémantiques et spatiaux existants (comme AGROVOC, GEMET, ...) pour garantir la qualité des descriptions des RI et pouvoir gérer ultérieurement les procédures d'expansions de requêtes. En effet, la localisation efficace des RI implique la gestion d'autant de requêtes différentes que de sortes de SI à interroger. Les réponses à ces requêtes sont d'autant plus difficiles à obtenir que les termes et les objets géographiques utilisés dans les descriptions n'ont pas été contrôlés en amont,
- la mise en place parallèle d'une infrastructure de données spatiales riche permettant la gestion simultanée des RI décrites par les métadonnées et leurs éventuels traitements ultérieurs.

Les besoins des développeurs concernent la diminution des efforts de développements pour implémenter (automatiquement) un nouveau standard (?) en répondant aux besoins identiques des utilisateurs de la même manière : en réutilisant un jeu de scripts unique et les mêmes composants pour les assister dans la réalisation de leurs tâches (WMS...). Ceci correspond finalement au besoin d'une *approche dite générique* car indépendante du contenu propre aux standards implémentés (?).

Approche générique vs approche spécifique

Les implémentations spécifiques traditionnelles sont obtenues par une traduction directe des spéci-

fications des standards en des *modèles physiques de données* qui constituent des systèmes de gestion des données hétérogènes et nécessitent donc des jeux de scripts hétérogènes pour les exploiter (voir figure 1).

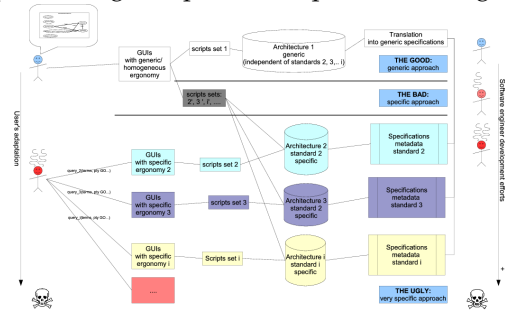


FIG. 1: Bénéfice d'une approche générique pour la gestion multi-standards

Par exemple en utilisant une génération de code (automatique ou non) SQL basée sur les spécifications des standards en UML ou XML-S/XSD, les modèles physiques de données qui en résultent vont être fortement hétérogènes (avec des noms de tables et de colonnes qui correspondent aux termes employés pour désigner le nom des éléments de métadonnées dans chaque standard spécifique). Les jeux de scripts en charge de traiter le contenu de ces tables doivent alors être adaptés à ces jeux de termes spécifiques (alors qu'il s'agit de réaliser les mêmes tâches, et sont donc hétérogènes d'un standard à l'autre).

En accord avec les listes de tâches et de besoins dressées auparavant, la figure 1 illustre le bénéfice d'une *approche générique* pour les utilisateurs comme pour les développeurs.

Jusqu'ici, les outils existants ne couvrent pas l'ensemble de ces besoins d'une manière satisfaisante, principalement parce qu'ils sont basés sur des approches spécifiques.

Des modèles génériques pour une gestion efficace des couples {élément de métadonnée, valeur}

Nous présentons dans cette section les modèles génériques actuellement préconisés et implémentés pour mettre en place un outil de gestion de métadonnées multi-standards doté de composants additionnels visant à assister les utilisateurs pour contrôler les valeurs des éléments de métadonnées principaux. En particulier, les modèles présentés seront focalisés sur les valeurs des éléments de métadonnées les plus cruciaux qui concernent les *descriptions thématiques et*

spatiales.

Un modèle générique pour gérer des standards de métadonnées hétérogènes

L'objectif est d'ajouter une étape intermédiaire en proposant un patron générique (modèle conceptuel présenté dans le diagramme de classes UML de la figure 2) qui fournit une couche d'abstraction permettant de décrire n'importe quel standard de métadonnées et donc de mettre en place un système de gestion de métadonnées générique. Ce modèle intègre également le nécessaire contrôle des valeurs, en particulier celles associées aux éléments de métadonnées principaux. Pour cela, le modèle proposé en figure 2 considère qu'un standard de métadonnées est un inventaire d'éléments de métadonnées structurés dont le contenu est rempli par des valeurs potentiellement contrôlées (selon des règles conformes aux spécifications des standards ou complétées par volonté des développeurs selon le contexte d'implémentation).

Cette approche peut finalement être comparée à l'objectif du DOM⁸ qui manipule les mêmes concepts pour gérer la structure des noeuds et des valeurs qui leur sont associées dans tout type de document. L'objectif étant dans ce cas restreint au cas spécifique des standards de métadonnées.

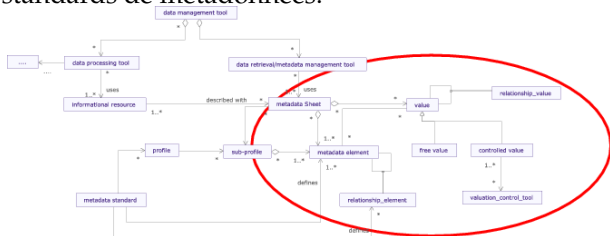


FIG. 2: Expression générique des spécifications d'un standard de métadonnées

Toutefois, un standard n'a pas pour objectif de définir ni de contrôler toutes les valeurs potentielles associées aux éléments de métadonnées (principaux en particulier), souvent un standard se contente de suggérer leur contrôle. Le contrôle des valeurs (termes, objets géographiques...) et de leurs relations fait l'objet d'autres standards spécifiques. L'intégration de ces outils complémentaires pour gérer efficacement des valeurs spécifiques est donc bien une tâche du développeur.

Nous nous focaliserons par la suite sur le cas spécifique de la gestion des valeurs associées aux descriptions spatiales et thématiques. Nous spécifions donc le modèle précédent pour permettre la gestion

⁸Document Object Model

complémentaire de ces valeurs particulières et de leurs relations.

Un modèle générique pour la gestion des valeurs (spatiales et thématiques) hétérogènes

Les différentes sortes de couples {élément de métadonnée, valeur} sont plus ou moins cruciales pour la localisation de données. En particulier, certaines valeurs sont spécialement complexes à contrôler. Parmi elles, les descriptions thématiques et spatiales sont toutes les deux et cruciales et complexes puisqu'elles correspondent aux éléments de métadonnées principaux impliqués dans l'essentiel des requêtes des utilisateurs (elles répondent aux questions Où? et Quoi?).

Nous allons donc nous concentrer sur ce cas particulier en proposant une formalisation des propriétés et des relations des concepts thématiques et spatiaux (illustrée en figure 3).

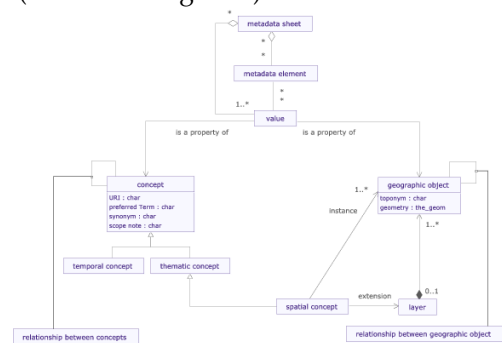


FIG. 3: Propriétés et relations entre concepts thématiques et spatiaux

L'utilisation de termes pour valuer des éléments de métadonnées principaux peut prêter à confusion :

- les utilisateurs formalisent souvent leurs descriptions ou recherche de RI en utilisant des termes similaires à ceux de l'exemple suivant : "espadon, température de surface, Madagascar, hiver... ",
- la gestion des relations sémantiques est essentielle puisqu'elle va par exemple permettre d'explorer les relations sémantiques de ces termes pour étendre ce type de requêtes et améliorer la localisation exhaustive des RI pertinentes. Par exemple, en étendant la requête à la recherche de RI décrites avec le terme ("Xiphias gladius... ") qui est un synonyme d'"espadon... ", étant entendu que ces deux

- termes désignent le même *concept* (qui pourrait tout aussi bien être désigné par une représentation graphique comme une image, une photo...),
- le cas d'un *toponyme* doit être considéré à part puisque plus complexe. En effet, un *toponyme* peut être considéré à la fois comme un terme et comme une description (géo)graphique pointant un *objet géographique*. Par exemple, l'objet géographique associé au terme/toponyme "*Madagascar*" pourrait tout autant avoir été désigné dans une interface cartographique sans rien changer à l'intention de l'utilisateur.

Pour limiter ces confusions, nous suggérons une gestion des relations sémantiques et spatiales entre les concepts thématiques et spatiaux ainsi que les objets géographiques qui intègrent la complexité des exemples précédents en s'appuyant sur la formalisation suivante : "*un concept spatial est une sorte de concept thématique dont les instances sont des objets géographiques*" (?), voir figure 3. Toutefois, il convient de préciser qu'un objet géographique n'est pas forcément désigné par un terme ou un toponyme du langage courant.

La figure 4 synthétise et précise le contenu des modèles précédents en un modèle unique permettant d'améliorer l'implémentation générique des standards de métadonnées en intégrant par défaut l'emploi de référentiels qui permettront de contrôler ces valeurs essentielles.



FIG. 4: Synthèse de l'approche générique proposée

Ce modèle a été conçu pour pouvoir s'appuyer sur les initiatives de référence actuellement, aussi bien pour les *standards de métadonnées* (XML Schemas, DTD...), que pour la *sémantique* (travaux du W3C sur le (Web) Semantic avec les standards : SKOS⁹/RDF/OWL) ou l'*information spatiale* (prise en charge des principaux formats pour l'information géographique). En particulier, ce modèle permet, au sein d'une même architecture, de *lier physiquement les éléments de métadonnées avec des ontologies* qui permettent de contrôler les valeurs associées (incluant

⁹sorte de RDF lié aux normes ISO 2788 et 5964

les descriptions spatiales) et d'étendre les requêtes efficacement.

De la même manière, il est possible d'ajouter des contrôles pour d'autres valeurs cruciales : en particulier les *descriptions temporelles et de contacts* associées aux RI. Ces descriptions sont aussi importantes (puisqu'elles permettent de répondre aux questions *Qui?* et *Quand?* mais moins complexes à contrôler. Les outils associés au contrôle de ces valeurs sont généralement des composants du type *calendrier* ou *annuaire* (en charge d'assister la désignation d'une date/période ou d'acteurs associés à la RI décrite).

La gestion de ces référentiels supplémentaires peut être faite indépendamment des standards de métadonnées implémentés. Cependant, les différents termes qui désignent les mêmes éléments de métadonnées principaux dans différents standards sont finalement valués en utilisant les différents objets gérés dans ces référentiels (traités pour calculer les valeurs sous la forme requise par les spécifications). Il est donc souhaitable de les gérer ensemble. En parallèle, en mémorisant dans une *table d'indexation générique* dédiée, quels objets (concepts, objets géographiques, dates/périodes, acteurs...) ont été associés aux RI pour les descriptions thématique, spatio-temporelle, humaine, il devient possible de répondre à l'essentiel des requêtes des utilisateurs indépendamment des standards qui ont été utilisés pour les décrire, en se basant sur des descriptions aussi riches que possibles (géométries 2D/3D, concept au lieu des termes...) qui ouvrent de nouvelles perspectives pour résoudre et étendre des requêtes complexes.

Implémentation des modèles avec des logiciels open source

Nous présentons dans cette dernière section l'intérêt d'une implémentation basée sur des applications *open source*.

Choix techniques sous-jacents

Les implémentations actuelles de l'architecture préconisée sont basées sur le logiciel open source *MDWeb* qui utilise lui-même d'autres applications *open source*. *MDWeb* :

- est un outil de catalogage *multi-standard* et *multi-langue* qui implémente une approche générique (comme *M3Cat*, *MetaCat*...),

- utilise une architecture client-server *trois-tiers* avec :
 1. des *GUIs* (dans des navigateurs Web) et des composants additionnels (pop-ups) pour assister convivieusement l'édition d'instances de standards ou la recherche de métadonnées, notamment :
 - la *description spatiale* avec des interfaces cartographiques en guise d'interfaces/*GUIs* conviviaux permettant en prime de visualiser les RI spatiales décrites : *Mapserver / Mapbuilder*,
 - la *description thématique* avec des *GUIs* de gestion de vocabulaires contrôlés pour mettre en place et parcourir des thésaurus / ontologies : "composant maison".
 2. *scripts de l'application* (PHP/JavaScript/XML avec Apache comme serveur Http),
 3. *stockage de données* : un *SGBDR* pour la gestion des standards de métadonnées & IG & instances de métadonnées associées & vocabulaires contrôlés : *Postgres* avec *Postgis* pour l'information spatiale gérée localement (WMS pour l'IG distante...). Import de fichiers *SKOS* dans *Postgres* en utilisant l'API java *JENA*. Répertoires XML.

La présentation de la conférence illustre les principales caractéristiques de ce type d'architecture trois-tiers pour un outil de gestion de métadonnées multi-standards et détaille l'infrastructure physique de données.

Exemples de jeu générique possible pour les GUIs

En utilisant *MDWeb* comme base pour implémenter cette approche, il est donc possible de couvrir à la fois les besoins des utilisateurs et des développeurs, en particulier en ayant un jeu unique de *GUIs* (?) tel qu'illustre en figure 5 pour :

- importer n'importe quel nouveau standard de métadonnées en traduisant préalablement ses spécifications (formelles ou non) pour pouvoir les intégrer dans le MPD (pour le moment seules les spécifications au format XML Schemas sont importées automatiquement),
- mettre en place des *profils* des standards qui ont été importés,

- édition d'instances de métadonnées assistée et contrôlée par des composants complémentaires communs à tous les standards sous formes de *GUIs* conviviaux (en particulier pour les descriptions thématique et spatiale) (voir illustration en figure 5),
- un *moteur de recherche multi-critères générique*,
- l'*import/export* d'instances de métadonnées standardisées (généralement XML).



FIG. 5: GUI pour l'édition d'instances de métadonnées

Conclusion et perspectives

La localisation de RI peut être largement améliorée par une meilleure gestion des métadonnées grâce à l'implémentation d'approches génériques qui permettent la mise en place d'architectures uniques (*GUIs*, jeu de scripts, base de données) capables de gérer aussi bien :

- des *standards de métadonnées hétérogènes* (import, profils, édition...),
- des *valeurs hétérogènes* : en particulier des *termes* et des *descriptions spatiales* contrôlés pour décrire les *éléments de métadonnées principaux*,
- une *table d'indexation commune* qui duplique les *valeurs* associées aux *éléments de métadonnées principaux* sous une forme homogène visant à améliorer l'efficacité du *moteur de recherche* (en évitant les *wrappers*), en particulier par des mécanismes d'expansions de requêtes,

- les *RI spatiales décrites par les métadonnées* qui peuvent ainsi être traitées après les avoir localisées : soit localement soit à distance en utilisant des protocoles interopérables ou des clients riches (*WMS, Qgis, Udig...*).

Cette sorte d'architecture est pertinente pour satisfaire aussi bien les *besoins et les tâches des utilisateurs que ceux des développeurs en minimisant les efforts d'adaptation et de développement* tout en intégrant en même temps des outils complémentaires de contrôle des valeurs des éléments de métadonnées principaux.

La *localisation des RI* est ainsi améliorée. En particulier, en intégrant des outils de gestion des descriptions sémantiques et spatiales dans une même architecture, la localisation des données va pouvoir s'appuyer plus simplement sur des expansions de requêtes indispensables à la qualité des recherches. Il devient donc possible de se focaliser sur des cas

d'utilisation type impliquant la gestion de relations spatiales et sémantiques comme : "trouver toutes les RI à moins de 1 mile de ce capteur (objet géographique) avec des mesures de tels paramètres physiques... ", en s'appuyant en particulier sur des objets géographiques 2D ou 3D et des concepts, plutôt que des termes. De plus, en utilisant des relations standardisées (sémantiques par le W3C ou spatiales par l'OGC), les différentes requêtes deviennent réutilisables par d'autres systèmes/outils similaires.

Plus généralement, l'utilisation massive de standards de l'OGC et de logiciels qui les implémentent, accroît la capacité de l'application proposée à interopérer avec des systèmes extérieurs.

Barde Julien

Monterey Bay Aquarium Research Institute (MBARI)

<http://www.mbari.org/staff/julien/julien@mbari.org>

Vers des services web dédiés à la cartographie thématique

Abson Sae-Tang, Olivier Ertz

Introduction

Les standards ouverts favorisent l'interopérabilité des systèmes; et l'OGC (Open Geospatial Consortium) est l'organisme de normalisation qui spécifie les standards permettant aux systèmes géographiques d'opérer de concert. Parmi les spécifications connues, l'OGC définit des standards comme Web Map Server (WMS), Web Feature Server (WFS), Geographic Markup Language (GML), ainsi que Styled Layer Descriptor (SLD) pour résoudre le problème de partage et d'interopérabilité des données spatiales. Le projet décrit dans cet article met l'accent sur l'utilisation de SLD pour favoriser l'interopérabilité des systèmes géographiques dédiés à la cartographie thématique.

Qu'est-ce que SLD ?

SLD signifie Styled Layer Descriptor (Descripteur de Style de Couche), il s'agit d'un encodage qui étend la spécification WMS pour permettre de personnaliser la symbologie à appliquer sur les entités spatiales. Celle-ci permet aux utilisateurs (ou systèmes) de déterminer avec quelles couleurs ou symboles quelles entités ou couches de données doivent être graphiquement représentées. SLD répond au besoin de l'utilisateur (ou logiciel) de contrôler le rendu visuel des données géographiques sur une carte. Les applications libres de la géoinformatique respectent et se basent fortement sur ces standards ouverts, et SLD est notamment implémenté par Geoserver, Mapserver, deegree parmi d'autres.

La prochaine version de ce standard est connue sous le nom de Symbology Encoding Specification qui est en cours de définition. La possibilité de définir des règles de style nécessite un langage de style que client et serveur doivent comprendre. Symbology Encoding fournit ce langage, alors que SLD en tant que profil WMS ne permet que l'application des règles de style aux couches WMS.

SLD convient-il à la cartographie thématique ?

Concrètement, SLD est une spécification utile et complète pour définir le style de rendu de vos cartes. Pour chaque couche vous pourriez dire "colorie toutes les entités ligne en bleu", "dessine tous les polygones avec un bord noir rempli de jaune clair", ou même "utilise de petits triangles sur tous mes points". Aussi, est-il possible de définir des styles plus complexes. Vous pouvez définir des règles de style basées sur les attributs des entités de la couche. Pour une couche de routes, vous pouvez définir les styles, ligne rouge de trois pixels de large pour les autoroutes, ligne noire de deux pixels pour les voies rapides, ligne noire de un pixel pour les routes secondaires, et ce grâce à un attribut indiquant le type de route.

Mais ce langage de style est-il aussi efficace pour la cartographie thématique? C'est-à-dire pour des cartes choroplèthes (Figure 1a), de symboles proportionnels (Figure 1b), superposés (Figure 1c), juxtaposés (Figure 1d), de secteurs (Figure 1e), de diagrammes en barres (Figure 1f), etc. Ces cartes thématiques utilisent principalement des règles de style basées sur les attributs d'entités pour transformer une valeur statistique en une représentation graphique sur la carte. Ainsi, ce projet représente tout d'abord une étude sur le potentiel de SLD pour répondre à des besoins en cartographie thématique.

Avec la spécification actuelle de SLD, il est plus ou moins possible de décrire des cartes thématiques, mais avec certains inconvénients :

Carte choroplèthe : une règle avec un filtre par classe (pour définir les limites de classe), chaque règle ayant son "symboliseur de polygone" (PolygonSymbolizer) avec la couleur de remplissage à appliquer.

Symboles proportionnels : un "symboliseur de point" (PointSymbolizer) avec un marqueur graphique, interne au système comme un cercle, une couleur de remplissage, et la taille du cercle contrôlée par une donnée attributive.

Symboles bivariés : un savant mélange entre les filtres et couleurs de la carte choroplèthe, et un "symboliseur de point" pour les symboles proportionnels. La taille des cercles et la couleur de

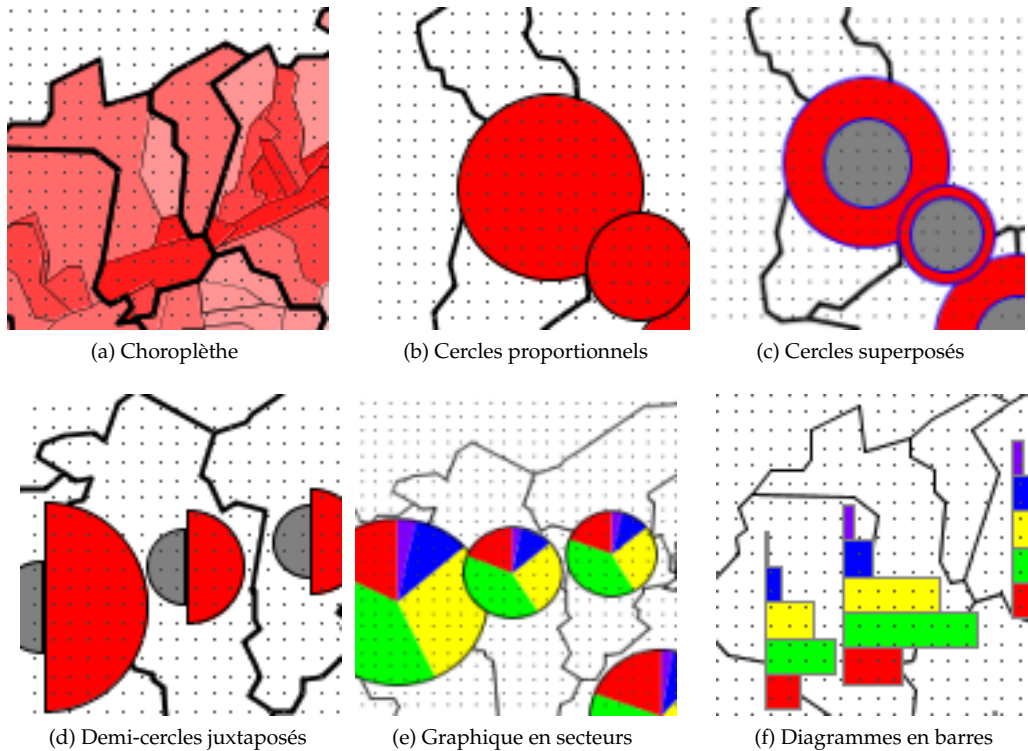


FIG. 1: Cartes thématiques

remplissage sont contrôlées par deux données attributaires.

Symboles juxtaposés : deux règles avec un filtre pour gérer l'ordre d'affichage (le fait que le plus petit symbole doit être devant le plus grand), et deux "symboliseurs de point" par règle, chacun ayant sa taille contrôlée par une donnée attributaire.

Pour des cartes de symboles juxtaposés, de graphiques en secteurs, de diagrammes en barres, cela devient bien plus complexe voire désagréable à décrire avec SLD. En effet, comment décrire deux demi-cercles juxtaposés "accrochés" à un même point ou centroïde ? De même pour les tranches d'un camembert ou les barres d'un histogramme.

Vers une extension dédiée à la cartographie thématique

L'étude initiale sur le potentiel de SLD nous a conduits à l'idée d'une extension du langage de symbolologie. Nous l'avons appelé SLD-T (même si cela n'a rien à voir avec WFS-T). Une telle extension étend la grammaire afin d'introduire la terminologie propre au métier de la cartographie thématique, de

simplifier la description des cartes thématiques, et de réduire la verbosité ainsi que la redondance.

Pour l'essentiel, l'idée est de créer un Thematic-Symbolizer abstrait (comme le PointSymbolizer) qui sera inclus dans le schéma SLD par extension de l'élément Symbolizer existant dans une Rule, avec des spécialisations pour chaque type de carte thématique (Figure 2).

CategoryThematicSymbolizer : pour des cartes avec classification (ex. : choroplèthe et bivariable). Ce symboliseur est construit sur la base d'un élément ThematicCategory pour décrire le type de classification (par valeur unique, par intervalle de valeurs, etc.).

SimpleThematicSymbolizer : pour des cartes sans classification (ex. : des symboles proportionnels). Il s'agit d'une simple enveloppe des symboliseurs "standards" afin qu'ils puissent hériter d'éléments génériques utiles du Thematic-Symbolizer comme la priorité de symbole et le placement (voir MultiThematicSymbolizer).

MultiThematicSymbolizer : pour représenter plusieurs symboles thématiques pour une même entité (ex. : deux symboles superposés ou juxtaposés). Un symboliseur (BaseSymbolizer) est utilisé pour définir les propriétés communes

de rendu (comme Stroke et WellKnownName) et ce afin d'éviter toute redondance. Pour des symboles juxtaposés, le processus de rendu est géré par un ordre grâce à l'élément Priority qui peut être contrôlé par un attribut d'entité. Pour la juxtaposition de symboles, l'élément Point-Placement (inspiré du TextSymbolizer) permet de définir un point d'ancrage (AnchorPoint) et un déplacement (Displacement) pour chacun des symboles.

ChartThematicSymbolizer : pour des symboles en diagrammes (par ex. : graphique en secteur et diagramme en barres). Un marqueur thématique (ThematicMarker, sur l'idée du marqueur graphique Mark) est utilisé pour spécifier le type de diagramme (en secteurs ou en barres). Les éléments ChartParts décrivent les tranches ou les barres qui composent le diagramme graphique. Comme il est possible de définir une rotation pour tous les ThematicSymbolizer, il est facile de décrire un histogramme.

