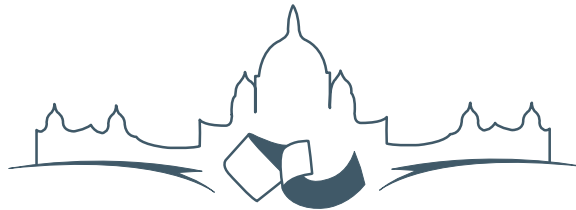

Journal de l'OSGeo

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
-

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>

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

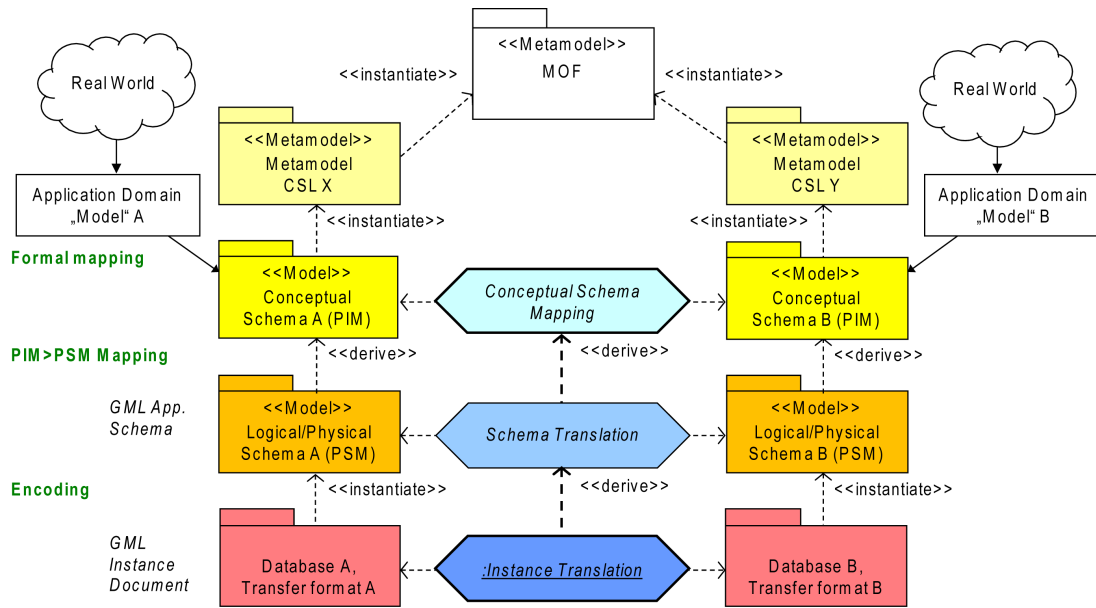


FIG. 1: Model-driven approach and semantic interoperability

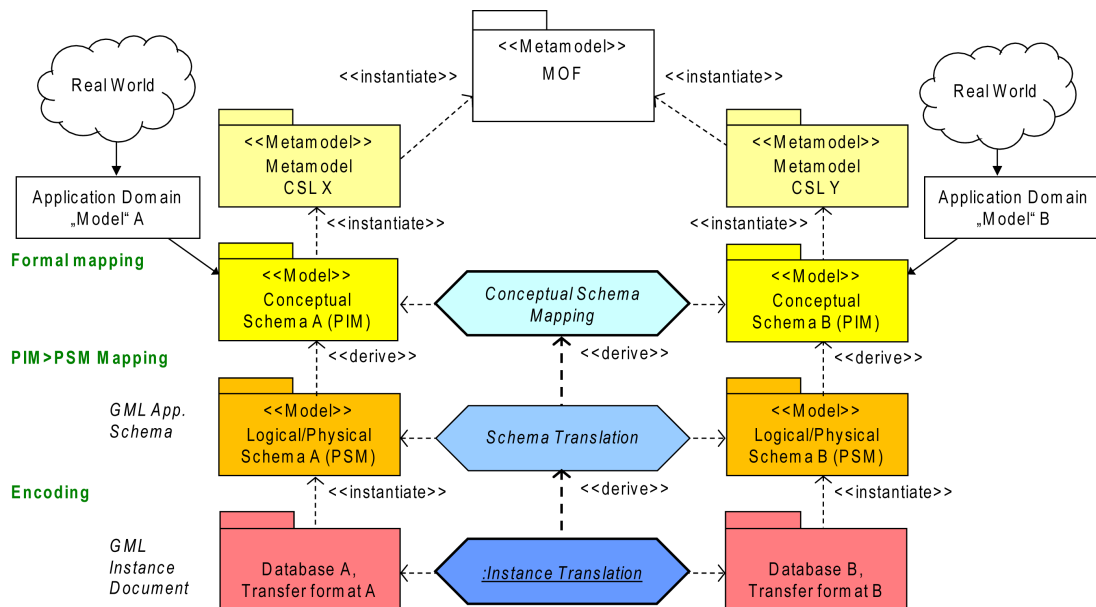


FIG. 2: Approche orientée modèle et interopérabilité sémantique

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) :

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

- 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 la 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 compréhensibles en dehors de scientifiques de l'informatique. Par conséquent, un métamodèle UML, ainsi qu'une syntaxe pour des 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

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".

