Using Geosynchronization for Incremental Update of INSPIRE Service Databases

Lassi Lehto, Eero Hietanen and Pekka Latvala
Department of Geoinformatics and Cartography
Finnish Geodetic Institute
Masala, Finland
lassi.lehto@fgi.fi, eero.hietanen@fgi.fi, pekka.latvala@fgi.fi

Abstract—**Opening of governmental geospatial datasets has lead to the creation of multitude of copies of these datasets, used as source information for various value adding services. Obligations set forth by the INSPIRE directive bring new data resources available via service databases, copied from the original sources. Without proper updating procedure these copy datasets become fast out-of-date. The Geosynchronization Service (GSS) proposal of the Open Geospatial Consortium (OGC) is a promising approach for resolving this problem. As an example, GSS is investigated in this paper as a tool for synchronizing INSPIRE datasets and keeping service databases up-to-date in a multi-provider situation.**

Keywords-geosynchronization; incremental update; service database; open data; INSPIRE.

I.   INTRODUCTION

Opening of governmental data resources is one of the most important trends affecting the provision of geospatial services in Europe. As datasets become freely available, many different actors, also in private sector, are able to simply download a dataset, combine it with some application-specific content or another open public sector dataset and create a dedicated online customer service that may have thousands of active users.

Without proper handling of updates the service databases may become quite fast out-of-date. In some cases the governmental datasets are being updated on a weekly or even daily basis. A copied service database can often be left untouched for a long time. This can potentially create a huge data quality problem.

In the context of the INSPIRE directive, European providers of spatial datasets are obliged to develop standardized services for content themes that in many cases involve multiple separate institutions [1][2]. In this situation, it is usually necessary to set up separate service databases, in which the content is integrated, harmonized and brought to the specified common schema. Again, the problem of data decay over time is obvious. The services become fast out-of-date, unless the service databases are regularly updated from the original sources.

A full data download and a complete replacement of the service database contents is often unpractical, due to vast data volumes, or local additions made in the copy dataset. Thus partial, feature level update of the service database becomes a necessity. The principles of incremental update have been investigated over the years in various research projects. The latest action in this area is the work of Open Geospatial Consortium (OGC) to specify a service type called Geosynchronization Service (GSS) that would allow incremental update of data resources over the network [3]. In this paper, the GSS approach is tested in the context of an INSPIRE service development project, in which the service database is composed of content coming from two separate data providers. The rest of the paper is organized as follows. Section II discusses the past work done in the area of incremental update and synchronization of geospatial data. Section III describes the principles of the OGC's GSS specification. Section IV describes the context of the case study and the implementations developed in it. Section V concludes the paper with discussion on the lessons learnt and prospects for the future work.

II.   PREVIOUS WORK

Incremental update has long been studied as a mechanism for synchronizing geospatial datasets across organizational borders [4][5]. Mostly these studies did not consider network-based solutions. Milanović et al. describe an example of an early service-oriented implementation [6]. In this experiment general OGC service standards were used to synchronize datasets among participating data providers. The examples, in which GSS specification has been adopted as a solution for data synchronization include distributed maintenance of the National Hydrography Dataset in Indiana [7] and New Hampshire [8]. A few software implementations have been developed based on the GSS specification, like CarbonCloud Sync of Carbon Project Inc. [9], userSmarts of ImageMatters [10] and GeoGig of Boundless [11].

III.   OGC'S GEOSYNCHRONIZATION SERVICE

A.   Principles

The Geosynchronization specification of OGC is based on the idea of collaborative capture of geographic information, in which one party acts as a Publisher proposing changes to be made to the target database [12]. Another important role is that of a Reviewer, who checks the proposed changes and either accepts or rejects them, based on some well-established quality criteria. The third role is a Subscriber, a party that is interested in receiving the authorized changes made to the target database to keep a copied dataset up-to-date, i.e. synchronized with the original.

These functionalities are formulated as a set of Web Service interfaces that support the needs of the identified actor roles. The main workflow of the Geosynchronization Service is depicted in the Figure 1.
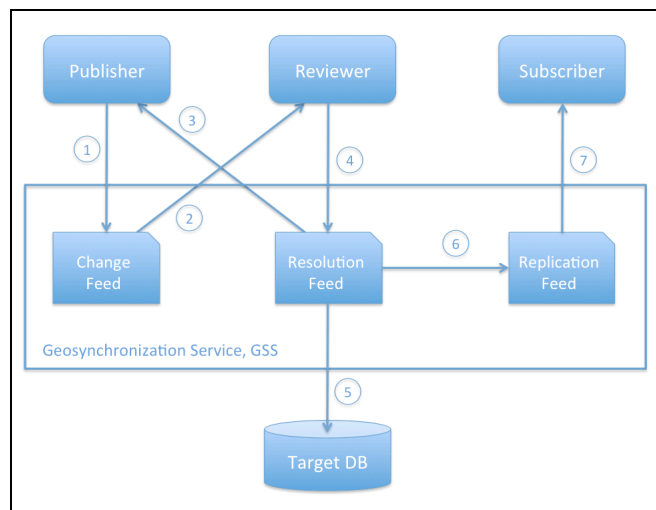

Figure 1. The main Geosynchronization Service workflow.