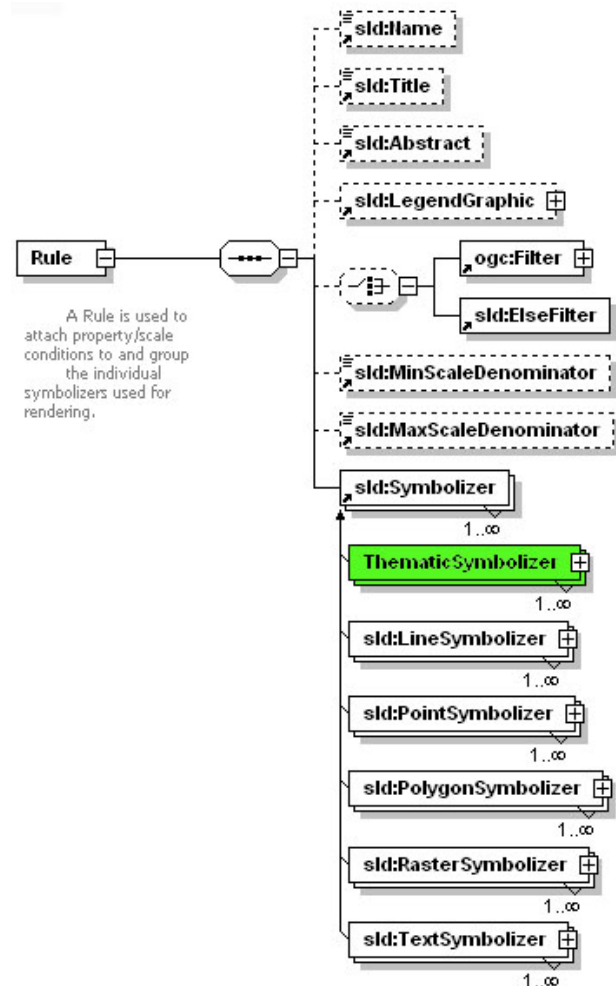


FIG. 2: L'élément Rule avec un nouveau symboliseur

Cet article ne survole que brièvement le contenu de l'extension. Le schéma XSD complet et le document de spécification illustré d'exemples est disponible en ligne.¹⁰

La spécification initiale SLD était à l'origine dédiée à WMS. L'utilisateur définit un document SLD qui est envoyé à un serveur WMS, retournant la couche géographique après avoir appliqué le style défini. Cependant, un tel document est aussi utile pour des applicatifs bureau conforme OWS (OGC Web Service). Celui-ci peut utiliser SLD pour l'appliquer au GML donné par une réponse WFS. Ainsi, un tel document de style et de symbologie a tout son intérêt à être utilisé côté serveur mais aussi côté client.

Considérons ce simple scénario : un cartographe réalisant une superbe carte pleine de sens avec son outil de bureau conforme SLD, et dès que celle-ci est prête, la publie sur son serveur WMS pour la par-

¹⁰Schema XSD et document de spécification : <http://geosysin.iict.ch>

tager. Il devra transférer sur le serveur les données ainsi que le document de style. Si l'interopérabilité concerne d'abord les données, cela concerne aussi finalement le partage des représentations cartographiques avec les styles appropriés ainsi que la symbologie, les utilisateurs voulant aussi partager des cartes thématiques.

Conclusion

L'étude initiale de ce projet a consisté à dépeindre le potentiel et comment utiliser la spécification SLD pour décrire des cartes thématiques. De plus, pour permettre à SLD de décrire des représentations thématiques complexes, le projet décrit une solution basée sur une extension de l'encodage de symbologie. L'extension a pour objectif d'introduire la terminologie propre à la cartographie thématique, de simplifier la description des cartes thématiques, et de réduire la verbosité ainsi que la redondance que SLD pourrait produire.

Comme preuve de concept, une première implémentation a démarré sur Geotools, seulement pour des symboles juxtaposés à ce jour. Cela a été assez facile grâce à un déjà bon support de SLD et modèle de rendu. Un avantage à utiliser Geotools réside dans le fait que cette librairie est utilisée par Geoserver côté serveur mais aussi côté client avec uDig. Mais il existe de nombreuses autres plateformes libres sur lesquelles une implémentation pourrait être menée, et aucun choix n'a encore été arrêté. Dans le futur, hormis une implémentation complète il est envisagé une mise en oeuvre dans le cadre d'un cas concret sur des besoins réels. Il est à noter qu'une implémentation complète requiert aussi le rendu des légendes de carte.

SLD est plus que jamais un élément clé pour une Infrastructure de Données Spatiales (IDS), et son acceptation est probablement entrain de franchir une étape. En effet, nous pouvons constater que de plus en plus de projets interprètent la spécification et proposent notamment un éditeur pour faciliter la vie de l'utilisateur qui souhaite sauvegarder sa carte dans ce format (Geoserver, Mapbender, OpenLayers, etc.). Aussi, du côté du "webmapping" orienté cartographie thématique, il est encourageant de voir MapServer à présent capable de créer des cartes thématiques complexes à base de diagramme en secteurs ou en barres, et de même, côté client, des applicatifs comme CarThema5 (basé sur gvSIG), JumpChart (basé sur OpenJUMP), OrbisGIS (basé sur Geotools et GDMS) bientôt capables de produire de telles cartes. Ce sont

toutes des initiatives allant dans le bon sens.

Remerciements

Les auteurs souhaitent remercier leurs partenaires, [MicroGIS SA](#) et [Camptocamp SA](#) pour leur collaboration.

Bibliographie

- [1] A. Sae-Tang, and O. Ertz, *Towards Web Services Dedicated to Thematic Mapping*, HEIG-VD, IICT/geo.SYSIN <http://geosysin.iict.ch>.
- [2] OGC Styled Layer Descriptor Specification, <http://www.opengeospatial.org/standards/sld>.
- [3] OGC Symbology Encoding Specification, <http://www.opengeospatial.org/standards/symbol>.
- [4] M.D. Teixeira, R. de Melo Cuba, and G.M. Weiss, *Creating Thematic Maps with OGC Standards Through the Web*, CPqD Telecom & IT Solutions, <http://www.gmldays.com/papers/Teixeira.html>.
- [5] M.A. Manso, A. Maldonado, R. Hernandez, D. Ballari, and J. Moya, *GEOSISMO : Visualization of Events and Seismologic Characteristics in the Internet*, Madrid Polytechnic University, http://redgeomatrica.rediris.es/ICA_Madrid2005/papers/manso.pdf.

Abson Sae-Tang
HEIG-VD IICT/geo.SYSIN

IICT/geo.SYSIN, <http://www.ict.ch>
musy-abson.sae-tang AT heig-vd.ch

Olivier Ertz
HEIG-VD *IICT/geo.SYSIN*

IICT/geo.SYSIN, <http://www.ict.ch>
olivier.ertz AT heig-vd.ch

Interopérabilité pour les données géospatiales 3D

Expérimentations avec CityGML et les services web OGC

par Hans Plum et Dr. Markus Lupp, traduit par Dr. Thierry Badard

Résumé

Le stockage, le traitement et la visualisation de données géospatiales 3D étaient déjà un sujet important dans le domaine des SIG même bien avant que le moteur de recherche numéro un introduise son globe virtuel. L'utilisation des recommandations de l'Open Geospatial Consortium (OGC) offre de nouvelles possibilités de combiner et utiliser des données géospatiales 3D. Les résultats obtenus à l'issue de premières expérimentations semblent prometteurs.

Introduction

Traiter et visualiser des données géospatiales 3D est devenu quelque chose de commun ces dernières années. Le nombre de solutions logicielles offertes témoigne de cet engouement ainsi que de l'intérêt pour le développement de CityGML. CityGML est un format d'échange basé sur GML, dédié à l'échange de modèles numériques tridimensionnels de ville et qui est déjà mis en oeuvre dans nombre de logiciels. Avec la définition de CityGML et l'implantation de services web OGC pour permettre l'accès et la visualisation de données géospatiales 3D, une convergence entre les domaines du traitement des données géospatiales 3D et des infrastructures de données spatiales s'opère.

Cet article traite de solutions s'appuyant sur la technologie issue du projet deegree. Ces solutions concernent "le stockage et l'administration de modèles tridimensionnels de ville pour Bonn, Berlin et Hambourg", "la visualisation de modèles numériques de terrain pour l'Agence Fédérale de Cartographie et de Géodésie d'Allemagne", "la réalisation d'un WFS CityGML transactionnel pour l'Open Geospatial Consortium" et seront présentées ici.

Les recommandations OGC pertinentes vis-à-vis de la 3D

Plusieurs spécifications et propositions de l'OGC sont d'importance pour la manipulation de données géospatiales 3D. On pense ici à CityGML en tant que modèle de données et format d'échange, aux WFS et WCS (Web Feature Service et Web Coverage Service) pour l'accès aux données et aux services WTS (Web Terrain Service) et W3DS (Web 3D Service) pour des fins de visualisation.

City Geography Markup Language (CityGML)

CityGML propose un modèle pour définir des objets 3D dans des zones urbaines. Il s'agit d'un schéma d'application GML, au sens où il permet la modélisation d'objets d'un domaine d'application en s'appuyant sur les structures mises à disposition par le Geography Markup Language. De ce fait, CityGML peut être considéré à la fois comme un modèle sémantique et comme un format d'échange.

CityGML fut jusqu'ici principalement développé par un groupe de travail issu de l'initiative sur les infrastructures de données spatiales de la Rhénanie-du-Nord-Westphalie, bien que les membres de ce groupe de travail proviennent de toute l'Allemagne. CityGML fut introduit à l'OGC dans sa version 0.3 et fut publié en tant que proposition (1). CityGML devint en un court laps de temps une recommandation officielle de l'OGC.

Web Feature Service

Un service d'accès aux entités géospatiales par le web ou Web Feature Service (WFS, (2)) permet d'interroger des données géospatiales encodées en GML. La spécification OGC pour l'encodage de filtres (Filter Encoding, (3)) est un langage XML permettant d'adresser des requêtes à un WFS d'une façon analogue à ce que le langage SQL permet. Un WFS qui rend possible non seulement la lecture des données mais aussi l'accès en écriture (création, mise à jour et suppression d'entités) est qualifié de WFS transactionnel (WFS-T).

WFS est une recommandation officielle de l'OGC et est actuellement dans sa version 1.1.0. Un WFS mettant en oeuvre cette version de la spécification se doit de supporter l'encodage des entités géospatiales en GML 3.1.1 – la même version qui sert de base à CityGML. Il est ainsi possible d'utiliser un WFS comme une couche d'accès à des données CityGML.

Web Coverage Service

Un service d'accès aux données géospatiales de couverture par le web ou Web Coverage Service (WCS, (4)) rend possible l'accès à toutes sortes de données en mode maillé, ex. des données raster ou des TINs. Des exemples de telles données sont celles produites par des processus de télédétection ou provenant de modèles numériques de terrain. Dans le contexte d'une infrastructure de données spatiales 3D, un WCS peut être utilisé pour accéder à des modèles de terrain. WCS est une recommandation officielle de l'OGC et est actuellement dans sa version 1.1.0.

Web Terrain Service

Un Web Terrain Service (WTS), dont les spécifications sont toujours à l'état d'une proposition soumise à discussion, génère des vues de scènes 3D. Au contraire d'un WMS qui crée des visualisations de données 2D, un WTS génère une image représentant des données 3D.

Malheureusement, le développement de la spécification WTS avance plutôt lentement. La version actuelle des spécifications préliminaires porte le nom de "Web Perspective View Service" (WPVS) pour exprimer le fait que ce service est en mesure de représenter des objets 3D en plus du "Terrain".

La figure 1 illustre le résultat produit par une requête GetView. Un modèle numérique de terrain est affiché et texturé à l'aide de photographies aériennes. Un certain nombre de bâtiments sont affichés par dessus le terrain (l'un d'entre eux de façon transparente).

WPVS crée des présentations d'objets 3D. L'opération la plus importante de ce service est GetView, qui retourne une image statique de paysages 3D. Cette opération peut donc être vue comme une extension de la méthode GetMap de WMS. Comparée avec cette dernière, GetView définit des paramètres additionnels permettant de spécifier la scène 3D. Parmi ceux-ci, on compte l'angle de rotation et l'azimut de la scène à afficher. L'image produite en retour par GetView est statique, il n'est pas possible

de naviguer directement au travers de la scène. Un client WPVS est donc, comparé avec de véritables visualisateurs 3D, assez peu interactif mais peut être mis en oeuvre sous la forme d'une application web reposant simplement sur DHTML et ayant nul besoin de recourir à des plugins. Un tel client web 3D en plus d'être simple présente l'avantage supplémentaire d'être facilement intégrable avec un autre logiciel client basé sur le web, comme par exemple les portails WMS.

Le défi lorsque l'on crée un client WPVS est de cacher la complexité d'une requête GetView derrière une interface graphique utilisateur intuitive et facile d'usage, qui permet la navigation dans des espaces 3D.

Cas d'utilisation

De façon à rendre possible les projets mentionnés en introduction, les cas d'utilisation suivants doivent être supportés.

Stockage de modèles numériques de ville

Les modèles numériques de ville sont souvent créés dans des systèmes de DAO et stockés dans des formats propres à ceux-ci. Cela pose un certain nombre de problèmes. Il n'est pas facilement possible de sélectionner des parties du modèle ou d'organiser les mises à jour. Pour cette raison, les organisations qui détiennent de tels modèles requièrent des données homogènes, pour lesquelles un stockage dans une base de donnée est la meilleure solution.

Pour mettre en oeuvre ce cas d'utilisation, il est alors nécessaire de stocker les documents CityGML dans une base de données – le plus probablement relationnelle. Pour accéder à la base de données, un WFS est un choix évident. De cette façon, les documents CityGML peuvent être directement insérés ou extraits de la base de données.

Pour restreindre l'accès au WFS, il est nécessaire d'utiliser un mécanisme de contrôle des accès. Dans les projets mentionnés dans cet article, des composants de deegree iGeoSecurity ont été utilisés à cette fin.

Visualisation Web

Les possibilités de visualisation offertes par les données géospatiales 3D constituent leur principal intérêt. Les domaines d'application sont le support à l'aménagement urbain ou la navigation. Dans le

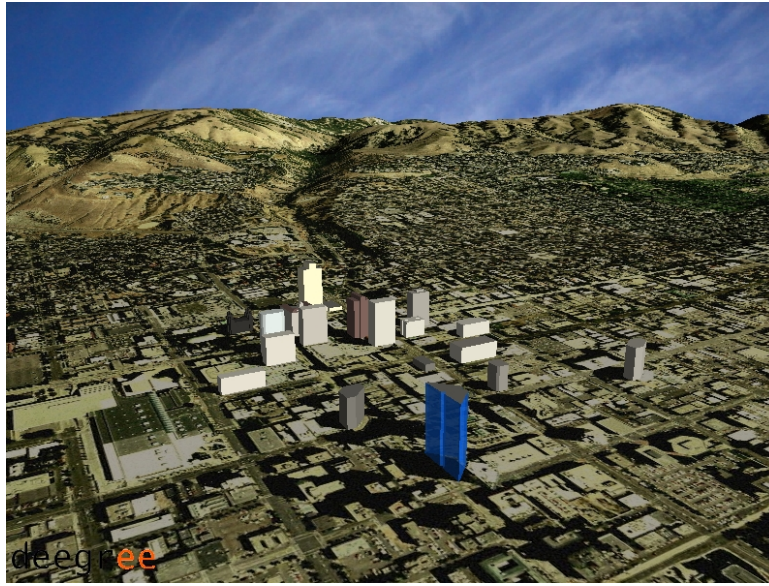


FIG. 1: Visualisation du terrain et des bâtiments en utilisant le WTS/WPVS de deegree

contexte des processus de planification, les géovisualisations 3D permettent de voir les impacts potentiels de certains projets avant qu'ils ne soient réalisés.

Les systèmes d'information à destination des touristes peuvent également tirer partie de la visualisation 3D. La reconnaissance de points de repère ou la navigation peuvent ainsi être améliorées. Le couplage de cartes classiques en 2D avec des visualisations de scène en 3D présente également un grand intérêt.

Pour des fins marketing concernant les données elles-mêmes, on peut voir sur Internet des modèles numériques de terrain et de ville. Le potentiel des données est ainsi démontré.

Architecture d'une infrastructure de données spatiales 3D

Dans les paragraphes suivants, l'architecture d'une infrastructure de données spatiales 3D s'appuyant sur deegree sera décrite (c.f. figure 2).

Les modèles de bâtiments sont stockés dans un système de gestion de bases de données spatiales, ex. PostGIS ou Oracle Spatial. Un WFS transactionnel (WFS-T) permet l'accès en lecture et en écriture aux modèles de ville. De façon à contrôler le flux d'informations, et particulièrement les transactions adressées au WFS, un proxy OWS (OGC Web Service) est utilisé pour prémunir les informations sur

le bâti contre les accès non autorisés. Le composant d'édition – principalement un système de DAO – accède aux données via le proxy OWS et le WFS.

Les données numériques d'élévation peuvent également être entreposées dans une base de données spatiales, tout particulièrement s'il s'agit de TINs ou de points. Les données rasters peuvent être quant à elles, sauveés sur le système de fichier. Afin de permettre une méthode d'accès rapide, les mécanismes décrits ci-dessus sont à privilégier. L'accès aux données numériques de terrain en format raster est fourni par un WCS. Tandis qu'un modèle de ville est représenté, le Web Coverage Service facilite l'accès aux modèles numériques de terrain correspondants.

Sur le côté droit de la figure 2, le processus de visualisation est illustré. Le WPVS de deegree accède aux données et les extrait de la base de données spatiales. En outre, il est possible d'intégrer des services WFS ou WCS externes. Les données de texture telles que des ortho-photos ou des cartes pour la navigation sont aussi requises. Via un WMS, ce type de données peut être intégré. Un client web WPVS fournit une interface graphique utilisateur qui peut être utilisée dans les navigateurs web.

Conclusion

Le développement de CityGML représente une étape importante pour la constitution d'infrastructure

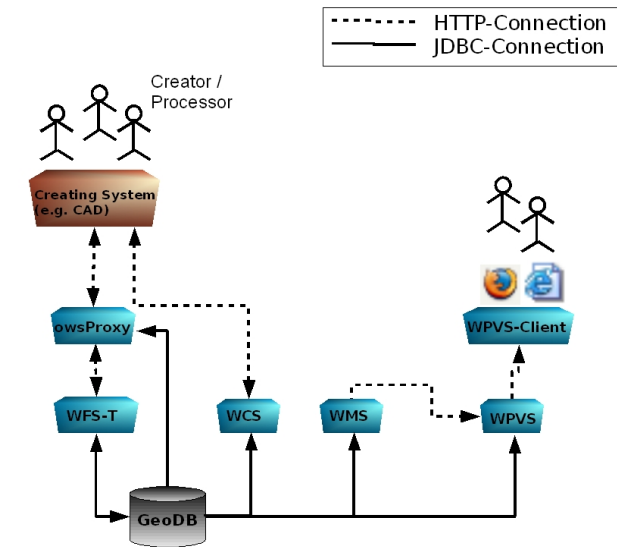


FIG. 2: Architecture d'une infrastructure de données spatiales 3D

tures de données spatiales 3D. Les expérimentations faites avec les composants du projet deegree dans de nombreux projets semblent prometteuses. Elles montrent qu'il est déjà possible de créer des infrastructures de données spatiales 3D avec des composants open source.

Les composants logiciels mentionnés sont disponibles à www.deegree.org. Au moment de l'écriture de cet article, la version RC1 des WPVS (incluant un client), WFS et WCS étaient disponibles sous la forme d'archives WAR facilement déployables.

06-057r1. https://portal.opengeospatial.org/files/?artifact_id=16675.

- [2] Vretanos, P.A. (2004) OpenGIS Web Feature Service Implementation Specification Version 1.1.0. OpenGIS Project Document 04-094. <http://www.opengis.org/specs/?page=specs>.
- [3] Vretanos, P.A. (2004) OpenGIS Filter Encoding Specification Version 1.1.0. OpenGIS Project Document 04-095. <http://www.opengis.org/specs/?page=specs>.
- [4] Evans, J. (2003) OpenGIS Web Coverage Service Specification Version 1.0.0. OpenGIS Project Document 03-065r6. <http://www.opengis.org/specs/?page=specs>.

Hans Plum, Dr. Markus Lupp
 lat/lon GmbH, Bonn, Germany
<http://www.lat-lon.de>
 plum|lupp AT lat-lon.de

Bibliographie

- [1] Gröger, G., T. Kolbe and A. Czerwinski (2006) City Geography Markup Language. OGC project document

Un Service Web orienté modèle pour une interopérabilité sémantique améliorée

Peter Staub, traduit par Benoit Véler

Introduction

Cet article traite de la recherche actuelle dans le domaine de l'interopérabilité des SIG. Nous portons notre intérêt sur l'hétérogénéité au niveau des modèles conceptuels de données. Le projet de Service web orienté modèle présenté ici, vise à améliorer l'interopérabilité sémantique. Les approches de l'interopérabilité des données telles que les services web OGC (OWS) sont combinées avec des méthodes d'interopérabilité. Une approche orientée modèle des modèles de conception de données, permet d'éloigner les traitements de données de tout système spécifique.

L'interopérabilité est une question cruciale à traiter, dans le contexte d'applications géospatiales et de communautés d'informations. L'usage de services web est bien établi, et utilisable de manière standardisée grâce aux efforts de l'OGC. Cependant, des OWS (?) tels que le Web Feature Service (WFS) (3) fournissent de l'interopérabilité au niveau des données, mais pas au niveau du modèle. La représentation des modèles conceptuels sont une précondition pour une interopérabilité sémantique, mais ne sont pas reconnues par OWS.

Parmi les initiatives européennes sur l'infrastructure des données géographiques – telles que INSPIRE¹¹ – le besoin d'interopérabilité augmente, non seulement au niveau des données, mais aussi des modèles de données. Le projet de recherche décrit dans cet article a été initié dans le contexte d'un projet dans la région du lac de Constance¹². Le projet mentionné vise à créer un Web-SIG transfrontalier pour des applications.

Dans ce projet de recherche, nous introduisons un **WFS orienté modèle (mdWFS)** qui combine à la fois les avantages de l'interopérabilité de données de l'OWS, et ceux de l'approche modèle utilisée dans les modèles conceptuels de données. De plus, le formalisme pour établir la représentation des modèles conceptuels est développé, et un prototype est mis en place. Grâce à cette combinaison, le mdWFS que nous introduisons est une approche qui fournit une

interopérabilité sémantique étendue.

Fondements de modèles de données et d'interopérabilité sémantique

L'approche orientée modèle

L'idée principale de l'approche orientée modèle est de décrire des modèles de données spatiales en utilisant un schéma de langage conceptuel (CSL). L'usage d'un CSL pour modéliser permet de conserver les structures de données indépendantes de tout système spécifique, ou de tout format de transfert, tel que XML ou GML. Potentiellement tous les formats de transferts peuvent être dérivés automatiquement du schéma conceptuel – à condition d'utiliser le compilateur approprié.

Si l'on souhaite obtenir une interopérabilité sémantique, il faudra créer une représentation des modèles conceptuels. La représentation d'un modèle conceptuel est convertie en fonctions de représentation \mathcal{F}_M d'un schéma source A vers un schéma cible B :

$$A \xrightarrow{\mathcal{F}_M} B$$

1. Description of the application domain with the chosen CSL \rightarrow *conceptual schema*, platform independent model (PIM)
2. Derivation of any format schema (e. g. a GML Application Schema) \rightarrow *logical and physical schema*, platform specific model (PSM)

L'approche orientée modèle consiste en quatre étapes (voir figure 2) :

1. Spécification d'un *domaine d'application* (par ex. "de quoi sommes-nous en train de parler")
2. Spécification d'un CSL avec un métamodèle UML cohérent
3. Description d'un domaine d'application avec le CSL choisi \rightarrow *schema conceptuel*, modèle indépendant d'une plateforme (platform independent model - PIM)

¹¹ site web du projet INSPIRE :<http://www.ec-gis.org/inspire/index.cfm>

¹²site web du Bodensee-Geodatenpool (Groupe de données spatiales du Lac de Constance) :<http://www.bodensee-geodatenpool.net>

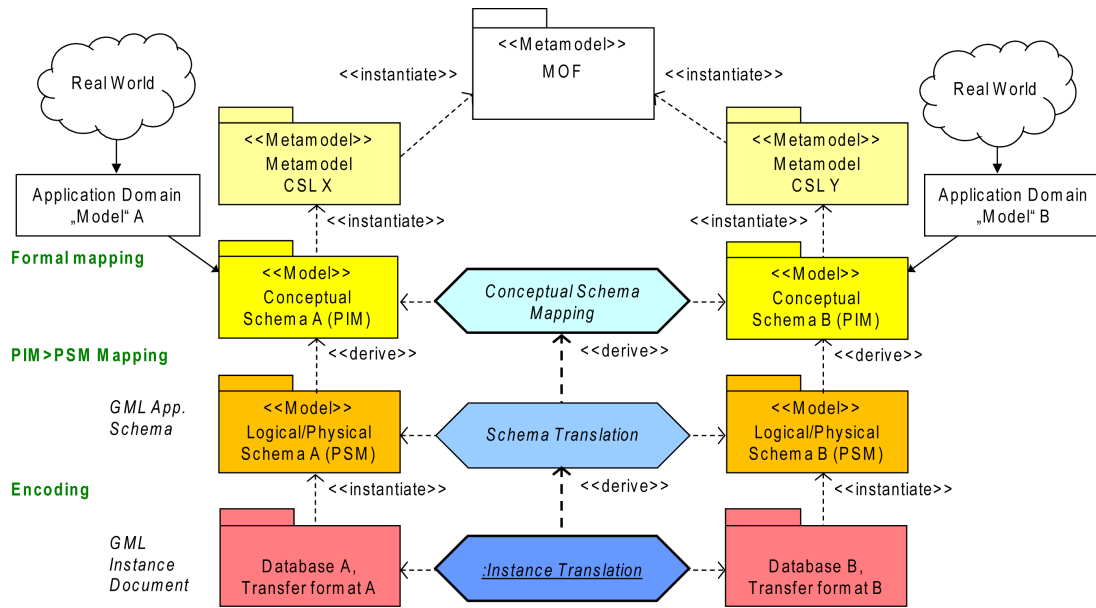


FIG. 1: Model-driven approach and semantic interoperability

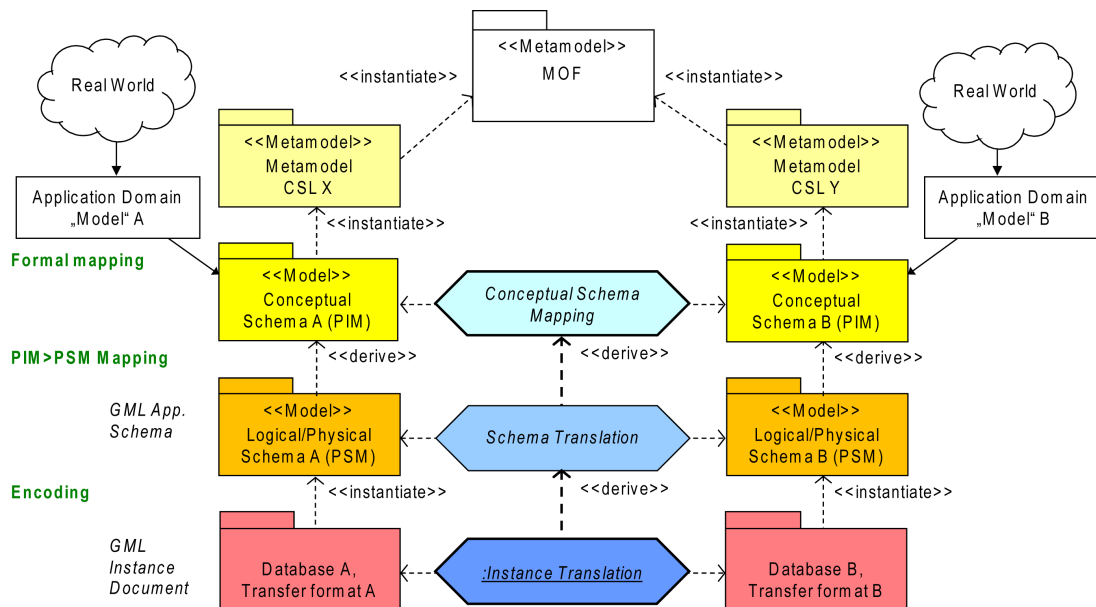


FIG. 2: Approche orientée modèle et interopérabilité sémantique

4. Derivation de schémas de tous formats (par ex. un Schema d'Application GML) → *schéma physique et logique*, modèle spécifique de plateforme (platform specific model - PSM)

Dans le projet de recherche présenté, le CSL (textuel) *Interlis* est appliqué à la modélisation des données. *Interlis* est un standard suisse (8) largement appliqué dans les applications cadastrales et d'aménagement. *Interlis* est basé sur un profil UML 2, et un compilateur¹³ produit des schémas XML (format *Interlis*), ou des Schémas d'Application GML pour tout modèle de données *Interlis*.