Éléments 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 différent est une relation géométrie/topologie : un bâtiment *sur* une parcelle

¹⁶Eclipse : <http://www.eclipse.org>

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

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

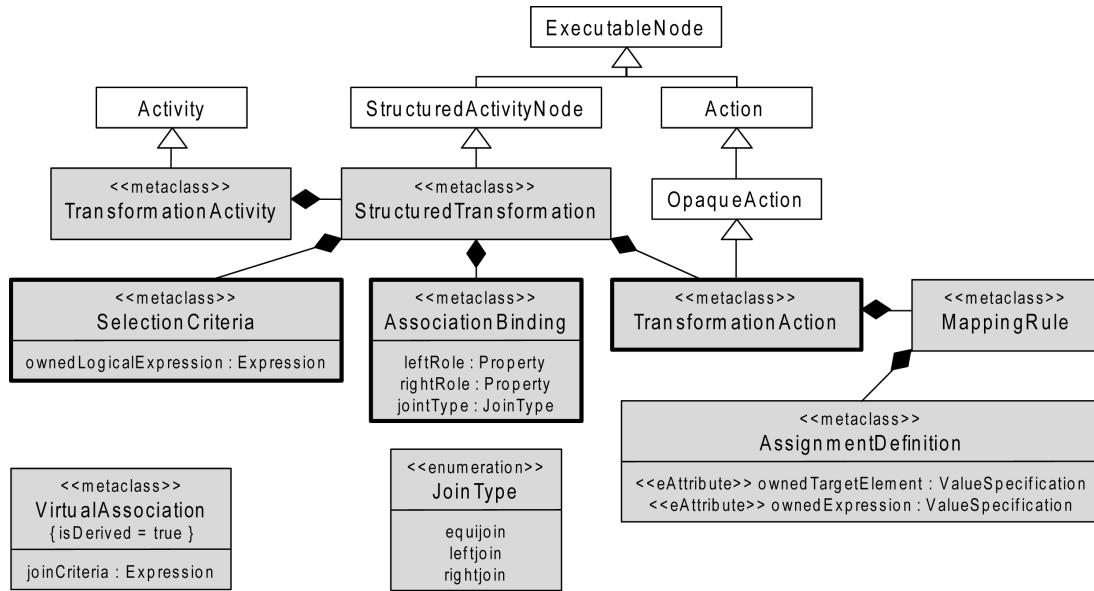


FIG. 3: Eléments du langage UMLT

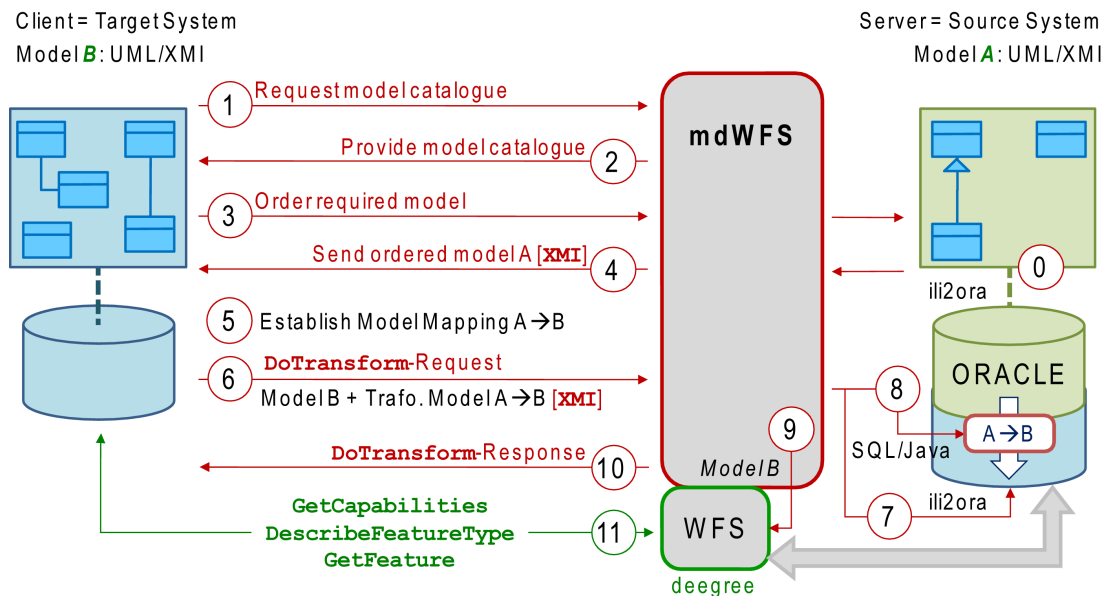


FIG. 4: Test d'installation du prototype

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

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 fournie 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

This PDF article file is a sub-set from the larger
OSGeo Journal. For a complete set of articles
please the Journal web-site at:

<http://osgeo.org/journal>