The Geosynchronization workflow commences with a Publisher proposing a change to be made in the target database (action number 1). This proposal is fed into the ATOM stream called the Change Feed. Looking into the change proposal from the Change Feed the Reviewer makes a decision, whether the change should be accepted or rejected (2). Thus the Reviewer represents the organization having responsibility and authority on the target database. If a given change proposal gets accepted by the Reviewer, information on the decision is sent to the respective Publisher via the Resolution Feed (3), the change is committed to the database (5) and the corresponding record of the change is created to the Replication Feed (6). If the change is rejected, only note of this decision is returned to the Publisher and no other action is taken. Finally, if a third party is interested in the changes happening in the target database, that party can register as a Subscriber to receive the changes from the Replication Feed (7).

All the important information in the Geosynchronization Service is conveyed through standard ATOM feeds [13]. These include Change Feed to manage the proposed updates, Resolution Feed to convey information on the validation decisions, and Replication Feed for synchronizing a remote copy of the dataset with the master database managed within the Geosynchronization Service. This mechanism facilitates a standards-based approach for retrieving only items that have been created since the previous request. The change requests are encoded as Web Feature Service (WFS) Transaction operations [14].

Main use cases of the Geosynchronization Service include maintenance of multi-resolution datasets by organizations on different levels of administration hierarchy (for instance municipality proposing updates on a dataset managed by a state level authority) and crowd-sourced

provision of updates to a governmental dataset. In both cases an outside actor (Publisher) proposes changes to be made on a dataset managed by an authority (Reviewer) that has to validate the changes before they are committed to the database. A further third party can then act as a Subscriber to keep a copied instance of the dataset up-to-date by checking the Replication Feed regularly and applying the changes to the copy dataset. The Replication Feed can be queried with full filtering capabilities, so that the Subscriber can only receive changes that are relevant for his purposes. If deemed appropriate, a GSS can be configured to automatically commit all changes directly to the Subscriber's database. A Subscriber can also opt for receiving only change proposals with information of their validity (Resolution Feed).

B. Status of the Specification

The Geosynchronization Service specification has been initially drafted in the context of the OGC Interoperability Program as a Canadian Geospatial Data Infrastructure (CGDI) pilot in 2007 - 2008. Since then the specification has been amended and refined in other OGC and local Canadian programs. The candidate standard was published as OGC Engineering Report in 2011. Since 2011 the GSS specification has not advanced in the OGC process and the official standard is not yet available.

IV. CASE: INSPIRE SERVICES IN A MULTI-PROVIDER SITUATION

The case implementation of the GSS specification described in this paper has been developed in the context of the European INSPIRE directive and the requirement it sets for EU member states to establish standardized services on geospatial content, as defined in common data specifications and content schemas. In many cases, the content specified in one single INSPIRE theme, like Hydrography (HY), involves datasets maintained by many state level organizations. The commonly agreed European level content schemas also often significantly differ from the data structures used in national datasets. All this necessitates maintenance of separate service databases, in which content is integrated from different data provider organizations and the data structures are designed to support INSPIRE schemas. This environment represents an ideal use case for a GSS piloting. On one hand, the data provider organizations involved need to keep their datasets synchronized, otherwise the quality of the resulting service output would be severely degraded. On the other hand, the service database has to be kept up-to-date by replicating to the service database the changes made in the original datasets.

In the case implementation discussed here, the organizations involved in the provision of INSPIRE Hydrography theme are the National Land Survey of Finland (NLS), the official national mapping authority, and the Finnish Environmental Institute (SYKE), a governmental organization responsible for supporting the state activities related to the protection of environment. The mapping authority maintains an up-to-date geometric representation of the physical hydrologic features and the environmental institute manages a rich record of data values describing the

status of those physical features. SYKE also maintains a network representation of the hydrography system, so there is a need to synchronize spatial geometries between the two organizations too.