A partir d'un Modèle d'Architecture de données, spécifié par le Groupe de Gestion des Objets (Object Management Group - OMG), la production d'un schéma de format pour un schéma conceptuelle est dénommée cartographie PIM-PSM. En termes de direction de cartographie, le PIM-PSM est également appelé une cartographie "verticale", alors que les représentations de modèles pour l'interopérabilité sémantique est "horizontale" (par ex. PIM-PIM) cartes (voir prochaine section)

Interopérabilité Sémantique

Techniquement, il y a deux éléments principaux caractérisant l'interopérabilité : 1) *Interopérabilité des données* = la capacité d'un système ou d'un procédé d'échanger des bases de données en utilisant des formats de transferts définis. 2) *Interopérabilité des Modèles* = la capacité d'établir des modèles conceptuels de cartes dans le but d'exécuter des transformations sémantiques des données.

Pour atteindre une interopérabilité sémantique, différents modèles de données doivent être représentés. Un traducteur interprète alors les règles de représentation à partir de la carte du modèle conceptuel et exécute automatiquement la séquence de traductions.

Les approches de transformation sémantique peuvent être classifiées comme suit (1) :

- *Niveau d'abstraction* : Les transformations sémantiques peuvent être réalisées à différents niveaux d'abstractions, (niveau conceptuel, niveau logique, et niveau physique (par ex. le format de transfert). Une transformation sémantique au niveau conceptuel est indépendante de la plateforme, alors que les approches aux niveaux logiques et physiques sont spécifiques à la plateforme.

- *Orientation* : Transformation sémantique horizontale entre différents schémas au même niveau d'abstraction (PIM-PIM; PSM-PSM) contre. Transformation sémantique verticale entre différents niveaux d'abstraction (PIM-PSM).
- *Niveau d'automatisation* : Créer des règles de représentation à la main contre. créer un schéma automatique de correspondance, qui est partiellement utilisable.

Résumés des approches existantes

Une possibilité est d'intégrer toutes les données dans un système centralisé. C'est très coûteux, et exige un niveau de connaissance expert. Pour intégrer les données dans le système central, 1 :1-des conversions de formats doivent être menées à bien. C'est souvent au prix de pertes, dans la mesure où un format de données qui est différent de celui d'origine, est en général incapable de reproduire toute la sémantique contenue dans la format de départ. De plus, l'inévitable redondance du stockage de données peut aboutir à la création de données périmées.

Des OWS existants, tels que WFS, ont des raccourcis concernant les échanges sémantiques. OWS permet l'interopérabilité syntaxique (par ex. pour les données), mais pas sémantique (par ex. correspondance des modèles). Les modèles conceptuels des sources, sont cachés des systèmes cibles, et les transformations sémantiques ne sont pas supportées. Par conséquent, le WFS manque de capacités à manipuler des informations sur les modèles conceptuels en plus d'informations sur les données.

Concept de WFS orienté modèle

Préconditions pour une transformation sémantique en ligne

Si l'on souhaite avoir un service web qui permette l'interopérabilité de données *et* qui soit capable de stocker et fournir des schémas conceptuels, un certain nombre de pré-conditions doit être respecté. Il doit être validé que les schémas conceptuels soient décrits (c. à.d. modélisés) en utilisant un CSL en format texte, avec une représentation graphique en UML 2 (et le format d'échange correspondant XMI). De plus, un langage en amont est encore nécessaire pour représenter le schéma de règles de cartographie, au niveau d'abstraction nécessaire. Finalement,

¹³compilateur *Interlis* : voir <http://www.interlis.ch>. Le compilateur est gratuit et opensource.

nous utilisons une interface WFS standard pour fournir une interopérabilité des données satisfaisante.

Besoins de l'application Web

Réaliser des transformations sémantiques sur le web signifie dans notre cas imaginer une application web. Cette application doit vérifier deux besoins particuliers :

1. Permettre l'accès à des données spatiales, depuis le schéma conceptuel des données d'origine, vers tout schéma défini pour le modèle cible.
2. Interopérabilité avec l'OWS existant.

L'interface mdWFS

Nous avons imaginé un service dénommé "model-driven Web Feature Service" (mdWFS), prenant en compte ces besoins. Le mdWFS a en charge le stockage et le rendu des schémas conceptuels, la conduite des transformations sémantiques (cartographies PIM-PIM) en interprétant les modèles de transformation. Après une transformation sémantique, le mdWFS configure un WFS standard, pour proposer un service pour les échanges de données. Le WFS standard est configuré d'après le modèle cible, mais délivre des données modifiées du modèle source.

Extensions du protocole WFS

- To provide a service protocol for the mdWFS, a new request parameter SERVICE=mdWFS is implemented.
- The GetCapabilities request is extended to provide a SchemaList. This list includes each conceptual schema that is available in the service.
- The DescribeFeatureType request is extended to provide the XMI format for transferring model information.
- Finally, a whole new request DoTransform is introduced. This request transfers the conceptual mapping schema to the mdWFS and invokes the semantic transformation.

Dans le but de créer un WFS capable de stocker et de proposer des schémas conceptuels, les spécifications de l'OGC doivent être étendues. Dans les spécifications mdWFS, les extensions appliquées sont les suivantes (1) :

- Pour fournir un protocole, un nouveau paramètre de requête est mis en place SERVICE=mdWFS.
- La requête GetCapabilities est étendue pour obtenir un SchemaList. Cette liste contient chaque schéma présent dans le service.
- La requête DescribeFeatureType est étendue pour fournir le format XMI pour transférer les informations sur les modèles.
- Finalement, une requête complètement nouvelle est introduite DoTransform. Celle-ci sert à transférer les représentation des schémas conceptuels vers le mdWFS et met en route la transformation sémantique.

UMLT, un Langage de Représentation de Schemas Conceptuels de Cartographie

Concept d'UMLT

Nous introduisons un langage conceptuel de représentation, qui peut-être utilisé pour créer des schémas de transformation sémantique. Ce langage doit vérifier plusieurs critères pour être utilisé. Les schémas de transformation doivent être compréhensible en dehors de scientifiques de l'informatique. Par conséquent, un métamodèle UML, ainsi qu'une syntaxe pour de notations textuelles utilisables par l'homme (HUTN) sont développés. Les schémas de transformations sont représentés respectivement de manière visuelle (diagrammes d'activité UML), textuelle, (dérivé d'Interlis CSL) et en XML (par ex. XMI). Les standards habituels dans le domaine de la modélisation de données sont pris en compte¹⁴. Nous plaçons également une couche d'abstraction pour les types des données spatiales.

Deux approches venant de l'OMG ont été examinées. Premièrement, le formalisme 'Meta Object Facility Query/Views/Transformations (MOF-QVT)' (5) : ce formalisme est conçu pour la transformation de métamodèles, par ex. UML→Java. Les modèles MOF-QVT sont difficiles à comprendre, et leur représentation visuelle aide peu. Le standard est complexe, dans la mesure, où il contient finalement trois langages : relations, opérations et racine. De plus, le standard MOF-QVT est surtout appliqué pour les représentations en PIM-PSM.

Une autre approche examinée est UML2 Activity. Un diagramme UML 2 activity peut être utilisé pour

¹⁴standard tels que ceux de l'OMG, de l'OGC et ISO

décrire des transformations en termes de séquences d'activités. Une description claire de la sémantique et des formats de transferts (XMI 2.1) est fournie dans la superstructure des modèles UML. Les modèles UML 2 sont compréhensibles et un certain nombre d'installations et d'APIs open source sont disponibles.

A cause des considérations ci-dessous, notre langage conceptuel de cartographie est basé sur une extension indépendante du métamodèle UML 2. Pour spécifier les éléments du langage, un modèle UML 2 est créé et la notation textuelle du langage est définie par un ensemble de règles grammaticales EBNF. A ce stade du projet, nous baptisons notre langage conceptuel de représentation "UMLT".

Eléments du langage UMLT

Les éléments du langage UMLT sont un héritage de UML 2 Activities (7). Nous introduisons les éléments suivants (see figure 3) :

- StructuredTransformation
- SelectionCriteria : selection des données renseignées avec une expression logique.
- VirtualAssociation : gestion d'objets renseignés n'étant pas associés avec des objets association. CEs objets importés peuvent avoir des attributs ou des clés étrangères qui sont évaluées en temps réel, de manière à obtenir des relations calculées¹⁵. Durant une transformation sémantique, de tels objets peuvent être associés *de manière virtuelle* si nécessaire. La VirtualAssociation est introduite (au contraire d'une association dérivée usuelle) pour fournir un moyen de spécifier explicitement la propriété de jointure de l'association avec l'expression joinCriteria.
- TransformationAction : héritage d'une opaqueAction UML, fournissant un élément d'activité qui ne peut pas être plus structuré. C'est une action élémentaire de transformation.
- AssignmentDefinitions : identifie les expressions et les type primitifs en tant que spécifications.
- MappingRule : L'objet cartographique. Construit comme une composition de définitions de règles.
- AssociationBinding : En sélectionnant des objets associés, on peut définir comment ces associations sont évaluées durant l'input.

- JoinType : Un type 'énumération', pour spécifier le type de jointure dans le lien d'association.

Prototype Implementation

Installation du prototype

Dans le contexte du projet de recherche présenté, un prototype de vérification est installé. En dehors de l'extension du protocole WFS, et des spécifications du langage UMLT, ce prototype est caractérisé par un 'model parser', un éditeur de cartographie de modèles, et un prototype 'test bed'. Le model parser et l'éditeur sont développés dans l'environnement Eclipse¹⁶. Le model parser crée un fichier XML à partir d'un modèle de données et d'un modèle de cartographie UMLT.

Dans le prototype de test bed, nous utilisons une base de données spatiale Oracle et une installation degree WFS¹⁷, sur laquelle est rajoutée mdWFS. La figure 4 montre les étapes d'une transformation sémantique utilisant mdWFS.

Nous avons au départ concentré nos efforts sur l'extension WFS and sur le langage conceptuel de cartographie UMLT. Nous considérons Oracle comme un RDBMS très adapté pour nos besoins, grâce à des fonctions spatiales puissantes. Par conséquent, nous utilisons le RDBMS que nous avons déjà en main, bien que ce ne soit pas une solution FOSS. Surtout, un mfWFS peut être appliqué à tout RDBMS avec un extension spatiale.

Avant de pouvoir commence à utiliser mdWFS, il faut configurer la base de données selon la source du modèle de données A. Cela peut être fait en utilisant un outil FOSS existant, dénommé "ili2ora"¹⁸. Cet outil permet de configurer une base de données ORACLE Spatial, selon un modèle de données UML/Interlis, et d'importer des éléments dans la base de données.

1. Le client B envoie une requête au catalogue de modèles de mdWFS
2. Le mdWFS fournit un catalogue des modèles disponibles
3. Le client B choisit un modèle source (par ex. le modèle A) et ordonne les informations sur le modèle

¹⁵Un exemple different est une relation geometrie/topologie : un bâtiment *sur* une parcelle

¹⁶Eclipse : <http://www.eclipse.org>

¹⁷page du projet deegree : <http://www.deegree.org>

¹⁸Source : <http://www.eisenhutinformatik.ch/interlis/ili2ora/>

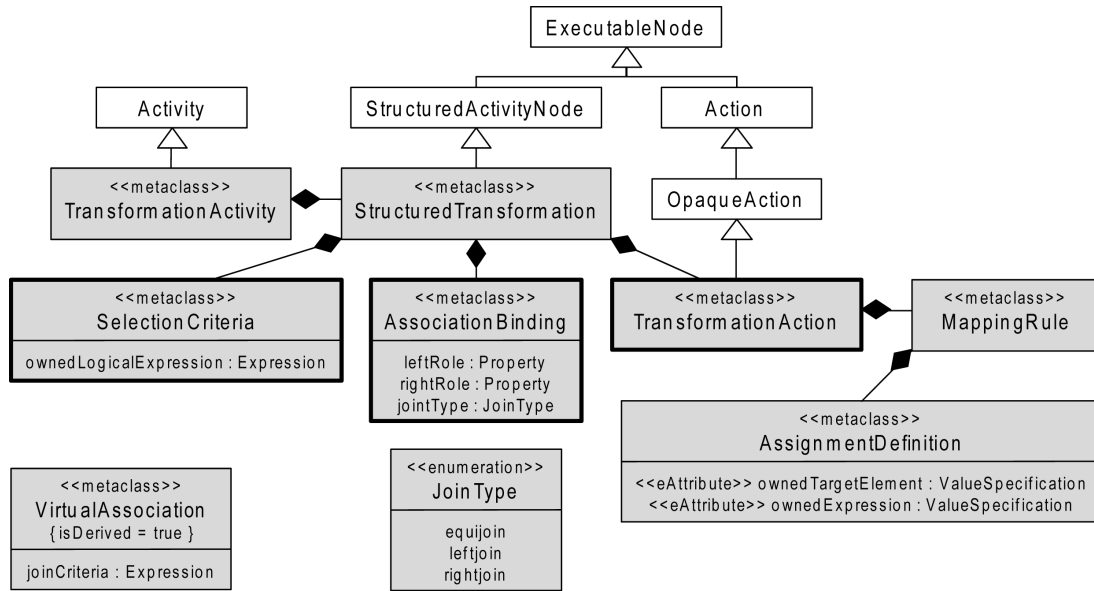


FIG. 3: Eléments du langage UMLT

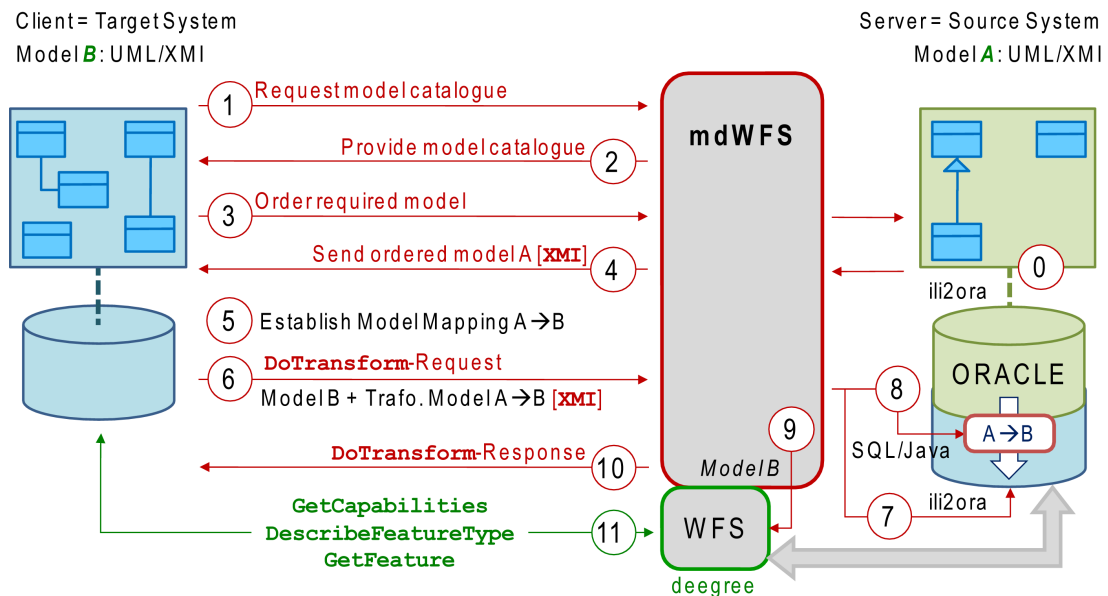


FIG. 4: Test d'installation du prototype

4. Le mdWFS pointe les informations A et envoie le modèle (XMI) ou un modèle référence au client B
5. Le client B crée le modèle $M : A \xrightarrow{\mathcal{F}_M} B$ en spécifiant les règles de représentation UMLT adéquates
6. Le modèle de transformation et le modèle cible B sont traités et traduits en XMI et envoyés au mdWFS par une requête `DoTransform`
7. Selon le modèle cible B , le mdWFS configure une base de données spatiale ORACLE utilisant à nouveau `ili2ora`
8. Les règles de représentation du modèle de transformation sont traduites en séquences SQL et en instructions Java, dans le but de transformer des données depuis le modèle source/serveur A vers le modèle cible/client B
9. Le mdWFS configure un WFS standard (degré) d'après le modèle cible B . Ce WFS est connecté à la base de données contenant la donnée transformée.
10. A la fin de la transformation, le mdWFS envoie une réponse `DoTransform` au client B
11. Le client B accède aux données transformées depuis le modèle/base de donnée A , transformé en modèle B , via des requêtes standard WFS.

Conclusions

Any given application domain can be characterised by different data structures. This leads to different data models. Therefore, conceptual model mappings must be established in order to achieve semantic interoperability.

L'évolution actuelle des systèmes d'IG montre qu'un langage conceptuel est en général utilisé pour les modèles de données. C'est une précondition nécessaire pour des transformations sémantiques au niveau conceptuel. Un domaine d'application peut être caractérisé par plusieurs structures de données. Cela conduit à différents modèles de données. Par conséquent, des modèles de représentations conceptuelle doivent être établis pour réaliser une interopérabilité sémantique.

Le nouveau *mdWFS* présenté dans cet article met en place la méthodologie de la transformation sémantique à un niveau conceptuel d'abstraction, qui permet une interopérabilité sémantique beaucoup plus étendue. Potentiellement, le mdWFS peut être intégré dans d'autres infrastructures basés OWS, grâce aux bases des standards de l'IG qui y sont ap-

pliquées.

Remerciements

Je voudrais remercier mes collègues au ETH Zurich et au TU Munich pour leur accompagnement et leur collaboration positive dans ce projet de recherche : Dr.-Ing. A. Donaubaue, H. R. Gnägi, A. Morf, F. Straub. J'ai également apprécié l'excellent conduite de projet des Prof. Dr. A. Carosio, ETH Zurich et Prof. Dr.-Ing. M. Schilcher, TU Munich.

Ce projet est co-financé par le BKG, Bureau Fédéral allemand de Cartographie et de Géodésie, et Swisstopo, le Bureau Fédéral Suisse de Topographie.

Bibliographie

- [1] A. Donaubaue, F. Straub, M. Schilcher (2007) *mdWFS : A Concept of Web-enabling Semantic Transformation*. Proceedings of the 10th AGILE Conference on Geographic Information Science, 2007, Aalborg.
- [2] H. R. Gnägi, A. Morf, P. Staub (2006) *Semantic Interoperability through the Definition of Conceptual Model Transformations*. Proceedings of the 9th AGILE Conference on Geographic Information Science, 2006, Visegrád.
- [3] OGC Open Geospatial Consortium (2005) *Web Feature Service Implementation Specification : 1.1.0*. OpenGIS implementation specification OGC 04-094.
- [4] OMG Object Management Group (2003) *MDA Guide Version 1.0.1*. OMG specification omg/2003-06-01.
- [5] OMG Object Management Group (2005) *MOF 2.0 Query/Views/Transformations Specification*. OMG specification ptc/05-11-01.
- [6] OMG Object Management Group (2005) *MOF 2.0/XMI Mapping Specification, v2.1*. OMG specification formal/05-09-01.
- [7] OMG Object Management Group (2007) *UML Unified Modeling Language : Superstructure, version 2.1.1*. OMG specification formal/2007-02-05.

[8] SNV Swiss Association for Standardization (2006) INTERLIS Reference Manual, version 2.3. Swiss standard SN 612031.

Peter Staub

*ETH Zurich, Institute of Geodesy and Photogrammetry,
GIS Group*

<http://www.gis.ethz.ch>

peterstaub AT ethz.ch

Spatial-Yap : un système de base de données spatiales déductive

David Vaz et Michel Ferreira, traduit par Sidone Christophe

Introduction

Le paradigme des bases de données déductives est une extension de celui des bases de données traditionnelles avec des possibilités de déduction. La connaissance n'est pas seulement représentée par extension¹⁹, mais également par des règles logiques intentionnelles. Une approche classique dans la construction d'un tel système de base de données déductif est de coupler un système de programmation logique avec un système de base de données relationnelle.

Le langage de requête Datalog (1), réduit les données attributs à un champ de valeurs atomiques, tel que nombres et chaînes de caractères, lesquels sont les données typiques stockées dans une base de données. Cependant, les bases de données actuelles gèrent bien plus de structures de données, telles que les attributs géométriques des relations spatiales.

Dans cet article, nous décrivons l'extension du compilateur Yap Prolog (2) un système de programmation gratuit, open-source, pour qu'il gère la donnée spatiale, fournissant une solution dernier cri pour la modéliser, la requêter et l'extraire. Une proposition pour étendre Datalog vers Spatial Datalog a été décrite dans la littérature, dans le cadre des bases de données contraintes (3).

L'approche suivie dans Spatial Yap est différente et plus proche de la communauté des bases de données spatiales, car elle est basée sur des notions spatiales, plutôt que des inégalités polynomiales. Spatial-Yap peut construire des relations spatiales logiques entre données vectorielles et procure un environnement de programmation déclaratif pour sa gestion, intégrant, par exemple, des spécifications de récursivité, inhérente aux relations topologiques, et par un puissant ADT, comme outil logique. Bien que Spatial-Yap est bien plus orienté sur le déclaratif que sur l'efficacité, le système est capable d'explorer les fonctions avancées de Yap, comme un traducteur Prolog vers SQL et un moteur basé sur des essais, pour en améliorer les performances.

Programmation Logique et Programmation Logique Inductive

La Programmation Logique (LP), dont Prolog est le langage canonique, tente d'implémenter l'idée de Colmerauer et Kowalski selon laquelle le calcul est contrôlé par inférence (4). La motivation du paradigme de la LP est de séparer *ce que* doit faire un programme du *comment* il doit le faire. Cela a été résumé par la devise de Kowalski :

algorithme = logique + contrôle.

Les programmes Prolog utilisent la logique pour exprimer un problème et s'appuient sur le système Prolog pour exécuter cette spécification. Prolog implémente un sous-ensemble logique de premier ordre comme les clauses de Horn.

$A :- B_1, B_2, \dots, B_n.$

ce qui veut dire : A est Vrai si B1 est Vrai et B2 est Vrai ... et Bn est Vrai. Ces règles donnent une interprétation procédurale qui se lit ainsi :

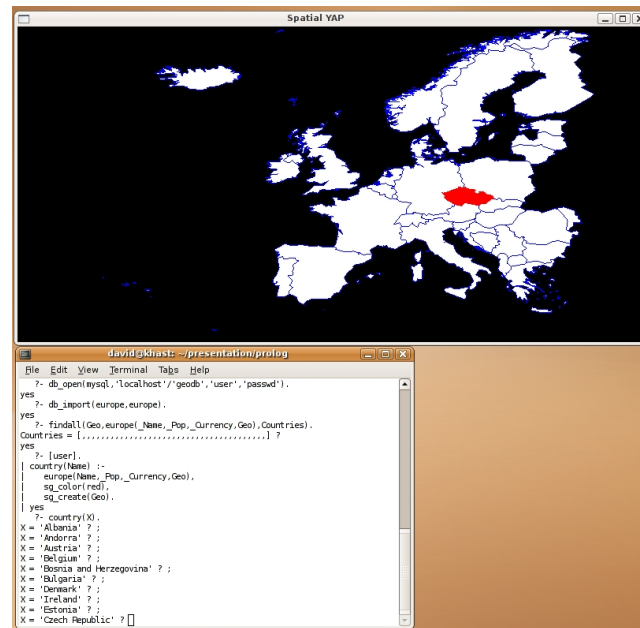
pour résoudre(exécuter) A résoudre(exécuter) B1 et résoudre(exécuter) B2 ... et résoudre(exécuter) Bn.

L'interprétation procédurale précise utilisée dans les programmes Prolog est une forme réduite de la résolution SLD (5).

Les systèmes de bases de données déductifs sont des systèmes de gestion de bases de données qui sont aussi conçus autour d'un modèle logique de données et dont le langage de requêtes est une version *set-oriented* de Prolog, connu sous le nom de Datalog. Les relations sont naturellement pensées comme la *valeur* du prédicat logique et la puissance expressive des expressions logiques est utilisée pour requêter de telles relations. La part déductive de tels systèmes vient du fait que les moteurs de Programmation Logique utilisent des *intentions* (ou compréhensions), pour exprimer des propriétés, et sont capables de matérialiser ces intentions dans des connaissances *extensionnelles* (tuples relationnels ou faits).

Ce processus de déduction, qui va d'intentions en extensions, est d'un point de vue calculatoire plus simple que le processus inverse qui consiste à aller d'extensions en intentions. Cependant, être capable de dériver une représentation intentionnel depuis des données extensionnelles, en inférant une

¹⁹Définition des termes : http://en.wikipedia.org/wiki/Extensional_definition



règle générale à partir d'exemples, est aussi crucial. C'est le but du paradigme de la Programmation Logique en tant que Programmation Logique Inductive (PLI) (6).

La Programmation Logique Inductive (PLI) est un domaine de recherche à l'intersection de l'Apprentissage Automatique et de la Programmation Logique. Les systèmes de PLI développent des prédicats à partir d'exemples et d'une base de connaissances, dont on déduit un programme logique qui confirme les exemples positifs et infirme les exemples négatifs. Pour déduire une théorie avec les propriétés attendues, beaucoup de systèmes de PLI suivent une sorte d'approche *generation-et-test* pour balayer *l'ensemble des hypothèses*. Une caractéristique importante de l'approche de la PLI pour la fouille de données est d'être multi-relationnelle, capable de formuler des théories qui impliquent des données ayant plusieurs relations, alors que beaucoup de techniques importantes de fouilles de données sont seulement capables de dégager des structures pour une unique relation. Ceci est particulièrement intéressant pour la fouille de données spatiales, qui est multi-relationnel (ou multi-couches) par définition.

Un système de base de données spatiales déductive

Spatial Yap résulte d'une interface complexe entre plusieurs composants. Les deux composants principaux sont le système Yap Prolog et le SGBD MySQL, qui sont couplés par une interface MYDDAS (7) (pour Mysql/Yap Deductive Database System). Cette interface est responsable du couplage des deux systèmes comme le montre la Figure 2. MYDDAS traduit les requêtes logiques en requêtes SQL de façon transparente, implémente la conversion des attributs MySQL en notions Yap et explore le moteur YapTab pour résoudre des requêtes récursives impliquant la base de données. Le niveau de sophistication de cette interface est très élevé : la recherche de tuples relationnels est directement implémentée dans le choix des points WAM, supportant les opérateurs simplifiés (8).

Afin de construire la base de données spatiales déductive, nous avons étendu l'interface MYDDAS afin qu'elle intègre des types géométriques MySQL. Deux composants supplémentaires sont fondamentaux pour construire Spatial Yap : une librairie d'opérateurs spatiaux basée sur la librairie classique GEOS et un composant de visualisation.

Quand on manipule des données spatiales il est essentiel d'être capable de représenter ces données graphiquement. Représenter une carte comme un en-

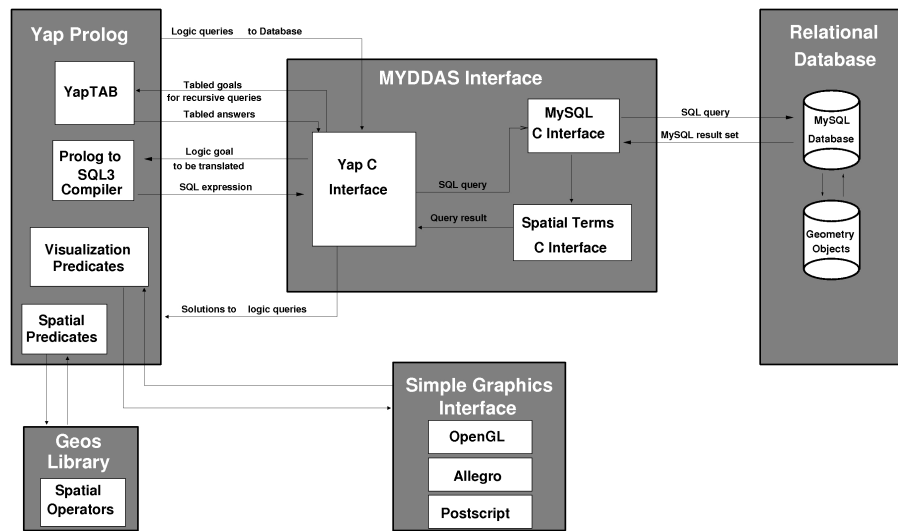


FIG. 2: Modèle Spatial Yap.

semble de notions Prolog est visuellement inacceptable du point de vue d'un utilisateur de SIG. Plus importante encore est la représentation d'une opération spatiale, comme l'intersection de deux polygones d'un point de vue graphique. Les analyses spatiales réalisées par l'utilisateur et la représentation des ensembles de résultats des requêtes spatiales nécessitent l'ajout d'un composant de visualisation à tout système de base de données spatiale. Nous aurions pu utiliser un des FOSS existants, comme MapServer, mais nous avons besoin de quelque chose de plus simple qui fonctionne à un haut-niveau graphique, étroitement lié avec le haut-niveau textuel de notre système Prolog. L'interaction entre ces deux fonctionnements haut-niveau était notre objectif principal, plutôt qu'un affichage graphique sophistiqué. La Figure ?? montre une copie d'écran de l'interaction entre les deux parties haut-niveaux de Spatial-Yap.

Applications en cours

Dans cette section nous décrivons deux projets utilisant Spatial-Yap en cours.

Étude du comportement du trafic dans la ville de Porto

Le but de ce projet est d'étudier le comportement du trafic dans la ville de Porto, deuxième plus important du Portugal, avec un réseau routier de 965 km, présenté Fig. 3. Nous nous intéres-

sons à la compréhension de facteurs affectant le trafic, pas uniquement concernant le temps et le jour, mais aussi incluant des entités géographiques intrinsèques, comme la présence d'écoles sur un tronçon de route et leur influence sur la congestion du trafic suivant certaines plages horaires. Un des objectifs plus ambitieux est la dérivation de la couche de signalisation routière, incluant les lumières et la localisation des stops et basée sur des modèles de mobilité mais aussi l'inférence des possibles destinations des conducteurs pouvant automatiquement activer les systèmes de navigation basés sur un fond des routes usuels.



FIG. 3: Road network of Porto.

Des algorithmes de routage et l'affichage des itinéraires calculés sont aussi implémentés en utilisant Spatial-Yap et son moteur. Les systèmes de PLI fournissent le support d'inférence sur les données géospatiales, comme les logs GPS.

Correction de la Classification Automatique des Forêts basée sur l'Analyse Spatiale

Un autre projet intéressant où nous utilisons Spatial-Yap vise à la surveillance globale des changements de biodiversité (9). Les gouvernements ont comme objectif ambitieux de réduire la baisse de la biodiversité d'ici 2010, et les scientifiques sont confrontés maintenant au défi de l'accès aux progrès réalisés dans ce sens. Le projet européen Corine Land Cover²⁰ (CLC) fournit des données pour deux années différentes (1990 et 2000), utilisant 44 classes d'occupation du sol. Ces données sont acquises à partir d'images satellites, grâce à la vectorisation de chacune des 44 classes d'occupation du sol en polygones, réalisée automatiquement, par reconnaissance de couleurs. Malheureusement les classes CLC ne sont pas les plus appropriées pour surveiller la biodiversité. Par exemple, la base CLC courante n'a que 3 classes pour la forêt (feuillus, conifères et mixtes). Par conséquent, une augmentation observée dans la zone forestière de feuillus pourrait être due à une augmentation de la superficie d'une plantation d'espèces exotiques, telles que l'Eucalyptus globulus, ou à une augmentation du nombre d'arbres dans les forêts natives de feuillus, les deux phénomènes ayant des implications différentes en terme de biodiversité. Le groupe de biologistes avec qui nous travaillons possède des cartes régionales de la région de l'Alto Minho au Portugal (see Fig. 4), aussi couverte par CLC. Ces cartes régionales sont réalisées sur place avec des techniques de cartographie lentes et coûteuses et l'identification sur place des espèces en forêt permet un meilleur niveau de détail dans la liste des classes. Notre projet essaie d'utiliser ces cartes régionales afin de dériver un ensemble de règles spatiales logiques qui permettent la caractérisation détaillée des données CLC pour la surveillance de la biodiversité.

²⁰Le Projet Européen Corine Land Cover (CLC) : <http://terrestrial.eionet.eu.int/CLC2000>

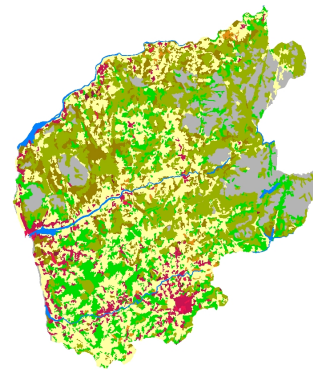


FIG. 4: Alto Minho Map.

Perspectives

La migration de MySQL à PostgreSQL est en cours. Notre but est d'étendre Prolog comme compilateur SQL, afin d'être capable de transférer dans le système de base de données la convergence des buts logiques, ce qui peut être préférable à l'indexation spatiale qui n'est pas actuellement disponible dans Spatial-Yap.

Spatial-Yap peut être téléchargé depuis myddas.dcc.fc.up.pt. Un manuel utilisateur et plusieurs articles avec des présentations plus approfondies de Spatial-Yap sont aussi disponibles sur cette même page.

Remerciements

Ce travail a été partiellement soutenu par MYDDAS (POSC/EIA/59154/2004) et par des fonds accordés à LIACC à travers le Programa de Financiamento Plurianual, Fundação para a Ciência e Tecnologia and Programa POSC. David Vaz est subventionné par une bourse de doctorat de type FCT PhD

grant SFRH/BD/29648/2006.

Bibliographie

- [1] J. D. Ullman. *Principles of Database and Knowledge-Base Systems*. Computer Science Press, 1989.
- [2] V. Santos Costa, L. Damas, R. Reis, and R. Azevedo. *YAP User's Manual*. Available from <http://www.ncc.up.pt/~vsc/Yap+>.
- [3] P. C. Kanellakis, G. M. Kuper, and P. Z. Revesz. Constraint query languages. *J. Comput. Syst. Sci.*, 51(1) :26–52, 1995.
- [4] R. Kowalski. Predicate Logic as a Programming Language. In *Information Processing*, pages 569–574. North-Holland, 1974.
- [5] J. W. Lloyd. *Foundations of Logic Programming*. Springer-Verlag, 1987.
- [6] S. Muggleton. Inductive Logic Programming. In *Conference on Algorithmic Learning Theory*, pages 43–62. Ohmsma, 1990.
- [7] T. Soares, M. Ferreira, and R. Rocha. The MYDDAS Programmer's Manual. Technical Report DCC-2005-10, Department of Computer Science, University of Porto, 2005.
- [8] T. Soares, R. Rocha, and M. Ferreira. Generic Cut Actions for External Prolog Predicates. In P. V. Hentenryck, editor, *Proceedings of the 8th International Symposium on Practical Aspects of Declarative Languages, PADL'2006*, number 3819 in LNCS, pages 16–30, Charleston, South Carolina, USA, January 2006. Springer-Verlag.
- [9] H. M. Pereira and H. D. Cooper. Towards the global monitoring of biodiversity change. *Trends in Ecology & Evolution*, 21(3) :123–129, March 2006.

David Vaz
DCC-FC & LIACC, University of Porto
[davidvaz AT dcc.fc.up.pt](mailto:davidvaz@dcc.fc.up.pt)

Michel Ferreira
DCC-FC & LIACC, University of Porto
[michel AT dcc.fc.up.pt](mailto:michel@dcc.fc.up.pt)

Études de cas

Le Projet DIVERT : Développement de Télématiques Inter-Véhicules Fiabiles

Hugo Conceição, Luís Damas, Michel Ferreira et João Barros, traduit par Marie Silvestre

Introduction

L'avènement de réseaux ad-hoc de communication sans fil entre voitures (car-to-car, C2C), à savoir des groupes de véhicules géo-localisés équipés de manière à pouvoir communiquer entre eux et s'organiser de façon autonome en un réseau collaboratif, laisse entrevoir une myriade de possibilités de partage et d'exploitation d'informations géographiques hautement dynamiques. Le manifeste du Consortium pour la Communication 'CAR 2 CAR' (1) décrit plusieurs scénarios dans lesquels ces réseaux pourraient être utilisés par exemple pour améliorer la sécurité routière (2), pour optimiser l'efficacité du trafic ou pour fournir des informations ou des divertissements au conducteur. Les systèmes *collaboratifs d'avertissement de risque de collision en empilement lorsque les voitures se suivent en file* (voir par exemple (3)), *de détection et d'avertissement de risque d'accident ou de notification de la localisation d'un danger* sont des exemples de telles applications de sécurité routière.

Concernant l'efficacité du trafic, les applications incluent *l'amélioration de la navigation et du guidage,*

où les véhicules équipés utilisent les informations concernant les conditions de trafic collectées auprès des infrastructures routières ou des autres véhicules, pour déterminer l'itinéraire optimal. Une autre application est le *conseil sur la vitesse optimale pour le passage aux feux verts*, où un feu de signalisation est capable de transmettre aux véhicules la vitesse optimale pour adoucir leur conduite et éviter de s'arrêter. De la même manière, la communication sans fil entre des véhicules proches peut fournir une *assistance C2C d'introduction* permettant aux voitures de rejoindre le flux de trafic sans l'interrompre.

Les réseaux ad-hoc de communication C2C sans fil vont également permettre des applications qui ne sont pas directement liées à la sécurité ou à l'efficacité du trafic, telles que la *notification de points d'intérêt* diffusés aux véhicules par les entreprises locales ou les attractions touristiques ou telles que *les diagnostics à distance* des véhicules. Une autre application concerne *l'accès à Internet dans les véhicules* par l'intermédiaire du réseau C2C, permettant de calculer des itinéraires passant par des points d'accès à Internet.

Le déploiement d'un réseau C2C et toutes ses applications soulèvent des défis intéressants. Il est évident qu'une technologie basée sur la communication entre des véhicules nécessite une distribution suffisante sur le marché pour avoir un effet. Le

Consortium pour la Communication C2C a estimé que le taux de pénétration requis est de 5% pour permettre la propagation des informations sur le trafic. Un lancement timide pourrait bloquer de nouveaux clients potentiels quant à l'équipement de leur véhicule avec un tel système de communication. L'adaptabilité du système de communication C2C est un autre point important qui doit être étudié. Le système doit fonctionner dans des situations où le trafic routier est de très faible densité aussi bien que lorsqu'il est de très grande densité, ce qui entraîne différents problèmes techniques. Le développement de protocoles de navigation collaborative est un autre défi relevé par les réseaux C2C. L'objectif de propager des informations sur le trafic est de permettre aux véhicules de calculer dynamiquement l'itinéraire le plus rapide. Clairement, de tels calculs dynamiques doivent se baser sur une collaboration inter véhicules, changeant les itinéraires dans une perspective d'optimisation globale du réseau routier.

Compte tenu de ces défis et de la complexité à modéliser les comportements de déplacement dans des systèmes de trafic distribués à grande échelle (voir par exemple (4)), le développement d'outils de simulation réaliste est une étape vitale pour le succès de l'implantation d'un réseau C2C. Motivés par ce besoin, nous présentons ici un framework open source de simulation temps-réel pour le déplacement de véhicules dans différents environnements routiers, incluant plusieurs états de conduite, la communication entre véhicules et des capacités de visualisation sophistiquée. Notre simulateur jette les bases d'une approche systématique permettant de quantifier les compromis de performance à faire entre variables pertinentes telles que le rayon de transmission, la part des véhicules qui communiquent, la fraîcheur des données et la connectivité du réseau, tout cela mettant en avant la dynamique d'une navigation coopérative.

Le Simulateur DIVERT

D'un point de vue abstrait, le réseau routier peut être vu comme un grand graphe dont la topologie est statique et déterminée par la géographie, par dessus lequel nous avons un graphe de communications aléatoires dont la concrétisation spatiale et le schéma de connectivité de chaque point dans le temps sont déterminés à la fois par la position des véhicules circulant sur le réseau routier et par les aspects transmission de l'interface sans fil qu'ils utilisent pour communiquer. Pour obtenir un modèle

de graphe routier réaliste, nous pourrions avoir recours à des informations géographiques de plus en plus disponibles alors que les caractéristiques des transmissions sans fil ont fait l'objet d'une étude approfondie, fournissant des modèles aléatoires très utiles et dont la complexité varie (voir par exemple (5)). Étant donné ces deux aspects, nous structurons notre prototype de simulation (appelé DIVERT - Development of Inter-VEhicular Reliable Telematics) en deux couches : une couche de simulation du trafic basée sur le réseau routier ; et une couche télématique sans fil basée sur le graphe de communications aléatoires. Nous décrivons ces deux couches ci-après.

Couche Trafic

L'information géographique qui permet à la couche de simulation du trafic de fonctionner est transmise à DIVERT dans un format couramment utilisé, tel que le format shapefile, qui décrit la géométrie et la connectivité du réseau routier. DIVERT inclut une interface utilisateur sophistiquée qui permet d'éditer les segments routiers de base pour les enrichir d'informations de bas niveau décrivant les entités de trafic. La figure 1 présente une copie d'écran de cette interface.

Actuellement, DIVERT a été créé pour utiliser les informations géographiques de la ville de Porto, la deuxième plus grande ville du Portugal. Son réseau routier couvre une superficie de 62 kilomètres carrés, avec 1941 rues comptabilisant une longueur totale de 965 kilomètres.

Dans DIVERT, nous modélisons deux types de véhicules : des véhicules qui circulent et qui communiquent, appelés *senseurs* ; et des véhicules qui circulent seulement. Pour chaque type de véhicule, DIVERT distingue, en plus les véhicules de taille normale, ceux de grande taille, en association avec les schémas de déplacement de chacun. Ces schémas de déplacement sont également influencés par une initialisation aléatoire d'attributs tels que l'accélération, le freinage, l'agressivité et la tolérance aux risques. Les senseurs ont un attribut de plus concernant le rayon de transmission sans fil.

DIVERT utilise les couches d'information géographique suivante pour le réseau routier :

1. Deux simples couches des axes centraux des routes représentant, au travers des polygones, la géométrie des segments de routes et leur connectivité topologique. Ces couches peuvent être transmises à DIVERT en tant que shapefile. Une copie de ces couches est présente dans

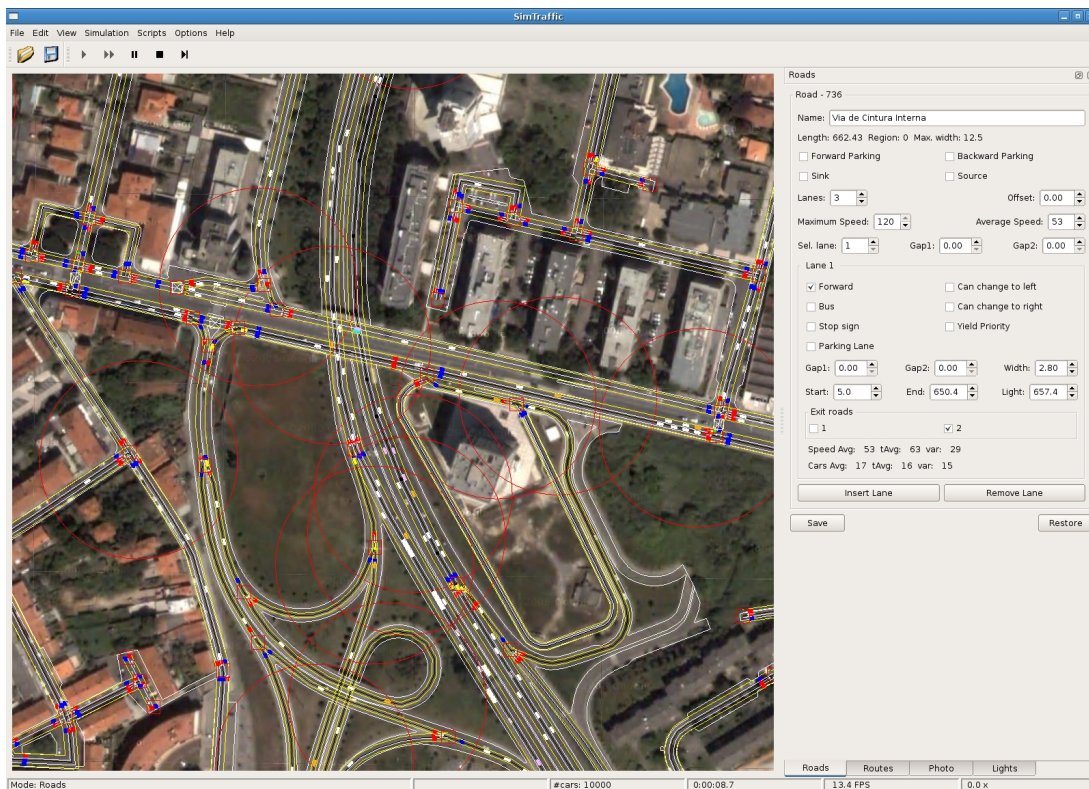


FIG. 1: L'interface d'édition de la route de DIVERT

chaque senseur et est utilisé pour le positionnement GPS et pour la propagation collaborative des conditions de circulation sur les segments de route.

Les couches de bas niveau décrivent en détail le réseau routier de Porto, incluant les informations sur les voies, leur connectivité, la visibilité des intersections, la localisation des feux de signalisation, les relations entre les feux de signalisation, les limitations de vitesse sur les segments de route et les stationnements. Ces couches doivent être éditées via l'interface DIVERT et sont utilisées par le simulateur de trafic.

Une couche raster d'images satellites de la zone de simulation améliore la visualisation du trafic dans notre prototype, qui se base pour le moment sur des données 2D. Le travail sur un modèle 3D est en cours, il permettra non seulement de simuler des accélérations plus réalistes, dépendant de la pente, mais également d'améliorer le modèle de rayon de transmission sans fil qui dépend des effets d'atténuation, de réflexion et d'ombrage en se basant sur une couche 3D de bâtiments.

Concernant les itinéraires des véhicules, DIVERT utilise actuellement un modèle hybride entre des itinéraires pré-définis et des itinéraires générés aléatoirement. Pour les itinéraires générés aléatoirement, notre système sélectionne arbitrairement un point de départ et une destination et calcule l'itinéraire en se basant sur un algorithme du plus court chemin, soit en distance soit en temps. Le chemin le plus court en temps utilise non seulement les limitations de vitesse des segments mais principalement la calibration dynamique de la mobilité moyenne provenant des précédents résultats de simulation. Les routes pré-définies ont une fréquence associée et sont choisies avec soins pour que la simulation se rapproche de la perception que nous avons de la distribution du trafic dans notre cas d'étude actuel, i.e. la ville de Porto. La figure 2 montre l'interface de DIVERT pour créer un itinéraire pré-défini.

La simulation du trafic est paramétrée par le nombre de véhicules et le pourcentage de senseurs parmi ces véhicules. La simulation est initiée en plaçant chaque véhicule de manière aléatoire sur un point de son itinéraire. Les véhicules qui arrivent à leur destination sont retirés. De nouveaux véhicules apparaissent également en cours de simulation, soit

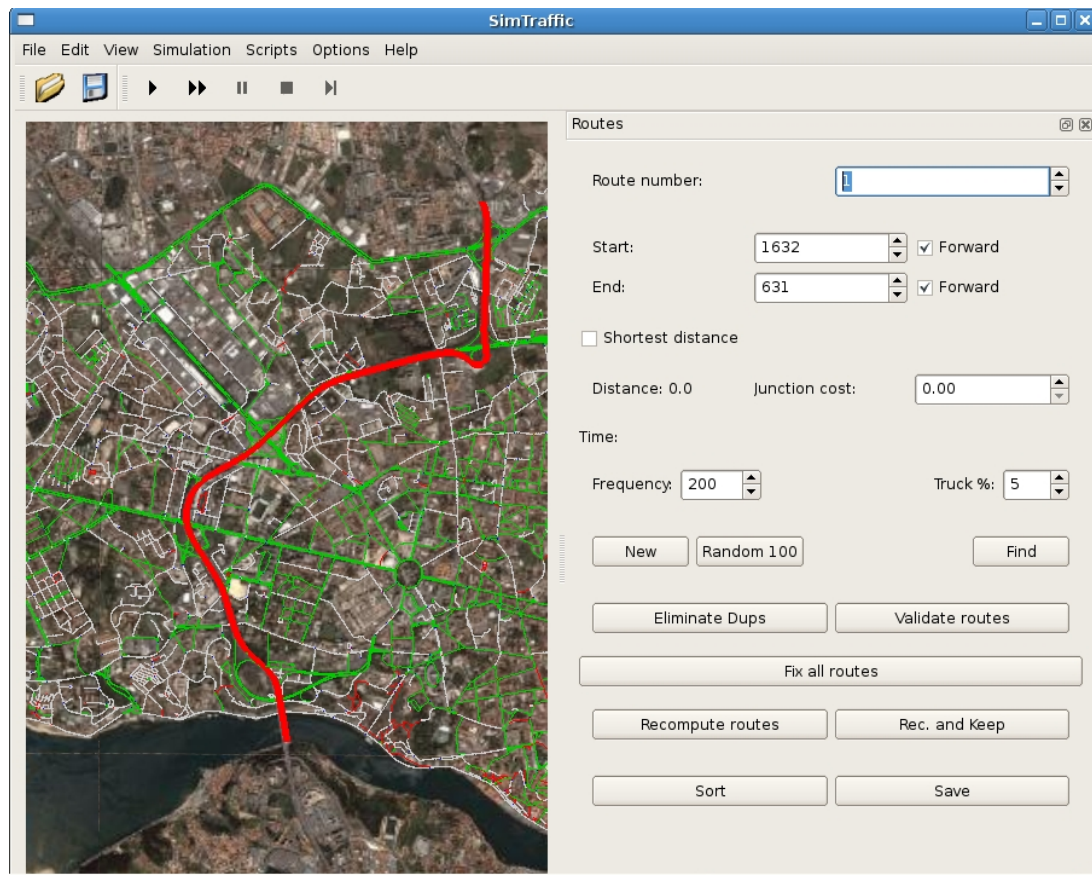


FIG. 2: Création d'itinéraires pré-définis dans DIVERT

au niveau de leur point d'entrée sur la carte soit à partir d'un stationnement, de sorte que DIVERT essaie de maintenir un nombre constant de véhicule pour la simulation.

Il est important de noter que la micro-simulation simultanée de milliers de véhicules, avec le degré de sophistication offert par DIVERT, représente un défi majeur en terme d'optimisation des algorithmes et d'efficacité des structures de données. L'implémentation de DIVERT est multi-thread, exploitant les architectures multi-coeurs de processeurs d'aujourd'hui. Une partitionnement géographique de la zone de simulation est effectuée, permettant à chaque partition d'être simulée indépendamment par un thread.

Couche Télématique Sans Fil

Afin de saisir les aspects de communication entre véhicules, il est nécessaire de définir le niveau d'abstraction en lien avec le canal de communication physique et l'architecture du protocole. À ce niveau préliminaire, nous avons opté pour un modèle

très simple, dans lequel les véhicules communiquent entre eux si leur éloignement est inférieur à un certain seuil, déterminé par le rayon de transmission. Le graphe géométrique aléatoire qui en résulte est largement acceptable en tant qu'une simple mais cependant raisonnable approximation du premier ordre du schéma de connectivité atteint par les réseaux ad-hoc mobiles (6). Une approche plus élaborée serait de considérer la perte du signal, les signaux multiples et les effets d'affaiblissement mais cela serait très pénalisant, en termes de complexité de simulation. Une autre alternative serait de considérer les collisions et les pertes de paquets sur le réseau sans fil. Nous sommes actuellement en train de considérer la possibilité d'intégrer ces aspects dans notre simulation afin d'obtenir un profil de connectivité plus riche.