The test implementation was designed according to the principles of the GSS, predominantly to synchronize the hydrography network geometries managed by SYKE, based on the frequently updated geometries of physical waters, maintained by NLS (see Figure 2). To facilitate the flow of updates from NLS to SYKE, a prototype GSS implementation was developed. As the data updates in this case can be seen as authoritative changes, the review/validation step was not included in the prototype and is thus not further discussed in the paper.
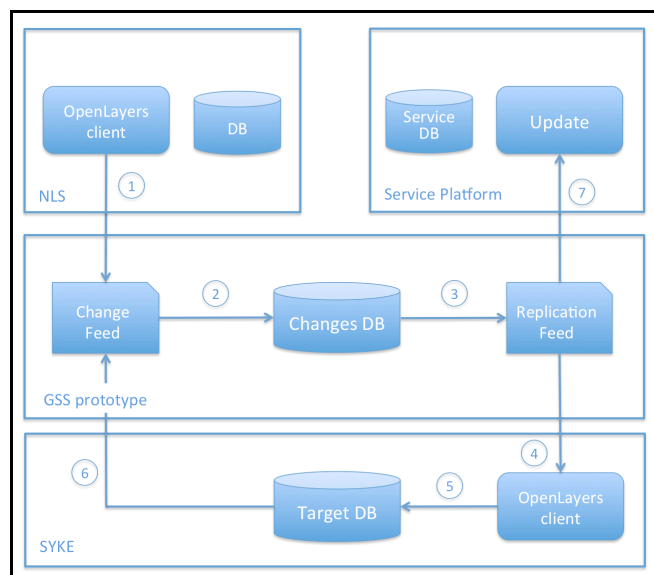

Figure 2.   The GSS prototype implementation architecture.

The GSS prototype uses PostGIS/PostgreSQL database (Changes DB) for storing the proposed changes. In the demonstration environment, the changes are produced using OpenLayers-based client application that formulates the changes as WFS-T messages (1). Once the GSS prototype receives a WFS-T message, it parses the message extracting all relevant information, and stores everything to the Changes database (2). This functionality is implemented as a Java Servlet. The Replication Feed is produced by another Java Servlet that, once receiving a replication request, processes it, retrieves the needed pieces of information from the Changes database and then formulates the response in the form of an ATOM feed (3).

The Changes database was designed to be simple. One single table contains all data items related to proposed changes. A record in the table corresponds to one change proposed to an individual feature. The columns of this database table are described in Table I.

The fields stored in the Changes database table are carefully selected to support efficient serving of the Replication Feed queries. The WFS-T DELETE operation represents a special challenge, as it does not contain the

attributes of the deleted feature, but only its identification. As the approach requires the geometry of the affected feature before the change to be stored into the change table, this information has to be acquired elsewhere. In the prototype implementation this is done by requesting the feature information from the target database before the change in question is applied to it. To improve storage efficiency, the geometry of the feature in the Transaction column could also be removed and dynamically generated from the Geom column when needed.

TABLE I.   COLUMNS OF THE CHANGES DATABASE TABLE.

Column	Data type	Description
GID	integer	Unique identifier of the feature
Type	string	Name of the feature class
ChangeTime	date/time	Timestamp of the change
Operation	string	INSERT/UPDATE/DELETE
Transaction	string	The change, encoded as a WFS-T transaction message
Geom	geometry	Geometry of the feature after the change. If operation is DELETE, then before the change.
EntryID	integer	Unique identifier of the change
Author	string	Publisher of the change

As the Changes database's Transaction column contains a complete representation of the affected feature after the change in every UPDATE operation, the Changes database also serves as a fully versioned data store. All known versions of a given feature can be requested from the GSS Replication Feed using the FEATUREID parameter. If the initial creation of the feature is also recorded as an INSERT operation, then the full history of the feature is available from the GSS.

The Replication Feed can be queried from the service by GetEntries operation. In this query the calling application can limit the result set by the following predicates: STARTTIME/ENDTIME (limits for the change timestamp), BBOX/GEOM (spatial extent inside which the changes are requested, as a rectangular bounding box or free form polygon), FEATUREID (returns changes related to the given feature or features), ENTRYID (returns the indicated change). The resulting Replication Feed is formatted as an XML-encoded ATOM feed, following the principles established in the GSS specification.

To facilitate illustrative visualization of the changes in a map display, an extra GetEntryIdObjectGeometry operation has been added to the prototype as an extension for the GSS specification. This enables geometries of an identified change to be requested from the Changes database. As the original change messages are stored in a separate column (Transaction), the Replication Feed can be quite efficiently generated.

Two OpenLayers-based map viewer user interfaces have been developed to demonstrate the functionality of the GSS.

One is used to initiate changes. The source dataset is queried via a WFS interface from the NLS DB and the transactions resulting from the edits done on the features in the map window are redirected to the prototype GSS instance. The other map viewer is used to visualize features from the SYKE's target database that is being synchronized with the original NLS database. By the client application a user can visualize the contents of the target database, together with the changes requested from the Changes database via the GSS instance (4). The user can then interactively select the changes that seem appropriate for the target database. Finally the selected changes are applied to the database through the WFS-T interface (5). The edits done on the SYKE's target database will again trigger change requests in the GSS's Change Feed, relating to the feature types maintained by SYKE (6)

In a similar fashion, the GSS Replication Feed can be accessed to keep a remote service database up-to-date (7). This could be a completely automated process as all the changes are authoritative and can be copied without further consideration to the service database.