La couche télématique sans fil simule la communication entre les véhicules. Plusieurs protocoles entre les véhicules peuvent être intégrés dans cette couche, tels que les protocoles utilisés pour des applications de sécurité ou pour la propagation des conditions de circulation et pour la navigation colla-

borative. L'implémentation de cette couche télématique sans fil dépend de la couche de simulation du trafic qui joue le rôle d'un serveur de positionnement global, émulant un récepteur GPS dans chacun des senseurs. L'interface entre les deux couches est donc réalisée grâce à des commandes de type GPS où le simulateur de trafic génère la position de chaque senseur, en latitude et longitude, leur vecteur de vitesse ainsi que le timestamp global qui permet la synchronisation temporelle pour l'échange de données entre véhicules. Le calcul des véhicules présents dans le rayon de transmission (paramétré) d'un senseur donné, l'implémentation de protocoles d'échange de messages entre les véhicules et le déclenchement des actions basées sur les informations collectées telle que la modification d'un itinéraire sont à la charge de la couche télématique sans fil. L'architecture des deux couches dans DIVERT permet qu'une action déclenchée, telle qu'une modification d'itinéraire, puisse être retransmise à la couche de simulation du trafic, affectant ainsi le comportement du véhicule.

Actuellement, les protocoles que nous avons implémentés en utilisant DIVERT visent à la propagation collaborative d'informations sur la circulation sur chaque segment de route. Chaque senseur stocke une structure de données commune où le couple (*AverageSpeed*, *Freshness*) décrit la vitesse moyenne atteinte par les senseurs traversant chaque segment de route, de même que la quantification de la rapidité de prise en compte de l'estimation de la vitesse moyenne. Ces couples sont mis à jour soit par chaque senseur en se basant sur la route qu'il emprunte et les informations du GPS, soit par les informations agrégées collectées par les communications sans fil diffusées par les autres senseurs. Typiquement les senseurs ne transmettent que des informations sur des segments de route pour lesquels la fraîcheur (*Freshness*) dépasse un certain seuil prédéfini. DIVERT est capable de simuler la propagation des informations de circulation en utilisant différents rayons de transmission sans fil. Nos résultats montrent qu'il existe une valeur critique pour le rayon de transmission au delà de laquelle la diffusion des informations de trafic est suffisante pour qu'un grand nombre de véhicules puissent calculer une carte d'embouteillages compréhensible et précise. Cette observation est liée de façon frappante au phénomène physique de la percolation qui est bien connu pour régir la connectivité de grandes classes de réseaux sans fil (7). Lorsque le rayon de transmission dépasse le seuil critique, le graphe représentant la connectivité sans fil acquiert un composant géant

sur lequel les informations de trafic circulent très vite et sur de longues distances.

Développements à venir

DIVERT est en développement constant. Actuellement l'implémentation compte 50000 lignes de code C++, incluant les composants d'interface graphique et de visualisation. L'interface de DIVERT permet également le lancement de programmes simples, écrits en Python, qui sont très utiles pour permettre à l'application de générer différents types de rapports, des vidéos de simulation ou de l'aide dans l'édition des cartes. Une redéfinition de l'architecture du simulateur en plusieurs modules indépendants est en cours. Notre objectif est d'avoir un grand nombre de modules indépendants qui simplifieront pour une grande communauté d'utilisateurs les modifications du simulateur pour l'adapter à des besoins spécifiques.

Un effort particulier est mis sur le développement d'un langage de spécification qui faciliterait la configuration de DIVERT avec différentes données géographiques. Les couches géographiques servant de base à la simulation s'appuient déjà sur des standards ouverts mais il reste encore de nombreux éléments pour lesquels le manque d'automatisation constitue un obstacle à une large utilisation de DIVERT. En particulier, la calibration réelle des itinéraires et de leur fréquentation est un problème crucial. Nous essayons d'atteindre l'automatisation à travers l'analyse et le traitement des journaux (logs) des téléphones portables qui permettent, dans les environnements urbains, de fournir des descriptions de trajets sur un réseau routier avec une grande précision. Cette technique est connue sous le nom de *floating car data*. L'automatisation de la construction de matrices d'origine/destination réalistes, avec des données géographiques définissant le réseau routier basées sur des standards largement utilisés, fournirait les données nécessaires pour tester DIVERT dans différents scénarios.

Nous continuons la conception, l'implémentation et le test des différents protocoles pour les communications entre véhicules au travers de DIVERT. Nous nous concentrons toujours sur l'efficacité du trafic qui nous a permis de soulever des problèmes concernant l'optimisation collaborative du trafic. Nous espérons voir d'autres protocoles, développés ailleurs dans le monde, utiliser le framework DIVERT et ceci dans tous les domaines de la communication C2C.

Pour ceux qui ont raté la démonstration de DI-

VERT au FOSS4G 2007, une vidéo de simulation est disponible en ligne.²¹

Remerciements

Nous remercions le Conseil Municipal de Porto pour nous avoir fourni les cartes du réseau routier. Ce travail a été en partie financé par la Fondation Portugaise pour la Science et la Technologie (FCT) dans le cadre du projet MYDDAS (POSC/EIA/59154/2004) et par des fonds accordés au LIACC et au IT grâce au Programa de Financiamento Plurianual et au POSC.

Bibliographie

- [1] CAR 2 CAR Communication Consortium Manifesto. Version 1.1 <http://www.car-2-car.org/>, 2007.
- [2] C. L. Robinson, L. Caminiti, D. Caveney, and K. Laberteaux. Efficient coordination and transmission of data for coopera-

tive vehicular safety applications. In *VANET '06 : Proceedings of the 3rd international workshop on Vehicular ad hoc networks*, pages 10–19, New York, NY, USA, 2006. ACM Press.

- [3] X. Yang, J. Liu, F. Zhao, and N. H. Vaidya. A vehicle-to-vehicle communication protocol for cooperative collision warning. In *MobiQuitous*, pages 114–123. IEEE Computer Society, 2004.
- [4] A. K. Saha and D. B. Johnson. Modeling mobility for vehicular ad-hoc networks. In *VANET '04 : Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks*, pages 91–92, New York, NY, USA, 2004. ACM Press.
- [5] P. Gupta and P. Kumar. The capacity of wireless networks. *Information Theory, IEEE Transactions on*, 46(2) :388–404, 2000.
- [6] F. Xue and P. Kumar. The Number of Neighbors Needed for Connectivity of Wireless Networks. *Wireless Networks*, 10(2) :169–181, 2004.
- [7] L. Booth, J. Bruck, M. Franceschetti, and R. Meester. Covering algorithms, continuum percolation and the geometry of wireless networks. *Ann. Appl. Probab.*, 13(2) :722–741, 2003.

Hugo Conceição
DCC-FC & LIACC, University of Porto
hc AT dcc.fc.up.pt

Luís Damas
DCC-FC & LIACC, University of Porto
luis AT dcc.fc.up.pt

Michel Ferreira
DCC-FC & LIACC, University of Porto
michel AT dcc.fc.up.pt

João Barros
DCC/FC & IT - University of Porto
barros AT dcc.fc.up.pt

²¹Démonstration de DIVERT au FOSS4G2007 : <http://myddas.dcc.fc.up.pt/divert/>

GRASS et la Modélisation des Risques Naturels

Une Approche Intégrée pour la Simulation des Flux de Débris — Premiers Résultats d'une Application dans les Andes Centrales

Martin Mergili et Wolfgang Fellin, traduit par Marie Silvestre

Contexte

Les flux de débris sont des mouvements rapides et massif d'eau et de débris, représentant un risque considérable lorsqu'ils interfèrent avec la population, les bâtiments ou les infrastructures. Ils sont souvent déclenchés par des pluies fortes ou prolongées ou par une fonte des neiges rapide. La mobilisation de matériaux se produit à cause de glissements de terrain provoqués par des mouvements de rotation ou de translation sur des pentes saturées ou présentant un surplomb ou à cause de détachements dus à l'érosion de surface ou au flux de débris lui-même. Différents modèles existent pour simuler les processus inclus dans les flux de débris, par exemple pour le détachement (*r.sim.sediment* dans l'environnement SIG GRASS), pour l'hydrologie du sol et la stabilité de la pente (14) ou pour les zones de dépôt des flux de débris (9; 7). Des approches plus intégrées basées sur les SIG, telles qu'ont tenté de le faire par exemple (1) ou (11), sont rares. De telles approches seraient bénéfiques pour une évaluation rapide des seuils hydrologiques de risques potentiels de flux de débris au vu des risques encourus. cet article décrit et débat du développement d'un tel modèle sous la forme d'un module raster pour GRASS. Le modèle est conçu pour de petits bassins versants (de quelques kilomètres carrés) et il est actuellement testé sur sept zones d'études le long du couloir routier international allant de Mendoza (à l'ouest de l'Argentine) au centre du Chili, croisant la partie la plus élevée des Andes (figure 1). Les résultats préliminaires pour la zone d'étude *Guido A* sont présentés.

Modèle

Implementation et conception du modèle

Le modèle de simulation est implémenté dans un module raster de GRASS appelé *r.debrisflow* et

basé sur le langage de programmation C. La gestion des données est facilitée par l'utilisation de scripts shell. Le modèle est actuellement à une étape intermédiaire de développement, des améliorations majeures aux niveaux technique et méthodique sont attendues très prochainement. De plus, une interface utilisateur pour la gestion des données devrait être créée. Pour le moment la dernière version de développement est téléchargeable sur le site internet du premier auteur. *r.debrisflow* est un environnement de travail constitué de nombreux sous-modules décrits plus en détail ci-après, le design général du modèle est illustré sur la figure 2. Les sous-modules peuvent être combinés de deux manières différentes, dépendant de la disponibilité des informations en entrée :

Mode de simulation 1 : La modélisation complète de l'hydrologie, de la stabilité, du détachement et du dépôt est effectuée pour un nombre défini de pas de temps au cours d'un événement pluvieux ou de fonte des neiges, nécessitant un jeu de données complet en entrée dont des données météorologiques, un modèle d'élévation, des paramètres mécaniques et hydrologiques concernant le sol et les caractéristiques de l'hydrologie de surface (dont l'occupation du sol).

Mode de simulation 2 : Les zones d'initiation du flux de débris sont définies manuellement (par exemple par de la cartographie de terrain ou par orthophotos) et seul le dépôt est calculé. L'avantage de ce mode est qu'il requiert nettement moins de données en entrée que l'autre mais, d'un autre côté, il ne permet pas de prédire des événements futurs.



FIG. 1: Zones d'études. Les résultats préliminaires pour Guido A sont présentés.

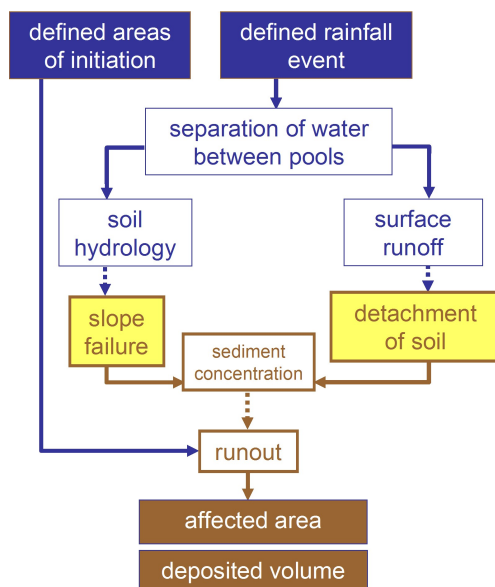


FIG. 2: Design général du modèle.

Données d'entrée sur l'eau

La pluviométrie est lue dans le fichier d'entrée et ajoutée au système en augmentant la valeur de la table de l'eau de surface de chaque cellule, valeur à laquelle on soustrait l'interception de la pluie. S'il y a une couverture neigeuse, la fonte de la neige est calculée pour chaque cellule en fonction d'un facteur de température journalier défini par l'utilisateur et ajoutée à la table de l'eau de surface.

Hydrologie du sol et stabilité du terrain

Pour ce sous-module, une approche tri-dimensionnelle raster est utilisée, allant jusqu'à la roche mère (si elle est connue) ou jusqu'à une profondeur de sol définie par l'utilisateur. Le sol est considéré comme homogène sur toute sa hauteur concernant ses propriétés physiques, hydrologiques et mécaniques. Le flux vertical est calculé grâce à la loi de Darcy. Si le contenu en eau d'une cellule dépasse 90 % du contenu maximum, le flux d'eau souterraine est supposé parallèle à la pente et on teste si la cellule est stable ou pas en utilisant une approche de stabilité infinie du terrain (14). Pour chaque pixel, le bas de la cellule la plus profonde avec un facteur de sécurité inférieur à 1 est considéré comme un plan de faiblesse (figure 3). Il a été mis en avant que cette approche constitue une approximation grossière de la réalité avec l'hypothèse de plus mauvais cas : le rôle stabilisant du mouvement vertical de l'eau est négligé et le rôle déstabilisant de la pente supposée parallèle est complètement intégré dans les cas de saturation. En réalité, ces deux composantes sont combinées, créant des conditions plus stables que dans le modèle.

Ruissellement de surface et détachement

Toute l'eau qui ne s'est pas infiltrée dans le sol est ajoutée à la table de l'eau de surface de la cellule correspondante. La vitesse du flux d'une cellule à une autre est calculée en utilisant l'équation de Manning. Si aucun canal d'écoulement vers une cellule en aval

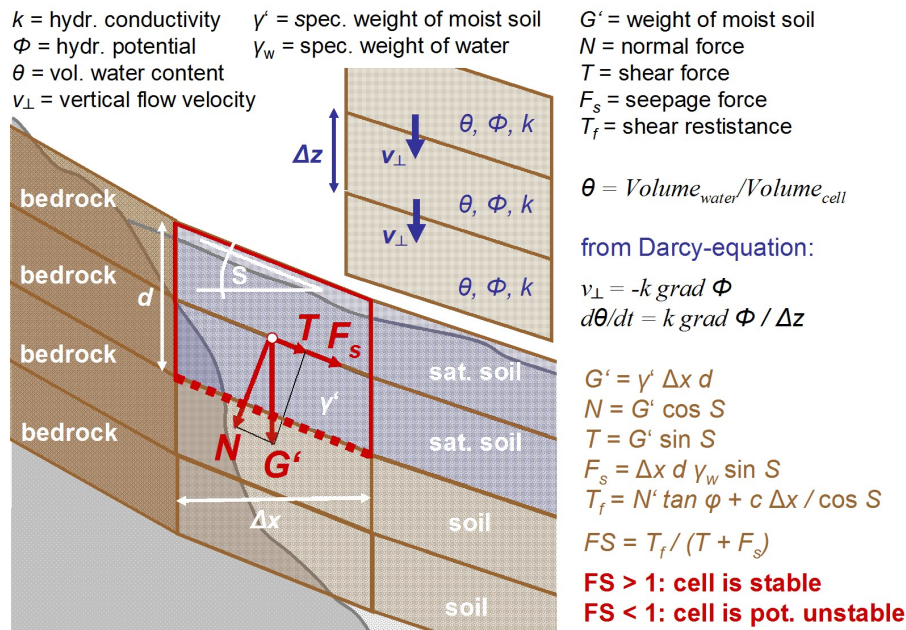


FIG. 3: Hydrologie souterraine et modèle de stabilité de la pente.

n'est indiqué, le flux total est transmis à toutes les cellules aval, pondéré par l'angle de pente. Si un canal est défini, le flux total y est transféré. La capacité de transport est calculée en utilisant l'équation de (15) qui est sensée être la plus adaptée aux conditions des zones d'études (6). Les taux de détachement sont dérivés de la capacité de transport. La structure du modèle est représenté sur la figure 4.

Dépôt du flux de débris et sédimentation

Le terme flux de débris est utilisé ici dans son sens strict : dans un régime de flux non-hydraulique et en excluant les écoulements d'eau fortement chargée en sédiments. C'est pour cette raison que la concentration en sédiment est calculée pour le volume de sol mobilisé dans chaque cellule :

1. Si la concentration en sédiments dépasse une valeur seuil (6), on considère que le volume mobilisé devient entièrement un flux de débris. Si la mobilisation a lieu à cause d'une rupture de pente, la cellule est marquée et le dépôt de toutes les cellules qui tombent est calculé à la fin de l'évènement. Si la mobilisation a lieu à cause du détachement, le dépôt est calculé directement à la fin du pas de temps. Il s'agit d'une première approximation (comparée à ci-dessous) car le modèle empirique de dépôt uti-

- lisé néglige le temps nécessaire à la déposition.
2. Si la concentration en sédiment est trop faible pour la création d'un flux de débris, le matériau est retiré par le ruissellement superficiel puis, soit déposé plus bas dans la pente soit enlevé du système, conformément à l'équation (15) (voir figure 4). Bien que le concept de ruissellement de surface ne fasse pas partie du concept de flux de débris, il fournit des informations complémentaires intéressantes et est donc également considéré.

Le dépôt du flux de débris peut lui-même être simulé en utilisant des modèles physiques (9; 7) mais ils sont complexes et leur intégration est difficile dans un environnement SIG. Il a donc été décidé d'utiliser une combinaison d'approches empiriques, en premier lieu pour estimer la distance du dépôt et la distribution du volume déposé (figure 5). (10) a développé une approche permettant de différencier les zones de raclage des zones de dépôts, en utilisant un angle de pente seuil et le ratio entre la distance verticale de raclage et la distance horizontale de dépôt. (2) et (8) ont utilisé le volume mobilisé, les angles et les distances de dépôt pour estimer l'amplitude du flux de débris. Ces approches, cependant, ont l'inconvénient de ne pas pouvoir faire la distinction entre les zones de raclage et celles de dépôt et, par conséquent, la distribution des matériaux dé-

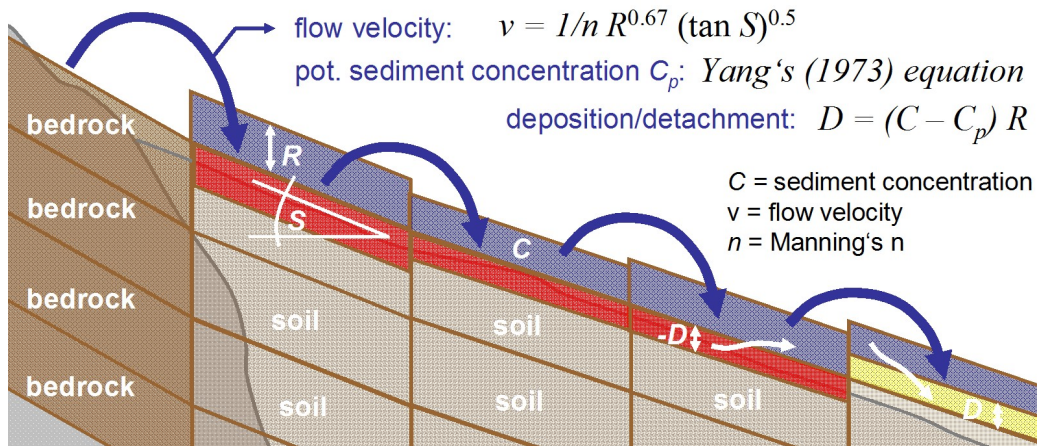


FIG. 4: Ruissellement de surface et modèle de détachement.

posés. Cette approche a été intégrée au modèle de la manière suivante :

1. L'approche de Vandre a été appliquée avec des paramètres définis pas l'utilisateur pour estimer les volumes érodés et déposés.
2. Les approches de Corominas et al. et de Rickenmann ont été appliquées indépendamment puis combinées en un indice.

Le flux de débris est dirigé en aval de manière séparée pour chaque cellule instable, suivant un chemin aléatoire (3), pondéré par l'angle de pente et l'existence d'un canal défini, jusqu'à ce que le critère d'arrêt pour les trois approches soit rempli. Bien que chaque cellule soit traitée indépendamment, le volume mobilisé requis pour la distance de dépôt est calculé à chaque passe d'initiation du flux de débris. Dans la zone de raclage, on considère que la colonne de sol saturé est entièrement retirée, sans jamais dépasser la profondeur d'initiation. On considère que les volumes initiaux et érodés se distribuent dans la zone de dépôt en prenant la forme d'un cône croissant par l'avant.

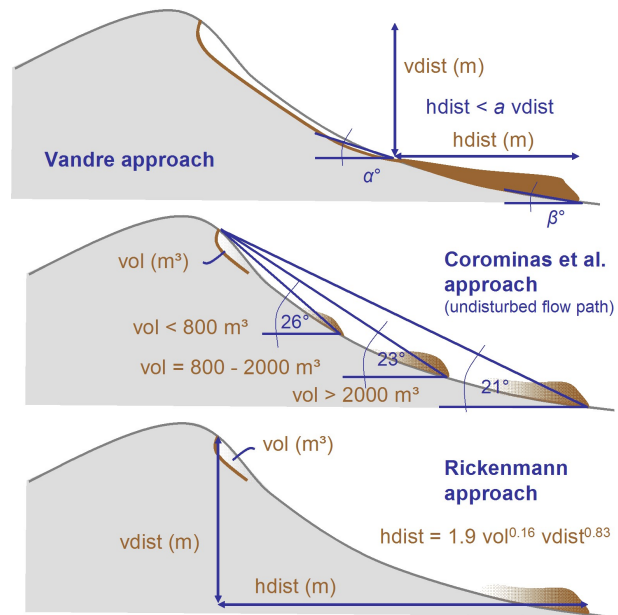


FIG. 5: Modèles de dépôt.

Premiers résultats

Le modèle a été testé sur la zone d'étude de Guido A (voir figure 1). Les sols de ce bassin, principalement constitués de sols granitiques résiduels, sont relativement homogènes. Il a donc été décidé d'utiliser un seul jeu de paramètres pour les sols sur la zone d'étude :

texture	ρ_d kg/m³	c_s N/m²	φ deg.	Θ_s	k_f cm/h
Sable	1850	0	40.0	0.43	29.7

ρ_d est la densité apparente sèche du sol, c_s est la cohésion du sol, φ représente l'angle de friction interne, Θ_s est la contenance maximale (saturation) en eau et k_f est la conductivité hydraulique saturée.

Les figures 6 et 7 illustrent les zones géographiques de l'initiation du flux de débris dans la zone d'étude Guido A et les types de changements de surface dus à un flux de débris, basés sur la cartographie des zones d'initiation et sur les types d'érosion et de dépôt calculés (mode de simulation 2). La ligne blanche qui coupe la partie droite des cartes correspond à la route internationale qui coïncide approximativement à la partie la plus en aval des dépôts de flux de débris observés. Les figures 8 à 11 montrent quelques résultats de simulation pour un événement pluvieux hypothétique de 100 mm, ce qui correspond à la valeur journalière maximale observée près de la station météorologique locale, consistant donc l'hypothèse du pire scénario (mode de simulation 1). Toutes les cartes montrent des schémas possibles en comparaison des observations de terrain. Les zones d'initiation et de dépôt du flux de débris sont correctement localisées mais sont surestimées par rapport aux observations de terrain (ce qui n'est pas surprenant étant donnée l'hypothèse prise du pire scénario). Les volumes calculés de sédiments déposés sur la route internationale entrent dans la même gamme de valeur que ceux reportés par les autorités responsables de cette route. En simulant l'impact de plus faibles événements pluvieux, les résultats du modèle correspondent bien aux conclusions de (5), à savoir que les flux de débris dans la Vallée Mendoza se produisent lorsque la pluviométrie journalière dépasse 6,6 à 12,9 mm.

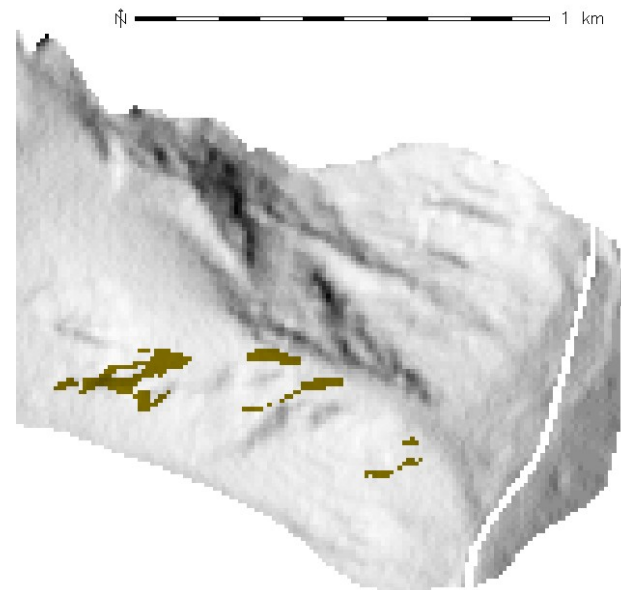


FIG. 6: Cartographie des zones d'initiation clairement identifiées de flux de débris précédents. La profondeur de l'initiation est estimée à 0,75 m selon les observations de terrain.

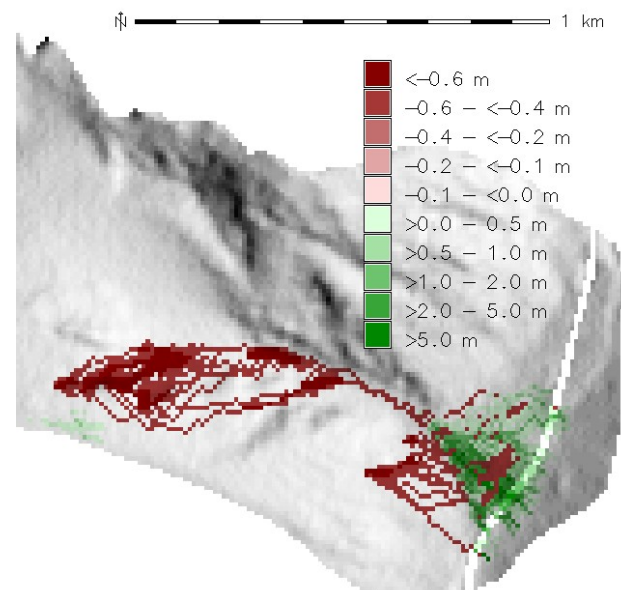


FIG. 7: Simulation des changements de hauteurs de terrain dus à un flux de débris, utilisant les zones d'initiation cartographiées.

Discussion et Perspectives

Bien que les résultats préliminaires semblent plausibles pour Guido A, le modèle de simulation souffre encore de nombreux défauts sur lesquels il

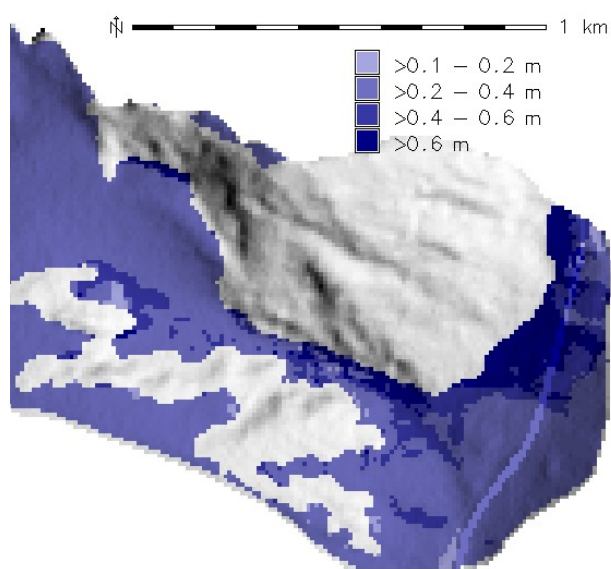


FIG. 8: Profondeur maximale de saturation calculée pour un événement pluvieux de 100 mm.

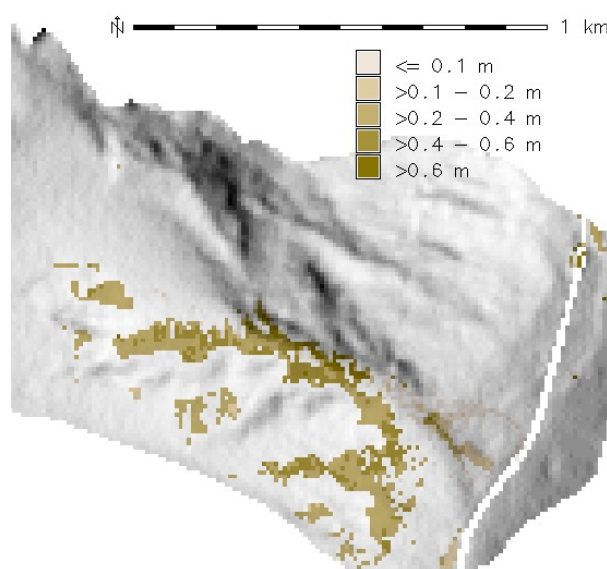


FIG. 9: Zones simulées d'initiation potentielle de flux de débris calculées pour un événement pluvieux de 100 mm.

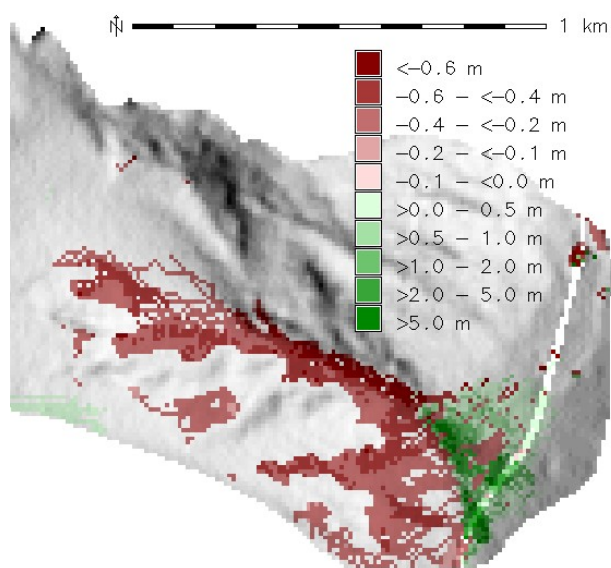


FIG. 10: Simulation des changements de hauteurs de terrain dus à un flux de débris déclenché par un événement pluvieux de 100 mm.

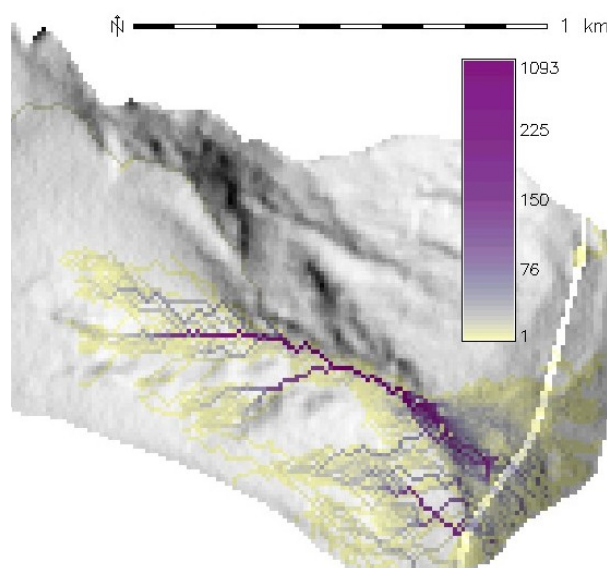


FIG. 11: Indice de flux de débris pour un événement pluvieux de 100 mm, dénombrant le nombre de cellules dont les matériaux mobilisés atteignent le pixel.

faut retravailler.

1. L'infiltration de l'eau dans le sol n'est pas encore modélisée d'une manière satisfaisante, il faudra donc affiner cette approche (modèle Green-Ampt). Les flux d'eau souterraine parallèles à la pente vont être intégrés, également pour permettre une meilleure approximation de l'hydrologie réelle des sols.
2. Le modèle de stabilité de la pente tel qu'il est appliqué pour le moment n'est valide que pour des pentes droites, infinies et sans cohésion. Pour de petites ruptures, cette hypothèse est suffisamment proche de la réalité mais pour des ruptures de pente rotationnelles plus marquées, elle n'est pas satisfaisante, la courbure de la pente doit être prise en compte. (12) et (13) pourraient servir d'exemples pour une telle approche.
3. Les approches empiriques pour le dépôt de flux de débris doivent être complétées par l'intégration d'un modèle physique de dépôt selon (9) et (7), ou du moins par une interface avec un modèle de dépôt qui ne soit pas basé sur les SIG.
4. Une extension intéressante serait d'introduire des éléments probabilistes dans le modèle de stabilité du terrain (en prenant le modèle SINMAP comme exemple) et dans la règle de distinction entre un flux de débris et d'autres types de mouvements.
5. Enfin, le modèle doit être appliqué sur les autres zones d'études (voir figure 1) et les résultats doivent être comparés avec prudence aux observations de terrain et aux données de validation (rapports sur les volumes de matériaux déposés sur la route internationale).

Grâce aux optimisations mentionnées, r.debrisflow sera un outil important pour l'évaluation de l'ampleur potentielle de flux de débris en réponse à des épisodes pluvieux ou de fonte des neiges définis, incluant la possibilité de déterminer des seuils météorologiques pour les risques de flux de débris. Cependant, il est important de noter que tous les résultats ne présentent que des occurrences potentielles avec les cas des pires scénarios ou probabilités - il ne sera probablement jamais possible de prédire la vraie réponse d'un terrain à un événement météorologique dans le monde réel car la nature est trop complexe pour être pleinement comprise dans tous ses détails.

Remerciements

Le projet est financé par le *Doktoratsstipendium aus der Nachwuchsförderung der LFU* et par des subventions venant de l'Office pour les Relations Internationales et les Facultés pour les Sciences Naturelles, Université d'Innsbruck, ainsi que de l'Académie des Sciences Autrichienne et du Gouvernement Fédéral de la Haute-Autriche. Merci à Hans Stoetter, Axel Borsdorf, Clemens Geitner et Stella Moreiras pour leur soutien.

Bibliographie

- [1] A. Burton, J.C. Bathurst (1998) Physically based modelling of shallow landslide sediment yield at a catchment scale. *Environmental Geology* 35,2-3 : 89-99.
- [2] J. Corominas, R. Copons, J.M. Vilaplana, J. Altamir, J. Amigó (2003) Integrated Landslide Susceptibility Analysis and Hazard Assessment in the Principality of Andorra. *Natural Hazards* 30 : 421-435.
- [3] B. D. Hughes (1995) Random walks and random environments : Volume 1 : Random Walks. Oxford University Press. 652 pp.
- [4] X. Li (2007) Finite element analysis of slope stability using a nonlinear failure criterion. *Computers and Geotechnics* 34 : 127-136.
- [5] S.M. Moreiras (2005) Climatic effect of ENSO associated with landslide occurrence in the Central Andes, Mendoza Province, Argentina. *Landslides* 2 : 53-59.
- [6] J.S. O'Brien (2003) FLO-2D Users' Manual Version 2003.06, July 2003. FLO-2D Software Inc., Nutrioso, Arizona, USA. 232 pp.
- [7] S.P. Pudasaini, K. Hutter (2007) Avalanche Dynamics : Dynamics of rapid flows of dense granular avalanches. Springer, Berlin Heidelberg. 602 pp.
- [8] D. Rickenmann (1999) Empirical Relationships for Debris Flows. *Natural Hazards* 19 : 47-77.
- [9] S.B. Savage, K. Hutter (1989) The motion of a finite mass of granular material down a rough incline. *Journal of Fluid Mechanics* 199 : 177-215.

- [10] B.C. Vandre (1985) Rudd Creek debris flow. In : D.S. Bowles (ed) : Delineation of landslide, flash flood, and debris flow hazards in Utah. Utah Water Research Laboratory, Utah State University, Logan, Utah, 117-131.
- [11] V. Wichmann (2006) Modellierung geomorphologischer Prozesse in einem alpinen Einzugsgebiet - Abgrenzung und Klassifizierung der Wirkungsräume von Sturzprozessen und Muren mit einem GIS. Eichstaeter geographische Arbeiten 15 : 231 pp. In German.
- [12] P.L. Wilkinson, M.G. Anderson, D.M. Lloyd (2002) An integrated hydrological model for rain-induced landslide prediction. *Earth Surface Processes and Landforms* 27 : 1285-1297.
- [13] M. Xie, T. Esaki, G. Zhou, Y. Mitani (2003) Three-dimensional stability evaluation of landslides and a sliding process simulation using a new geographic information systems component. *Environmental Geology* 43 : 503-512.
- [14] M. Xie, T. Esaki, M. Cai (2004) A time-space based approach for mapping rainfall-induced shallow landslide hazard. *Environmental Geology* 46 : 840-850.
- [15] C.T. Yang (1973) Incipient motion and sediment transport. *Journal of the Hydraulics Division. Proceedings of the American Society of Civil Engineers* 99 : 1679-1703.

Martin Mergili

*University of Innsbruck, Institute of Geography
and Austrian Academy of Sciences, Mountain Research
www.uibk.ac.at/geographie/personal/mergili
[martin.mergili AT uibk.ac.at](mailto:martin.mergili@uibk.ac.at)*

Wolfgang Fellin

*University of Innsbruck, Institute of Infrastructure
www.uibk.ac.at/geotechnik/staff/fellin-en
[wolfgang.fellin AT uibk.ac.at](mailto:wolfgang.fellin@uibk.ac.at)*

Une Base de Données Spatiales pour l'Intégration des Données du Projet de Gestion des Ressources Naturelles du Rondonia

Luis Fernando Bueno, Luiz Gilberto Dall'Igna, Marcelo Vitor Amaral Campos, Thiago de Lima Martarole, Diego Gomes Ferreira, traduit par Sidonie Christophe

Mots clés : Base de données spatiales, ZSEE-RO, PLANAFLORO, Intégration de données

Résumé

Cet article présente la mise en place d'une base de données spatiale pour l'intégration d'informations provenant du Zonage Socio-Economique et Ecologique de l'Etat du Rondonia (ZSEE/RO). L'objectif du projet est de modéliser une nouvelle base de données pour le ZSEE/RO, rendant ainsi les données et informations utilisées dans les différents projets disponibles. Toute l'évaluation a alors été effectuée en s'appuyant sur la quantité de données et d'informations du ZSEE/RO. Des conversions de fichiers ont été réalisées afin de les rendre accessibles dans les formats standards. Ensuite la cohérence des données converties a été vérifiée afin d'assurer un contrôle de qualité à toutes les échelles. Les applications mises au point exclusivement pour le ZSEE/RO ont été reconfigurées et installées. Les principaux résultats obtenus jusqu'ici ont trait à la reconnaissance de la quantité de données ZSEE/RO et à la création de conditions nécessaires pour développer l'accès aux données d'origine. Divers fichiers ont été convertis en formats universels. De plus, les problèmes qui ont été détectés, concernant la qualité des données, sont les données non-géocodées, les incohérences dues au changement d'échelle et l'incohérence entre les données papier et les données numériques.

Introduction

Cet article s'intéresse à la mise en place de la base de données spatiale pour l'intégration de ZSEE/RO, qui est en cours de développement en partenariat avec le Secrétariat de Planification et de Coordina-

tion Générale de l'Etat du Rondonia. Le PLANAFLORO (Gestion des Ressources Naturelles du Rondonia) est à l'origine du ZSEE/RO.

L'objectif général du projet consiste à modéliser une nouvelle base de données du ZSEE/RO, permettant ainsi aux différents projets d'accéder aux données. Il vise tout particulièrement à l'élaboration des travaux suivants : évaluer le contenu des bases de données du ZSEE/RO de diverses origines, modéliser et construire une base de données spatiale pour le stockage des données et informations, et convertir les données dans un nouveau format Internet.

Le projet PlanaFloro

Le décret d'Etat No. 3782 du 14 Juin 1988 divise l'Etat du Rondonia en zones de protection et zones d'activités agricoles et agro-forestières, créant ainsi le ZSEE/RO, Zonage Socio-Economique et Ecologique de l'Etat du Rondonia (4). Le Rondonia a été le premier état brésilien à avoir adopté des lois de préservation de la nature avec le Projet de Gestion des ressources Naturelles du Rondonia - PLANAFLORO, dédié à la question de l'octroi des terres pour les populations autochtones, qui s'est déroulé entre 1992 et 1999 au sein du ZSEE/RO.

En tant que composante du PLANAFLORO, le ZSEE/RO a appuyé une enquête sur le terrain à l'échelle du 1/250 000è, concernant divers domaines de connaissances comme : la végétation, la géologie, la pédologie, la géomorphologie, la faune, l'occupation et l'utilisation du sol, etc. Les partenaires responsables de cette mission consistaient en différentes équipes de travail, une pour chaque domaine de connaissances. Par conséquent, les données thématiques ont été stockées en utilisant différentes solutions en accord avec les nécessités des différentes équipes. L'implémentation a été faite sous ArcINFO; cependant cette base était pauvre en attributs (Dall'Igna, 2005). Les études thématiques finalisées sont devenues la base du diagnostic de l'Etat du Rondonia, et, en utilisant leurs résultats, une ap-

proche écologique du ZSEE/RO a été élaborée.

Procédures initiales

Afin de définir la méthodologie de travail pour l'intégration d'un important volume de données concernant le ZSEE/RO, voici quelques éléments qui étaient à prendre en compte : les données originales étaient disponibles sous des formats hétérogènes, beaucoup d'entre elles étaient inconnues des techniciens des agences de l'état et des chercheurs potentiels. L'obsolescence technologique a aussi été considérée, même si le PLANAFLORO n'a été initié qu'en 1992. Ainsi le projet a consisté à évaluer et collecter données et informations pour le ZSEE/RP, créer des archives sur des supports numériques, personnaliser des logiciels pour PLANAFLORO et produire du matériel papier, comme des cartes, des lettres et des rapports.

Les principaux formats de fichiers trouvés ont les suivants :

- ADF : Fichiers de couverture ArcINFO 7.x. Les thèmes couverts par le ZSEE, les données vecteur et les données descriptives, aéroports, courbes de niveaux, aires protégées par la loi, etc. sont sous ce format.
- LAN : Fichiers Raster. Images satellite Landsat, avec une résolution de 30 mètres, couvrant l'ensemble de l'Etat du Rondonia.
- GRA : Fichiers graphiques. Cartes générées par le ZSEE pour impression.
- RTL : Fichiers graphiques. Tous les thèmes couverts par le ZSEE comme l'hydrologie, la géologie, l'hydrogéologie, la géomorphologie, la végétation et toutes les cartes préparées de l'échelle 1/250 000 à l'échelle 1/1 000 000.

En plus des formats listés ci-dessus, des rapports, des images et des feuilles de calcul aux formats .DOC, .XLS, .TIFF, .JPG et .EPS ont été identifiés.

Des conversions de fichiers ont été réalisées afin de les rendre accessibles et standards, depuis leurs formats propriétaires nécessitant des logiciels spécifiques, en des formats universellement accessibles par diverses technologies. Les fichiers qui étaient au format vectoriel .ADF ont été convertis au format Shapefile ESRI. Les images satellite au format .LAN ont été converties en GeoTiff et les fichiers graphiques .RTL et .GRA ont été convertis au format .PDF. Les rapports et images initialement en .DOC, .TIF, .JPG et .EPS ont été aussi convertis en .PDF. Toutes les feuilles de calcul, bases de données, et quelques fichiers TXT ont été chargés en tant que

tables dans PostGreSQL. Quelques-unes des applications développées et utilisées spécifiquement pour le ZSEE/RO ont été reconfigurées et installées. Leurs fonctionnalités vont être testées pour déterminer si ces applications seront utilisées, remplacées ou mises à jour.

TerraView, ArcView, ArcInfo et IDRISI sont des logiciels également utilisés. La base de données géographiques du ZSEE/RO a été structurée avec ArcINFO, des extensions ont été développées pour ArcView pour la visualisation et la consultation. L'équipe pédologie a utilisé IDRISI dans ses analyses donnant la possibilité de travailler avec les données de SIGTERON aussi bien qu'avec celles d'ArcInfo et d'ArcView.

TerraView a été choisi comme une alternative, en tant que logiciel libre capable d'importer des shapefiles et des images au format geoTIFF, et dans les formats vecteur et raster universels. L'autorisation pour sa redistribution et/ou ses modifications est sous la licence GNU GPL. TerraView est une application basée sur la géo-librairie TerraLib. On peut ainsi manipuler des données vecteur et raster, les deux formats étant stockés dans des SGBD relationnel ou spatial du marché, comme ACCESS, PostgreSQL, MySQL et Oracle (7). La Figure 1 présente l'interface de TerraView.

Ensuite la cohérence des données converties a été vérifiée, avec l'intention de vérifier la qualité de l'information à différentes échelles. Les procédures de vérification de la cohérence ont impliqué le géoréférencement de points de données disponibles, la superposition des rasters entre eux et la vérification des données géocodées avec les liens entre la géométrie des objets et leurs attributs descriptifs.

Base de données spatiales ZSEE/RO

Une infrastructure pour le stockage et la diffusion des données spatiales a été implémentée, grâce à des logiciels libres : PostGreSQL (5) avec l'extension PostGIS (6), les serveurs cartographiques MapServer et GeoServer (2) ainsi que le serveur de catalogues GeoNetwork (3).

Les couches vecteur ont été converties et stockées dans une base de données PostgreSQL grâce aux fonctionnalités fournies par l'extension PostGIS. Les données descriptives des objets spatiaux, contenues dans des fichiers de formats hétérogènes, comme des fichiers .DBF et des feuilles de calcul numériques, ont été converties en tables PostgreSQL et reliées

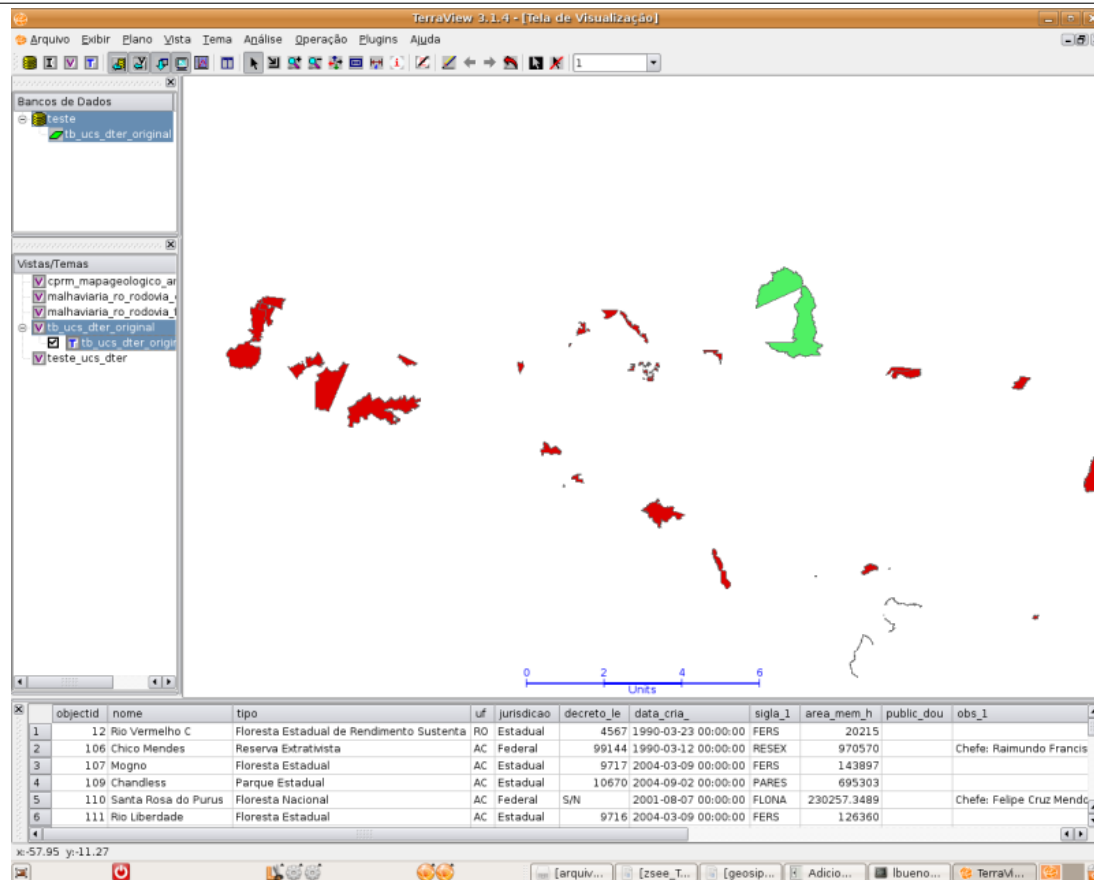


FIG. 1: L'interface Terraview

à leurs représentations géométriques. Les données vecteur stockées comportent des informations sur différents sujets étudiés par le projet ZSEE/RO : géologie, sol, occupation et utilisation du sol, données socio-économiques, lignes électriques, flore, faune, climatologie, etc. De plus les données se réfèrent à des images raster Landsat à une résolution de 30 mètres, qui couvrent l'Etat du Rondonia en entier.

Toute la documentation thématique du ZSEE/RO a été organisée et stockée dans GeoNetwork. La fonctionnalité de recherche littérale dans GeoNetwork permet d'accéder simplement à un grand nombre d'informations. Pour chaque thème traité par ZSEE/RO, une métadonnée correspondante a été créée où toute la documentation concernant ce sujet est fournie, à travers les ressources de distribution de GeoNetwork.

L'outil de visualisation pour les données spatiales choisi est I3Geo. I3Geo (Integrated Interface for Internet of Geoprocessing tools) est une application développée pour l'accès et l'analyse des données géographiques. Basée sur un logiciel libre, MapServer

en particulier, elle utilise Internet Explorer et Firefox comme navigateurs internet standards. I3Geo est sous licence GPL et peut être utilisé et incorporé par toute institution intéressée, sans coût. En adoptant les standards internationaux pour l'interopérabilité, I3Geo intègre des fonctions qui facilitent l'accès distant aux données en permettant l'établissement de réseaux coopératifs. Les opérations qui sont normalement accessibles uniquement dans les programmes des ordinateurs personnels, qui opèrent donc en local, sont disponibles dans I3Geo : génération de graphes, analyses de données sous forme tabulaire, opérations spatiales, etc. (8).

Résultats

Le principal résultat obtenu à ce jour est la reconnaissance du contenu de ZSEE/RO et la création des conditions nécessaires pour développer l'accès aux données d'origine. Ensuite, beaucoup de fichiers ont été convertis en formats universels, totalisant plus

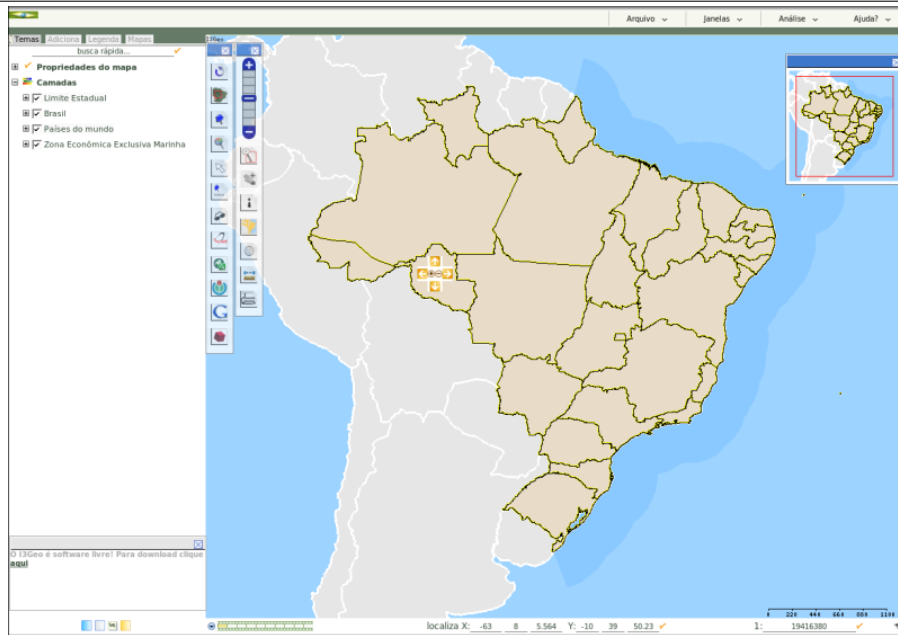


FIG. 2: L'interface I3Geo

de 500 fichiers convertis, et stockés dans la base de données spatiales et disponibles à la consultation. De plus, des problèmes de qualité des données ont été détectés, comme les données non-géocodées, l'incohérence des données à différentes échelles et l'incohérence des données disponibles sur cartes papier par rapport aux cartes stockées au format numérique.

Bibliographie

- [1] *Vulnerabilidade natural à erosão da Folha Presidente Médici à Rondônia*. 2005. 138p. Dissertação (Mestrado em Desenvolvimento regional e Meio Ambiente) Fundação Universidade Federal de Rondônia, Porto Velho.