V. CONCLUSIONS

The prototype GSS implementation described in the paper has been developed in the context of an INSPIRE Download Service piloting project. The experiences gained in the work point out that the GSS approach can be utilized to synchronize two databases for achieving consistent representation of geospatial datasets and thus for facilitating the integration of INSPIRE content in multi-provider environment. At the same time the GSS serves as a tool for keeping the copied service database up-to-date.

The GSS developed in the project is essentially based on standardized approaches. The update information is transmitted and stored as standard WFS-T transactions. GSS response messages are formulated as ATOM-conformant feeds. OGC-compliant service interfaces are used to access the data stores, both to retrieve and to modify the contents of the backend databases. The selected approach facilitates integration and usage of the proposed system with various off-the-shelf commercial and open source solutions.

An interactive process has been developed for selecting relevant updates using the capable filtering options of the GSS query interface. A map interface-based tool is used for fine-tuning the set of updates to be applied to the target database. These functions enable highly varied application needs to be taken into account in the update process.

The experiences gained in developing the GSS implementation confirm that the approach taken in the OGC's Geosynchronization activity is usable and can be adopted in the context of multi-provider INSPIRE Download Service provision.

Future developments could involve for instance adding of schema transformation capabilities to the GSS to enable synchronization among heterogeneous schema structures. Another need for further investigation is the scalability of the proposed approach in case of larger data volumes and various types of data, and the methods, by which this could be ensured.

REFERENCES

[1] European Commission, "INSPIRE Directive", 2007. at: http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2007:108:0001:0014:EN:PDF [accessed: 2014-10-09].

[2] European Commission, "COMMISSION REGULATION (EC) No 976/2009 of 19 October 2009 implementing Directive 2007/2/EC of the European Parliament and of the Council as regards the Network Services", 2009. at: http://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:02009R0976- 20101228&from=EN. [accessed: 2014-10-09].

[3] Open Geospatial Consortium, Specifications Working Group on GeoSynchronization Home Page. at: http://www.opengeospatial.org/projects/groups/geosyncswg [accessed: 2014-10-09].

[4] M. Scheu, W. Effenberg, and I. P. Williamson, "Incremental update and upgrade of spatial data", 2000. Zeitschrift fur Vermessungswesen, 125(4), 2000, pp. 115-120.

[5] A. Cooper and A. Peled, "Incremental Updating and Versioning", 2001. 20th International Cartographic Conference, Beijing, Vol 4, 2001, pp. 2804-2809.

[6] A. Milanović, J. Guimet, E. Rodellas, and M. A. Bolívar, "Interorganizational Geo-Synchronization Using Open Geospatial Consortium's (OGC) Technologies to Share and Harmonize Data in Catalonia", INSPIRE Conference 2010, Krakow, at: http://geoportal.cat/geoportal/eng/documents/articles/Ponencia_INSPIRE_2010.pdf [accessed: 2014-10-13].

[7] J. Sparks, "Indiana High & Local-Resolution NHD Update Geosynchronization". Framework Data Exchange through Automated Geosynchronization , Final Report, 2011. at: https://www.fgdc.gov/grants/2010CAP/InterimFinalReports/242-10-2-IN-FinalReport.pdf [accessed: 2014-10-13].

[8] R. Chormann and J. Harrison, "New Hampshire NHD GeoSynchronization Network", 2010 NSDI CAP Final Report, 2012, at: https://www.fgdc.gov/grants/2010CAP/InterimFinalReports/238-10-2-NH-FinalReport.pdf [accessed: 2014-10-10].

[9] N. Goldstein, "GeoSynchronization – Synchronization of Geo-Databases through Interoperability", 2010. at: http://proceedings.esri.com/library/userconf/feduc10/papers/user/goldstein.pdf [acessed: 2014-10-10].

[10] ImageMatters, userSmarts Home Page, at: http://www.imagemattersllc.com/products/usersmarts.php [accessed: 2014-10-13].

[11] GeoGig Home page, at: http://geogig.org [accessed 2014-10-13].

[12] Open Geospatial Consortium, "OWS 7 Engineering Report – Geosynchronization Service", 2011. at: http://portal.opengeospatial.org/files/?artifact_id=39476 [accessed: 2014-10-09].

[13] M. Nottingham and R. Sayre [eds.], "The Atom Syndication Format", 2005. at: http://tools.ietf.org/html/rfc4287 [accessed: 2014-10-09].

[14] P. A. Vretanos [ed.], "OpenGIS Web Feature Service 2.0 Interface Standard", 2011. at: http://portal.opengeospatial.org/files/?artifact_id=39967 [accessed: 2014-10-09].