- [2] What is Geoserver. (online). 2007 <http://docs.codehaus.org/display/GEOS/Home> May 2007
- [3] GeoNetwork opensource Community website. (online) 2007 <http://geonetwork-opensource.org/> March 2007
- [4] *Dos projetos de desenvolvimento, ao desenvolvimento dos projetos : o PLANAFLORO em Rondônia*. 2002. 285p. Tese (Doutorado em Ciências Humanas/Sociedade e Meio Ambiente) - Universidade Federal de Santa Catarina, Florianópolis.
- [5] PostgreSQL Global Development Group, PostgreSQL 8.1.0 Documentation. (online) 2005 <http://www.postgresql.org/docs/manuals/>
- [6] PostGIS Manual (online) 2005 <http://postgis.refractor.net/documentation/> March 2007
- [7] Projeto TerraView (online) 2007 <http://www.dpi.inpe.br/terraview> April 2007
- [8] Wikibooks I3geo (online) <http://pt.wikibooks.org/wiki/I3geo> April 2007

Luis Fernando Bueno, Luiz Gilberto Dall'Igna
Sistema de Proteção da Amazônia - SIPAM
Avenida Lauro Sodré, 6500
Aeroporto, Porto Velho, Rondônia
CEP 78903 - 711, Fone (69) 3217 6360
[luis.bueno AT ipam.gov.br](mailto:luis.bueno@ipam.gov.br), [luiz.dalligna AT sipam.gov.br](mailto:luiz.dalligna@sipam.gov.br)

Marcelo Vitor Amaral Campos, Thiago de Lima Martarole, Diego Gomes Ferreira
Instituto Luterano de Ensino Superior de Porto Velho - ILES/ULBRA

*Rua João Goulart, 666
Bairro Mato Grosso Porto Velho, Rondônia
CEP 78915 - 450*

*phone (69) 3216 7600
campos.mv AT gmail.com, shinoda.br AT gmail.com,
diegogferreira AT msn.com*

GeoSIPAM

Logiciel libre et Open Source appliqué à la protection de l'Amazonie brésilienne

Luis Fernando Bueno, Weberson Gabriel, Pablo Filetti Moreira, Artur Henrique Villa Real F. Ramos, Fabio Augusto das Dores Silva, Marcelo Campos Brito, traduit par Sidonie Christophe

Keywords : *Données spatiales, Amazonie brésilienne, SIPAM, Infrastructure, GeoSIPAM*

Résumé

Cet article présente l'infrastructure GeoSIPAM implémentée et maintenue par le Système de Protection de l'Amazonie - SIPAM pour le stockage et la diffusion de données spatiales. Des applications open source libres (FOSS) ont été utilisées pour le stockage et la diffusion des données spatiales. Le système de gestion de base de données relationnelle PostgreSQL a été choisi, avec son extension PostGIS pour le stockage et la manipulation des données spatiales. Plusieurs applications (GeoServer, GeoNetwork, InterMap) ont été adaptées pour le SIPAM. Les adaptations effectuées ont concerné des procédures de développement améliorant ces outils utilisés quotidiennement par les collaborateurs. Les interfaces applicatives de GeoNetwork et d'InterMap ont été modifiées afin de les rendre accessibles et d'y ajouter des éléments. Un soin particulier a été apporté pour que l'infrastructure soit adaptée au stockage, à la manipulation et à la diffusion des données spatiales. La personnalisation a permis l'intégration des applications à l'environnement organisationnel du SIPAM. Il a aussi été vérifié que l'utilisation des standards OGC était correcte afin que les systèmes soient interopérables.

Introduction

La base de données du Système de Protection de l'Amazonie - SIPAM intègre des informations de l'Amazonie Brésilienne Légale (BLA) à jour. L'utilisation de ces informations dans les projets développés par le SIPAM et par les agences partenaires fournit de la connaissance qui facilite la planification et la coordination des actions gouvernementales globales, visant à la protection, l'intégration sociale et le développement durable de la région.

Ce travail présente l'infrastructure GeoSIPAM implémentée et maintenue par le SIPAM pour le stockage et la diffusion des données spatiales. GeoSIPAM vise à offrir l'intégration et l'évaluation des données afin d'aider à la planification et à la coordination des actions des politiques publiques de développement en Amazonie Brésilienne Légale. En particulier, les objectifs de GeoSIPAM sont les suivants : afficher sur Internet ou Intranet les métadonnées des projets portés par le SIPAM ; mettre à disposition des cartes, des images géocodées et de l'information textuelle associée sur Internet ; visualiser des cartes, des images, et des informations pertinentes pour d'autres institutions, à travers l'utilisation des protocoles standards OGC (1).

Les FOSS dans GeoSIPAM

Le développement et l'utilisation de logiciels libres en géomatique ont mis à disposition un nombre croissant d'outils. Lors de la conception de GeoSIPAM, il a été décidé d'utiliser des logiciels open source libres (FOSS) pour le stockage et la diffusion des données spatiales. Les outils choisis sont les suivants :

- PostgreSQL Système de gestion de base de données relationnelle
- PostGIS Extension spatiale pour PostgreSQL
- GeoServer OpenGIS Web Feature Server Transactionnel
- GeoNetwork Catalogue pour gérer les ressources géoréférencées sur le web
- InterMap Map viewer, généralement configuré comme étant intégré à GeoNetwork

Infrastructure pour l'Analyse Spatiale

L'architecture logicielle choisie est organisée en couches impliquant

- un système de gestion de base de données (SGBD) pour le stockage, la gestion et la manipulation des données
- des cartes et des serveurs de catalogues pour la diffusion et l'information sur les données
- autant que des interfaces pour l'accès aux données basé sur les standards établis par l'OGC.

Le SGBD PostgreSQL a été choisi, ainsi que son extension PostGIS pour le stockage et la manipulation des données spatiales. Les interfaces des applications GeoNetwork et InterMap ont subi des modifications pour les rendre accessibles et uniques.

La figure 1 présente la visualisation de la carte interactive au sein de l'interface personnalisée d'InterMap.

Les applications GeoServer, GeoNetwork et InterMap sont des outils développés en Java qui implémentent les standards Web Map Services (WMS), Web Feature Services (WFS) et Catalogue Service Web (CSW). Elles ont aussi été personnalisées pour une utilisation au SIPAM. Les personnalisations effectuées ont concerné des procédures de développement qui ont permis aux collaborateurs de réaliser leurs tâches quotidiennes. Quelques personnalisations réalisées par l'équipe GeoSIPAM sont présentées par la suite.

Initialement, InterMap répondait aux exigences du standard WMS, mais pas au standard WFS, les deux étant proposés par GeoServer. Dans InterMap un service a été créé pour transporter des demandes vers GeoServer à travers un service WFS, rendant ainsi possible le téléchargement de fichiers raster. Ainsi une option pour l'appel du service GetFeature de WFS a été développée dans le format SHAPE - ZIP et le service services.map.DownloadService a été implémenté. Un bouton correspondant a été ajouté dans la barre d'outils de la couche, ainsi que la figure 2 le montre. Cette fonctionnalité a été implémentée par des modifications du code source, impliquant de la programmation en JavaScript, en XSL et en Java.

```
public Element exec(Element params,
    ServiceContext context) throws Exception
{
    // Get request parameters
    int id = Integer.parseInt(
        params.getChildText(
            Constants.MAP_SERVICE_ID
        ));

    // Get the MapMerger object from the user session
    MapMerger mm = MapUtil.getMapMerger(context);

    // Get the layer name
    String nome = mm.getService(id).getName();
    // Get the WMS server url
    String serverUrl = mm.getService(id).getServerURL();

    // Declare and set the WFS server url
    int pos = serverUrl.indexOf("/wms")+1;
    String wfsFeatureInfoUrl = serverUrl.substring(0, pos)
        + "wfs?request=GetFeature&service=WFS&version=1.0.0";
}
```

```
+ nome;
Element e1 = WmsGetCapClient.getCapabilities(serverUrl);

// Return the necessary information for
// the service execution
return new Element("response")
    .addContent(WmsGetCapClient.getCapabilities(serverUrl))
    .addContent(new Element("serviceName").setText(nome))
    .addContent(
        new Element(
            Constants.URL).setText(
                WfsFeatureInfoUrl));
}
```

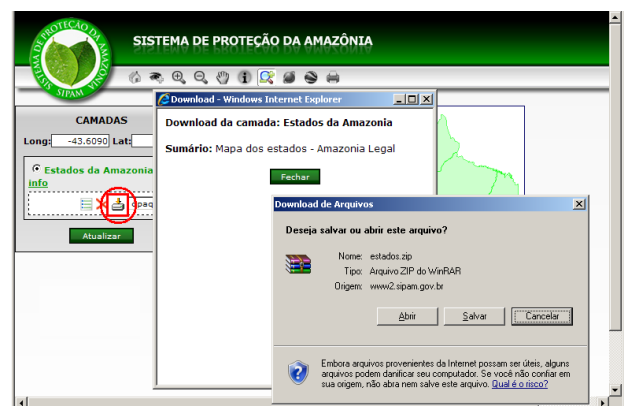


FIG. 2: Interface InterMap Personnalisée pour le téléchargement de couches

L'option qui permet de proposer une couche à télécharger est réalisée par un service WFS. Le GetFeature de WFS permet de télécharger indifféremment toutes les couches disponibles sur le serveur. L'option a été créée pour permettre/empêcher le téléchargement dans l'écran du FeatureType de GeoServer. Cette information est aussi envoyée au demandeur WMS par le service getCapabilities. Le bouton Téléchargement sera montré dans InterMap dès que l'option correspondante sera qualifiée dans GeoServer, comme présenté à la figure 3. La fonctionnalité a été implémentée grâce à des modifications du code source de GeoServer écrit en Java.

Une autre fonctionnalité implémentée permet la visualisation de données en même temps que données fournies par l'API Google Earth. Un bouton inséré dans la barre d'outils d'InterMap crée une fenêtre flottante en mouvement qui montre les données spatiales, synchronisant les données raster de GeoServer avec les images de l'API de Google Earth. La position et le zoom sont conservés même après des modifications dans la visualisation de l'API Google Earth. La figure 4 présente la fenêtre Google Earth implémentée dans InterMap. Cette

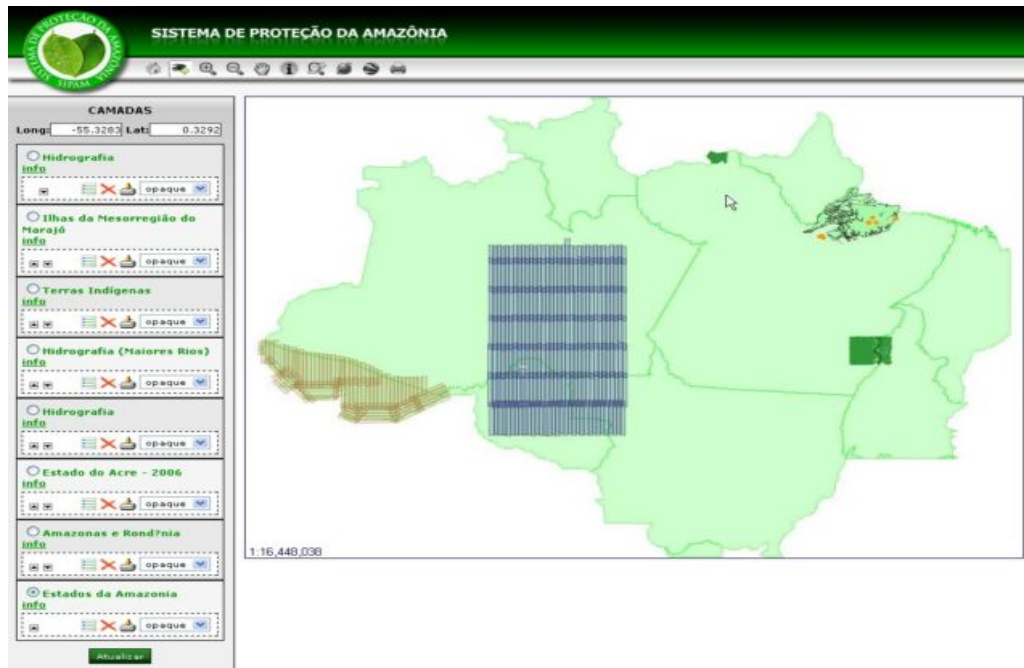


FIG. 1: Visualisation de la carte interactive, utilisation de l'interface personnalisée d'InterMap

fonctionnalité a été implémentée grâce à l'inclusion de code Javascript dans le fichier `im_main.xml`. La ligne de code ci-dessous montre la référence à l'API Google, dans le fichier `im_main.xml`.

```
<script
  src="http://maps.google.com/maps? \
    file=api&v=2&key=GOOGLE_API_KEY"
  type="text/javascript">
</script>
```

L'expression `GOOGLE_API_KEY` doit être remplacée par la clé obtenue sur le site de l'API Google²². Pour obtenir la clé, vous devez accéder au site et fournir l'URL du serveur sur lequel InterMap est installé. La clé est envoyée sur l'adresse email du demandeur, qui doit avoir un compte GMail. Les fonctions "addWMS" et "montaMapa" ont été codées et ajoutées au fichier `im_main.xml`. La fonction "addWMS" est utilisée pour reconstruire la liste des couches à montrer à l'API Google, à chaque fois qu'une nouvelle couche est ajoutée dans InterMap. La fonction "montaMapa" est utilisée pour construire la carte Google. Le code source des fonctions "addWMS" et "montaMapa" est présenté ci-dessous.

```
// Add WMS servers and layers types
```

²²Google API site : <http://www.google.com/apis/maps/>

```
function addWMS(servidor, camada){
  achou = false;
  //Check if the server exists.
  // If it does, add a new layer.
  // If it does not register a new
  // server and layer
  for (contador = 0; wmsurl.length>contador; contador++){
    if( wmsurl[contador] == servidor) {
      // server found.. add layer
      layersG[contador] = layersG[contador] + ',' + camada
      achou = true;
    }
  }
  if(!achou) {
    // server not found, register in the vector
    pos = wmsurl.length;
    wmsurl[pos] = servidor;
    layersG[pos] = camada;
  }
  listalayers = listalayers + camada
}

// Set up the google map on the screen
function montaMapa() {
  var camadaWMS = new Array();
  if( visivelG ) {
    mapG = new GMap2(document.getElementById("mapaG"));

    // A handler is called when a
```

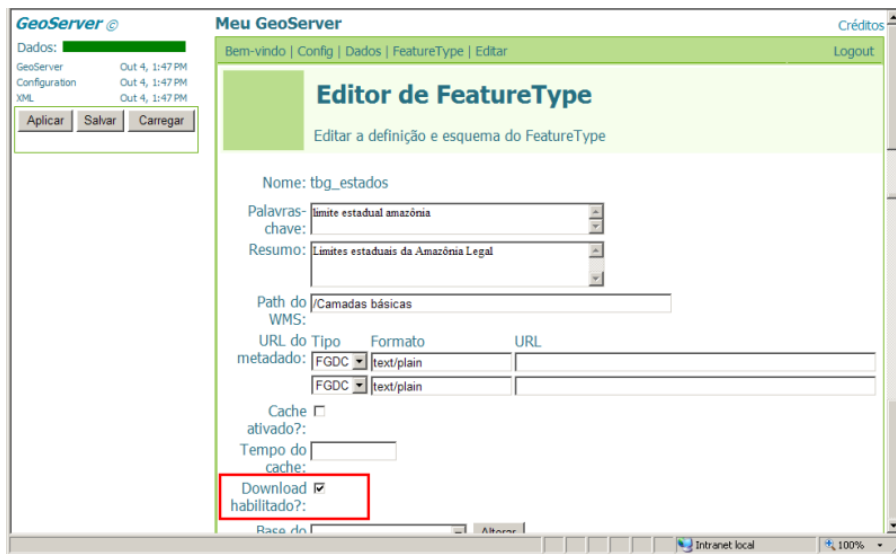


FIG. 3: Interface de Geoserver personnalisée permettant le téléchargement

```
// map movement break event is called
GEvent.addListener(mapG, "moveend", function() {
// If the movement was done at the Google map,
// the Intermap map changes too
if( actOrigem == '' ){
var center = mapG.getCenter();
// Obtain the center
main.map.zMapCenLat = center.lat();
// in LatLong
main.map.zMapCenLng = center.lng();
main.map.moveIntermap();
// moves the Intermap map
}
actOrigem = '';
if(navm){setTimeout("pngfix()",1000); }
});
// Handler is called
// when a drag action event is called
GEvent.addListener(mapG, "dragstart", function() {
actOrigem=""
});

// Handler is called
// when a zoom event is called
GEvent.addListener(mapG, "zoomend", \
function(zoomant, zoomatu) {
if( actOrigem == '' ) {
var bbox = mapG.getBounds();
// Obtain the center
main.map.zmapminlat = bbox.getSouthWest().lat();
// em latitude e longitude
main.map.zmapminlng = bbox.getSouthWest().lng();
// e move o mapa
main.map.zmapmaxlat = bbox.getNorthEast().lat();
main.map.zmapmaxlng = bbox.getNorthEast().lng();
```

```
main.map.zoomIntermap(zoomant, zoomatu);
// no intermap
}
if (actOrigem != 'X'){actOrigem = '';}
});

// Handler is called
// when a drag action event is called

var layer1=[G_SATELLITE_MAP.getTileLayers()[0], \
G_HYBRID_MAP.getTileLayers()[1]];
var custommap1 = new GMapType(layer1, \
G_SATELLITE_MAP.getProjection(), \
"Google", G_SATELLITE_MAP);

var layer3=[G_SATELLITE_MAP.getTileLayers()[0], \
G_HYBRID_MAP.getTileLayers()[1]];
for( contador=0; wmsurl.length > contador; \
contador++){
// create tile layers
camadaWMS[contador]= new GTileLayer( \
new GCopyrightCollection(""),1,17);
if( wmsurl[contador] != '' )
{ if( layersG[contador] != '' ) {
camadaWMS[contador].myLayers=layersG[contador];
camadaWMS[contador].myBaseURL=wmsurl[contador];
camadaWMS[contador].getTileUrl=CustomGetTileUrl;
if(navm) {camadaWMS[contador].myFormat='image/png'}
} }
layer3[contador+2] = camadaWMS[contador];
}

var custommap3 = new GMapType(layer3, \
G_SATELLITE_MAP.getProjection(), \
"Intermap", G_SATELLITE_MAP);
```

```

mapG.getMapTypes().length = 0;
mapG.addMapType(custommap3);
mapG.addMapType(custommap1);
SWLatLng = new GLatLng(SWLat, SWLng)
NELatLng = new GLatLng(NELat, NELng)
bbox      = new GLatLngBounds(SWLatLng, NELatLng)
zoom      = mapG.getBoundsZoomLevel(bbox)
gmapt = mapG.getMapTypes()
zoomMax = \
    gmapt[1].getMaximumResolution(mapG.getCenter())
if( zoom > zoomMax) {zoom = zoomMax;}
actOrigem = 'X'
mapG.setCenter(bbox.getCenter(), zoom);
mapG.addControl(new GLargeMapControl());
mapG.addControl(new GMapTypeControl());
with(document.getElementById( \
    "google").style){top = "113px";left = "239px";}
actOrigem = ' ' } }

```

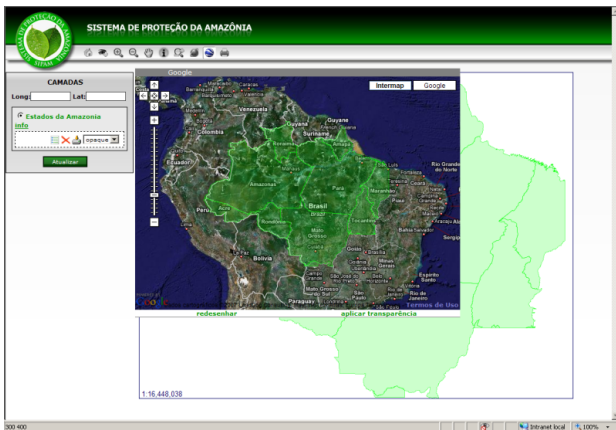


FIG. 4: Interface InterMap personnalisée avec l'API Google

A cause du grand nombre de couches disponibles sur les serveurs WMS, la visualisation de la liste des couches sous forme d'une liste déroulante devient inadéquate. L'affichage des couches a été modifié afin d'être chargé sous forme d'un arbre avec les options "refermer" et "explorer". La fonctionnalité a été implémentée par l'ajout de code Javascript dans le fichier `im_get_services.xml`. L'API dTree, un arbre Javascript libre, a été utilisée. dTree peut être récupéré en ligne.²³ Dans ce cas, la stratégie adoptée a été de mettre les couches des serveurs WMS dans un tableau. dTree crée ensuite l'arbre des couches à partir de ce tableau. Figure 5 présente la visualisation des couches sous forme d'un arbre.

²³dTree JavaScript API : <http://www.destroydrop.com/javascripts/tree/>

Résultats

Une infrastructure ayant les capacités de stockage, de manipulation et de diffusion des données spatiales a été choisie et implémentée. La base de données a été remplie avec des données spatiales concernant l'Amazonie Brésilienne Légale, incluant des sujets tels que l'hydrographie, la géologie, l'utilisation et l'occupation du sol, les ressources minérales, les limites administratives, les routes, les sols, etc. Des données spatiales venant des autres partenaires du SIPAM, comme l'Agence Nationale de l'Eau (ANA), le Service Géologique Brésilien (CPRM), l'Institut Brésilien de Géographie et de Statistiques, IBGE ont aussi été intégrées dans GeoSIPAM. Le volume des données stockées grandit chaque jour. Les personnalisations locales des packages SIG FOSS disponibles ont permis d'implémenter des éléments du SIPAM qui sont spécifiques à l'environnement organisationnel. Il a aussi été vérifié que l'utilisation des normes OGC respecte l'interopérabilité des divers systèmes. Les données spatiales et les produits élaborés pour l'équipe opérationnelle du SIPAM ont été enregistrés dans le cadastre et stockés dans le système GeoSIPAM. Cette information est en cours de diffusion sur le réseau interne mais aussi de façon public en général, à travers le portail web du SIPAM sur Internet. En plus des principaux partenaires qui accèdent constamment au site, on peut trouver les Secrétariats à l'Environnement des Gouvernements d'Etat de l'Amazonie, la Police Environnementale, des universités régionales et d'autres encore. Des contrôles statistiques

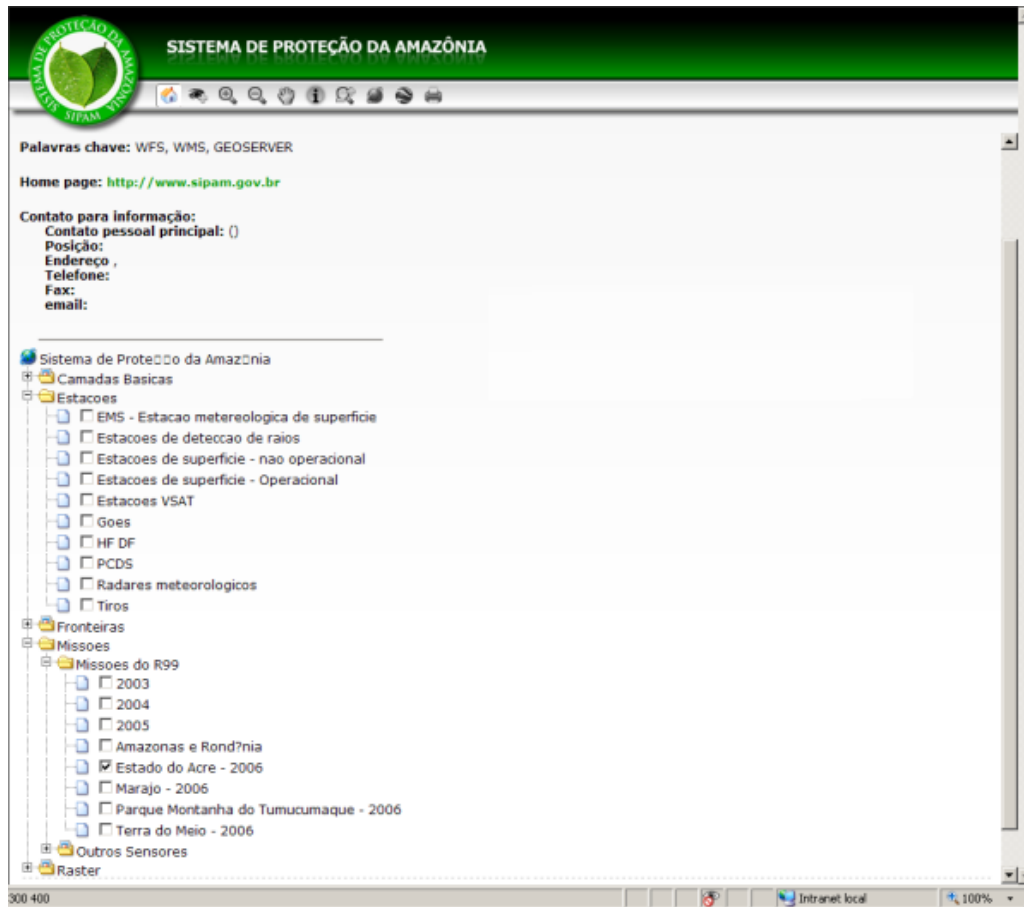


FIG. 5: Interface InterMap personnalisée avec la visualisation de la liste des couches

ont montré plus de 100,000 connexions à GeoSIPAM.

Bibliographie

[1] Bueno, L.F., P.G. Zuza e W. Gabriel. GeoSIPAM : Manual do Usuário. Presidência da República, CENSIPAM, CTO/Pv,

2007, 52 páginas.

- [2] Câmara G.; C. Davis, e A.M.V. Monteiro. Introdução à Ciência da Geoinformação. (online). 2001. <http://www.dpi.inpe.br/gilberto/livro/introd/>. 14 April 2007
- [3] Garnett, J. e C. Holmes. What is Geoserver. (online). 2007. <http://docs.codehaus.org/display/GEOS/Home> 10 May 2007.
- [4] GeoNetwork opensource Community website. (online). 2007. <http://geonetwork-opensource.org/> 19 March 2007.
- [5] Landrö, Geir. dTree. (online). 2003. <http://www.destrodrop.com/javascripts/tree/> 05 Setembro 2007
- [6] PostgreSQL Global Development Group. PostgreSQL 8.1.0 Documentation. (online). 2005. <http://www.postgresql.org/docs/manuals/> 05 May 2007
- [7] Refrations Research. PostGIS Manual. (online). 2005. <http://postgis.refrations.net/documentation/> 27 March 2007
- [8] Wheeler, D.A. Why Open Source Software/Free Software (OSS/FS, FLOSS, or FOSS)? Look at the Numbers! (on line) http://www.dwheeler.com/oss_fs_why.html 03 June 2007

Luis Fernando Bueno
System of Protection of Amazon - SIPAM
SPO Área 05, Quadra 03, Bloco "K" Brasília - DF
CEP 70610 - 200
[luis.bueno AT sipam.gov.br](mailto:luis.bueno@sipam.gov.br)

Weberson Gabriel
[weberson.gabriel AT sipam.gov.br](mailto:weberson.gabriel@sipam.gov.br)

Pablo Filetti Moreira
[pablo.moreira AT sipam.gov.br](mailto:pablo.moreira@sipam.gov.br)

Artur Henrique Villa Real F. Ramos
[artur.ramos AT sipam.gov.br](mailto:artur.ramos@sipam.gov.br)

Fabio Augusto das Dores Silva
[fabio.silva AT sipam.gov.br](mailto:fabio.silva@sipam.gov.br)

Marcelo Campos Brito
[marcelo.brito AT sipam.gov.br](mailto:marcelo.brito@sipam.gov.br)

Le Système de Suivi de la Déforestation Amazonienne

Une Importante Base de Données Environnementale Développée sur TerraLib et PostgreSQL

Freitas, U. M., Ribeiro, V. O., Queiroz, G. R., Petinatti, M. R. et Abreu, E. S., traduit par B. Véler

Résumé

L'Institut National du Brésil pour la Recherche Spatiale, et la Fondation pour les Sciences, les Technologies et les Applications de l'Espace ont développé un système complet de suivi, basé sur la technologie Open Source TerraLib²⁴ (Camara, G., et al., 2000), dans le but de cartographier et calculer le taux annuel de déforestation de l'Amazonie au Brésil. TerraLib rend possible l'archivage de données géographiques raster et vecteur dans un certain nombre de DBMS, propriétaires ou non, dont PostgreSQL. Il contient aussi des méthodes pour le traitement et l'analyse de données en format image et vecteur. Une application client, nommée TerraAmazon, a été développée en C++ et avec QT (version 3), boîte à outils libre d'interface graphique, qui fonctionne sous Linux et Windows. Les données sont gérées par PostgreSQL version 8.2, tournant sur un serveur LINUX. L'application prend en charge tout le flux de traitement des données, rassemblant environ 600 images satellites, les préparant, les segmentant et les classifiant pour une interprétation humaine ultérieure, et une édition dans un environnement multi-utilisateurs complémentaire. La base de données contient approximativement 2 millions de polygones complexes, et 20 Go d'images satellite en haute résolution sont rajoutées chaque année, en utilisant le schéma de résolution pyramidale de TerraLib. Un site web est fourni pour la visualisation et l'analyse des données en pleine résolution, utilisant l'extension PHP et le serveur OGC WMS de TerraLib.

Introduction

Le Brésil conduit un large projet environnemental pour suivre et représenter la déforestation dans l'éco-région de l'Amazonie, en utilisant des images satellites. Tous les ans, une carte et le taux de dé-

forestation annuelle sont produits et publiés sur Internet par l'Institut National du Brésil pour la Recherche Spatiale ("Instituto Nacional de Pesquisas Espaciais" – INPE). L'éco-région de l'Amazonie brésilienne couvre une superficie de 4,7 millions kilomètres carrés. Étant donné cette immense superficie, la tâche est très exigeante. Chaque année, une couverture complète de la région par des images satellites d'une résolution de 20 à 30 mètres, est acquise, traitée automatiquement, et analysée par des spécialistes de la télédétection.

Le produit final des données sur la déforestation a une précision cartographique compatible avec une échelle de 1 :250 000. Ce projet est dénommé PRODES – raccourci pour Projet de Déforestation Légale de l'Amazonie - démarré à la fin des années 1980, et qui a évolué d'une procédure d'interprétation analogique à une procédure entièrement numérique. La méthodologie actuelle a été mise en place en 2005, et ses caractéristiques techniques sont présentées dans cet article.

Avant que le nouveau système ne devienne opérationnel, les cartes de déforestation étaient produites avec SPRING, un SIG et logiciel libre de traitement d'image, développé par l'INPE.²⁵ Dans le but de produire la carte de déforestation en entier, 229 bases de données indépendantes, couvrant chacune la superficie d'une image satellite LANDSAT 5, étaient nécessaires. Cette méthodologie avait créé un environnement complexe à gérer, dans la mesure où chaque base de données était transférée d'une station de travail à une autre pour subir un traitement spécifique, impliquant des dizaines de spécialistes.

En plus de la complexité de cette première méthodologie, de nouveaux besoins forcèrent le gouvernement brésilien à l'améliorer en 2005. La première nouvelle exigence fut d'introduire une source multi-satellite, dans le but de garantir la disponibilité de données, y compris dans le cas d'interruption d'un satellite. Des images d'une résolution de 20 m venant du capteur CCD du satellite CBERS (China Brazil Earth Resources Satellite), des images de LANDSAT 5 d'une résolution de 30 m et de DMC (Disaster Monitoring Constellation) d'une résolution de 32 m sont désormais utilisées. La figure 1 montre les images satellites utilisées pour la cartographie de la défore-

²⁴Librairie TerraLib : <http://terralib.org/>

²⁵Le projet SPRING de l'INPE : <http://www.dpi.inpe.br/spring>

tation en 2005. La seconde exigence était le besoin d'une publication rapide des données, dans le but de créer les conditions permettant au gouvernement d'appliquer ses politiques plus rapidement, avant la période suivante de déforestation.

L'utilisation d'images CBERS en tant que source de données principale a porté le nombre d'images à 570 et l'usage de la méthodologie précédente aurait créé un environnement encore plus complexe pour la gestion des données. La solution était de créer une base de données unique, compatible avec toutes les opérations, dans un environnement intégré et distribué.

La technologie choisie pour atteindre les objectifs du projet fut TerraLib. TerraLib est une bibliothèque Open Source développée par l'INPE et distribuée sous licence GNU LGPL.²⁶ TerraLib rend possible l'archivage de données géographiques raster et vecteur dans un certain nombre de Systèmes de Gestion de Base de Données (SGBD), propriétaires ou non, dont PostgreSQL. Il contient aussi des méthodes pour le traitement et l'analyse de données en format image et vecteur. FUNCATE, en contrat avec l'INPE a développé la suite complète de programmes informatiques pour traiter toutes les données et fournir les cartes et les taux annuels de déforestation, dans un environnement complètement libre. Cette suite de programme a été nommée TerraAmazon.

Méthodologie

La déforestation et les incendies en découlant ont lieu en Amazonie durant une courte période : la saison sèche, de juillet à septembre. Après cette saison, la déforestation est normalement impossible, à cause de fortes précipitations. En se basant sur ce fait, le taux annuel de déforestation est calculé sur la période allant du 1^{er} août de l'année précédente au 31 juillet de l'année en cours. Cette dernière date correspond à la fin de la saison sèche dans la plus grande partie de l'Amazonie. Pour obtenir le taux de déforestation, les images acquises durant la période de saison sèche sont analysées. Le taux de déforestation annuel est estimé par interpolation, en considérant que la déforestation se déroule de manière linéaire durant la saison sèche. Par ailleurs, les endroits touchés sont estimés dans les régions couvertes par les nuages, partant du principe que le ratio de déforestation est le même, que les surfaces soient couvertes ou non par des nuages. Des informations détaillées

sur cette méthode peuvent être trouvées sur le site de l'INPE.²⁷

TerraAmazon gère toutes les opérations requises par le projet déforestation, dans un environnement interactif et intégré, à l'aide d'une base de données centralisée. Dans le but de profiter pleinement de TerraAmazon, la région Amazonie est entièrement divisée en cellules, chacune d'entre elles étant manipulée par un seul spécialiste de télédétection à la fois.

Les cellules sont créées en divisant l'emprise du projet avec une grille géographique d'un pas de 0,25 degrés entre chaque ligne. Chaque spécialiste peut verrouiller une ou plusieurs cellules en utilisant un schéma de transactions. L'expert manipule ces cellules avec les outils de traitement d'images et d'édition vectorielle disponibles dans TerraAmazon. Les étapes suivantes sont appliquées à chacune de ces cellules :

1. Import d'une image TIFF
2. Enregistrer l'image avec une image de référence, et sauvegarder les points d'amer.
3. Vérifier l'image en utilisant l'image référence image, et
4. Si l'image n'est pas acceptée, reprendre à l'étape 2
5. Créer les images du sol et les ombrages
6. Si l'image a une couverture nuageuse supérieure à un seuil défini, alors
 - (a) Classifier l'image pour extraire les régions avec des nuages
 - (b) Convertir ces régions en polygones de nuages
7. Segmenter les images d'ombrage et du sol
8. Combiner les vecteurs complémentaires (déforestations des années précédentes, zones non forestières, hydrographie) avec les vecteurs de la segmentation, et ceux de la couverture nuageuse (s'il y en a).
9. Interpréter et éditer les vecteurs combinés pour créer de nouveaux polygones de déforestation et de couverture nuageuse.
10. Analyser les polygones obtenus. S'ils ne sont pas approuvés, reprendre à l'étape 8, et
11. Mettre les résultats à disposition.

Les outils de traitement d'image disponibles sur TerraAmazon sont : import d'images au format TIFF, géoréférencement à partir de points d'ancrage, composition colorée et réhaussement, analyse de mélange de modèles, segmentation et classification.

²⁶Bibliothèque TerraLib : <http://www.terralib.org>

²⁷Informations détaillées disponibles en ligne : <http://www.inpe.br/prodes>

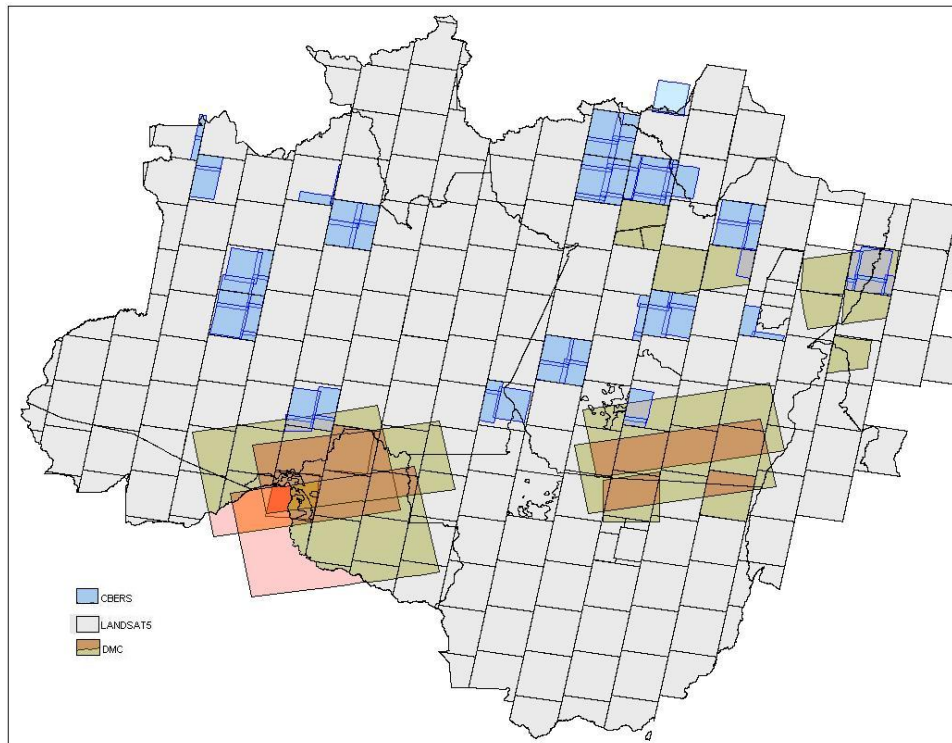


FIG. 1: Les 221 images CBERS, 223 images LANDSAT et 18 images DMC utilisées en 2005

Pour l'édition vectorielle, TerraAmazon comprend : conversions raster/vecteur et vecteur/raster, édition d'objets vectoriels qui prennent en compte l'accrochage des points et la topologie et un ensemble d'opérations sur des éléments géographiques (union, différence, intersection, superposition).

D'autres outils TerraAmazon proposent la visualisation et des procédures de connection et de déconnection à la base de données.

Détails sur la construction de TerraAmazon

TerraAmazon est une application cliente de base de données, développée sur la bibliothèque de composants géographiques TerraLib, utilisant le langage de programmation standard C++ et une interface graphique construite avec la bibliothèque libre d'interface graphique QT.²⁸ TerraAmazon peut être lancé sur un environnement LINUX ou MS-Windows. Toutes les données sont gérées par le SGBD PostgreSQL²⁹, tournant sur serveur LINUX.

²⁸QT Library : <http://trolltech.com>

²⁹SGBD PostgreSQL : <http://www.postgresql.org>

Chacune des cellules de 0,25 degrés est bloquée par le spécialiste de télédétection dans un schéma de transaction à long terme, délimité par les opérations de connections et de déconnection. Le champ de la cellule est utilisé pour sélectionner toutes les représentations géographiques disponibles, dans le but de réduire le nombre d'éléments géographiques, garantissant ainsi la possibilité de manipuler les éléments graphiques. Pour une visualisation rapide, ces éléments sont stockés dans la mémoire cache et indexés par un R-Tree linéaire (Guttman, 1984). La figure 2 montre une région de l'Amazonie avec les bordures de cellules surlignées en vert.

Des opérations de restriction topologiques sont utilisées par TerraAmazon pour l'édition de nouveaux polygones de nuages et d'aires de déforestation. Avant le stockage d'un nouveau polygone d'aire de déforestation ou de nuage dans une base de données, TerraAmazon retire du polygone les anciens polygones créés. La figure 3 montre les étapes pour obtenir un nouveau polygone de nuages.

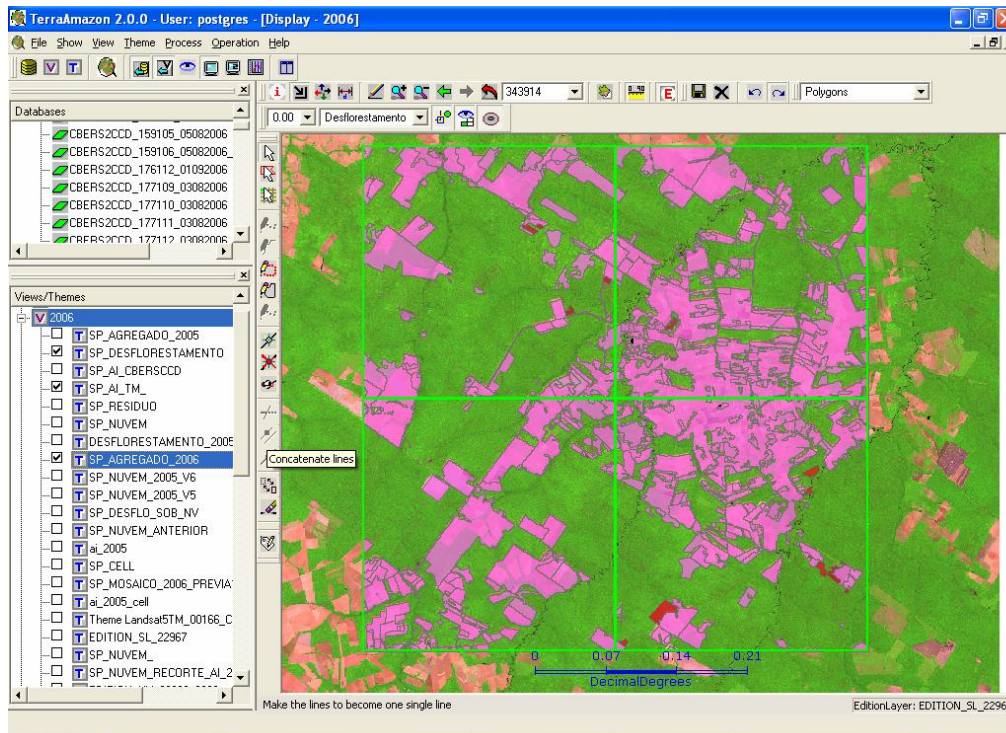


FIG. 2: Cellules (surlignées en vert) sélectionnées pour l'édition dans petite partie de l'Amazonie.

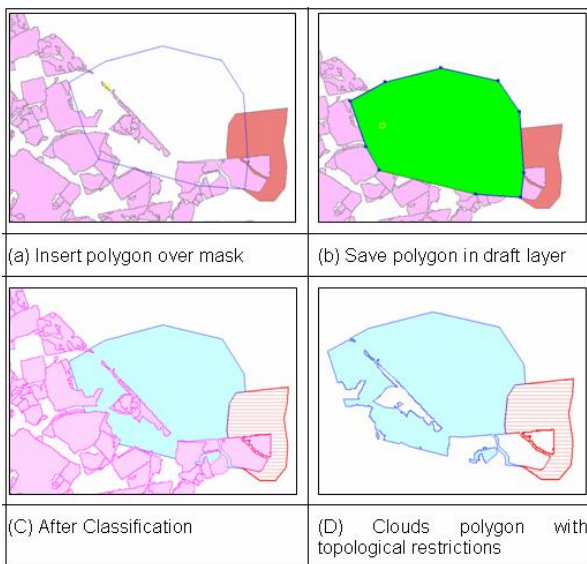


FIG. 3: Restrictions topologiques appliquées à un nouveau polygone de nuage par TerraAmazon. Le polygone stocké dans la base de données est celui représenté en (D).

L'ensemble des outils de TerraAmazon est constitué de :

1. Import d'image TIFF ;
2. Géoréférencement utilisant le modèle polynomial calculé à partir de points d'ancrage ;
3. Réhaussement de l'image et composition colorée ;
4. Analyse par mélange de modèles ; (Shimabukuro, 1987) ; (Shimabukuro ;Smith, 1991)
5. Segmentation d'image avec l'Algorithme de croissance par régions ; (Bins et al., 1996)
6. Étiquetage des régions ;
7. Classification par Clusters ;
8. Classification K-means ;
9. Conversions raster/vecteur et vecteur/raster ;
10. Interface graphique pour édition vecteur avec accrochage des points et contrôle topologique ;
11. Ensemble d'opération sur des éléments graphiques : union, différence, intersection, et superposition ;
12. Procédures de connexion et de déconnexion utilisant les index des cellules.

Un site de distribution sur internet est fourni, basé sur une application PHP tournant sur un serveur Linux fonctionnant avec Apache. L'application Web a été créée comme sur-couche à la bibliothèque TerraLib, en utilisant l'extension TerraPHP.

Les éléments du site internet contiennent :

1. Visualisation continue des données en pleine résolution ;
2. Visualisation d'images, en utilisant la résolution pyramidale ;
3. Export en pleine résolution de portions définies par l'utilisateur ;
4. Accès à des données par WMS - Web Map Service ;
5. Requêtes d'utilisateurs, incluant la déforestation par des municipalités et à l'intérieur de zones protégées ;
6. Classement des parcelles de déforestation par les municipalités ;
7. Indices de déforestation par cellule ;

La figure 4 montre le site Internet³⁰.

Le Projet Déforestation en Quelques Chiffres

Les chiffres suivants montrent l'énorme quantité de travail effectuée par TerraAmazon :

- Pour créer la carte de déforestation pour la période 2004-2005, 221 images CBERS, 223 images LANDSAT et 18 images DMC ont été utilisées, et pour 2005-2006, 70 images CBERS et 211 images LANDSAT.
- Pendant la phase d'interprétation, en 2006, jusqu'à 20 personnes ont pu accéder simultanément au système. Ces utilisateurs ont ajouté 213 693 nouveaux polygones de déforestation, et 595 575 nouveaux polygones de nuages dans la base de données.
- La base de données finale contient 2 380 880 polygones, classifiés en différentes catégories. Ces polygones sont complexes - le plus important possède 69 925 vertex, le nombre moyen de vertex est de 59, et le nombre moyen de trous par polygone est de 7.
- Le volume actuel de données stockées dans la base est de 237Go et inclus des images en pleine résolution, utilisant le schéma de compression multi-résolution pyramidale RLZ.

Conclusion

TerraAmazon a rempli les objectifs imposés par le gouvernement brésilien et fonctionne depuis 2005. Le technologie TerraLib est une alternative pour

créer un système SIG complexe, en utilisant uniquement des logiciels open source, et a rendu faisable l'installation de TerraAmazon. De plus, TerraAmazon s'est montré robuste, facile à gérer et fiable dans un environnement multi-utilisateur et en exploitation intensive.

Remerciements

Ce travail a été soutenu par l'INPE - Institut National pour la Recherche sur l'Espace. Les auteurs souhaitent remercier le groupe de développement et d'utilisateurs de FUNCATE pour leur efforts. Leur collaboration est réellement très appréciée. Nous sommes reconnaissants au Dr. Laercio Massaru Namikawa pour son soutien et ses suggestions.

Bibliographie

Gutmman, A. R-trees a dynamic index structure for spatial searching. *ACM SIGMOD International Conference on Management of Data*, 47-57, 1984.

[Shimabukuro, Y.E. (1987)] Shade images derived from linear mixing models of multispectral measurements of forested areas. Ph. D. Dissertation, Department of Forest and Wood Sciences, Colorado State University, Fort Collins, Colorado. 274p.

[Shimabukuro, Y.E. ; Smith, J.A. (1991)] The least-squares mixing models to generate fraction images derived from remote sensing multispectral data. *IEEE Transactions on Geoscience and Remote Sensing*, v. 29, n. 1, p. 16-20

[Bins et al. (1996)] L. Bins, L. M. Fonseca, G. J. Erthal, F. Mitsuo Ii (1996) Satellite Imagery Segmentation : a Region Growing Approach ; Anais do VIII Simposio Brasileiro de Sensoriamento Remoto : Salvador

[Camara, G., et al. (2000)] TerraLib : Technology in Support of GIS Innovation ; II Brazilian Symposium on Geoinformatics, GeoInfo2000. São Paulo

Vanildes Ribeiro, Ubirajara Freitas, Gilberto Queiroz, Mario Petinatti, Eric Abreu

FUNCATE - Fundação de Ciência, Aplicações e Tecnologia Espaciais

<http://www.funcate.org.br>

[vanildes AT funcate.org.br](mailto:vanildes@funcate.org.br)

³⁰Site internet de diffusion : <http://www3.funcate.org.br/prodes2>

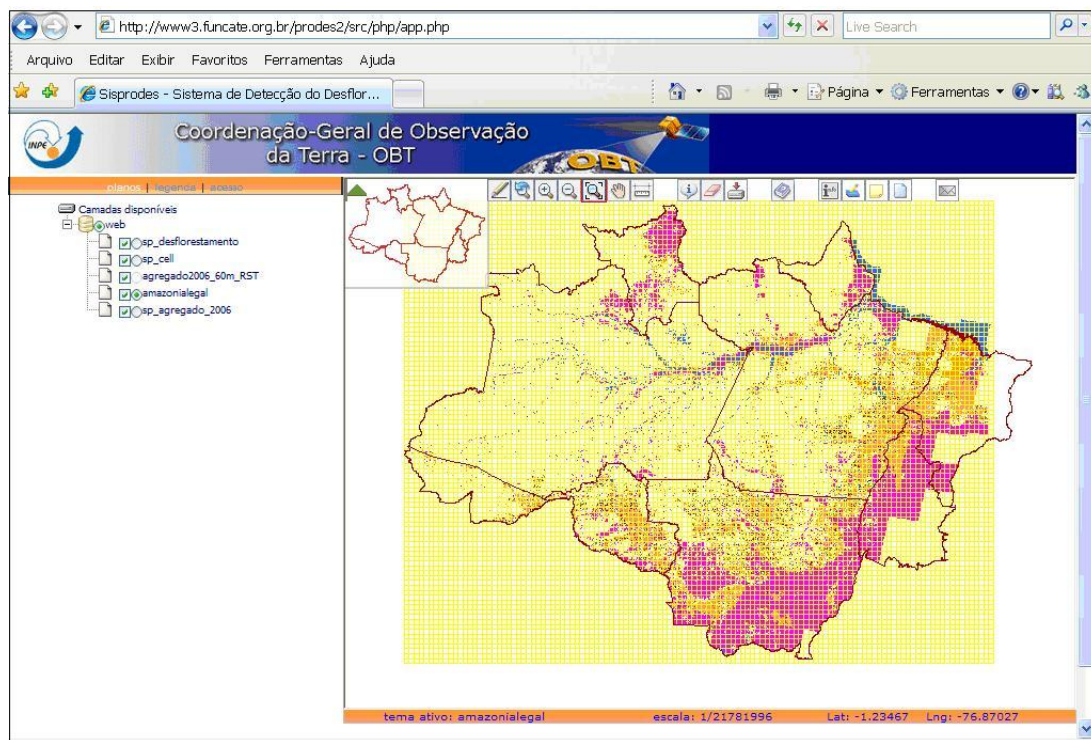


FIG. 4: Site internet TerraAmazon sur les zones de déforestation.

The [Open Source Geospatial Foundation](#), or OSGeo, is a not-for-profit organization whose mission is to support and promote the collaborative development of open geospatial technologies and data. The foundation provides financial, organizational and legal support to the broader open source geospatial community. It also serves as an independent legal entity to which community members can contribute code, funding and other resources, secure in the knowledge that their contributions will be maintained for public benefit. OSGeo also serves as an outreach and advocacy organization for the open source geospatial community, and provides a common forum and shared infrastructure for improving cross-project collaboration.

Publié par l'OSGeo, le Journal de l'OSGeo a pour objectif de publier les résumés des conférences, étude de cas et introduction, et les concepts liés à l'open source et aux logiciels en géomatique.

Équipe éditorial :

- Angus Carr
- Mark Leslie
- Scott Mitchell
- Venkatesh Raghavan
- Micha Silver
- Martin Wegmann

Rédacteur en Chef :

Tyler Mitchell - [tmitchell AT osgeo.org](mailto:tmitchell@osgeo.org)

Remerciements

Divers relecteurs & le project GRASS News

Le *Journal de l'OSGeo* est une publication de la *Fondation OSGeo*. La base de ce journal, le source des styles L^AT_EX 2_ε a été généreusement fourni par le bureau d'édition de GRASS et R.



Ce travail est sous licence Creative Commons Attribution-No Derivative Works 3.0 License. Pour lire une copie de cette licence, visitez : creativecommons.org.



All articles are copyrighted by the respective authors — contact authors directly to request permission to re-use their material. See the OSGeo Journal URL, below, for more information about submitting new articles.

Journal en line : <http://www.osgeo.org/journal>

OSGeo Homepage : <http://www.osgeo.org>

Courrier postal : OSGeo

PO Box 4844, Williams Lake,

British Columbia, Canada, V2G 2V8

Téléphone : +1-250-277-1621



ISSN 1994-1897