Uncertain geoinformation representation and reasoning: a use case in LBS integration

Roula Karam, Franck Favetta, Robert LAURINI
LIRIS CNRS UMR 5205, INSA de Lyon
Université de Lyon
F-69621 Villeurbanne, Lyon, France
{roula.karam, franck.favetta, Robert.laurini}@insa-lyon.fr

Rima Kilany CHAMOUN
ESIB, Faculté d'Ingénierie
Université Saint. Joseph
B.P:11-514, Beirut, Lebanon
rima.kilany@usj.edu.lb

Abstract— **Due to the fact that mobile devices are in widespread use, many applications including Location Based Services (LBS) had been involved to deliver relevant information to customers anywhere at any time and thus based on their profile and geographical position. However, with the increasing number of heterogeneous databases, many problems may arise related to 1) the interoperability of geographical databases, 2) the integration of geospatial data / metadata of the same location service, provided differently by many operators and 3) the uncertain positioning results due to GPS/Radio coverage technologies, geo coding conversion functions and mobile devices' limitations. In this paper, the 'uncertainty' issues on geographic, place names and semantic details for homologous objects towards a complete map conflation were solved by our MPLoM framework (Location Integration) and our Visual Domain Ontology application (Cartographic Integration) with an extension of Web Ontology Language named CartOWL.**

Keywords: GIS, LBS, Cartography, Ontology, OWL, CartOWL.

I. INTRODUCTION

Nowadays, the co-existence of many and heterogeneous geographical databases (GDB) covering the same areas, rise the need to study how these data/metadata may be integrated together. In the general context of geographical databases' interoperability, our concerns in this article are to study the feasibility of location based services' (LBS) integration, proposed by several providers, and their limitations especially in terms of precision for location and cartographic conflation.

We mean by LBS, a particular type of applications in which services are device-oriented like tracking or person-oriented like navigation and information entertainment. Any LBS application implies user's profile and his geographical location.

User's profile is saved as preferences in the system middleware database but the geographical location is captured by its GPS built in device via satellite system or his current cell via radio mobile coverage system. In both cases, uncertainty is presented due to the inaccurate positioning systems. Since May 2000, the GPS service provides commercial users with an accuracy of 5 to 15 meters due to some limitations that are beyond our control while other sources of error can be controlled via differential GPS technique or more sophisticated costly implementations that are used to improve the accuracy.

For example, in order to obtain a GPS position reading, many limitations are encountered such as:

- One needs to occupy the base point and he cannot get there (highway with high traffic, etc.).
- The elevation readings from GPS receivers are not very accurate.
- The signals transmitted from the satellites have to pass through ionosphere before reaching the GPS receiver and are delayed.
- The satellite orbits are affected by gravitational pulls, which cause disturbance in their orbits and create errors in the position.
- The multipath fading is applied to the placement of the receiver and the faulty clock in the GPS receivers themselves can cause errors.

Besides, if the position of the user via GPS is not totally correct or even worse via radio system where the coverage unit is the correspondent cell, the list of services within the geographical zone is uncertain and we may encounter the same object listed twice at 50 meters of difference. This could be related to the different positioning rules applied by many providers thus geocoding conversion functions from postal address to longitude/latitude address, etc. As a matter of fact, not only user's position is uncertain, the geographical positions of location based services may include errors as well in the process of GDB integration.

A use case scenario is presented below to describe clearly other uncertain issues related to LBS integration.

Let us take the application to look at the nearest restaurant in your area with the navigation instructions to get there.

First of all, one might encounter the answer of an Italian restaurant listed by two different providers, not exactly located at the same place (50 meters of difference). The same Italian restaurant is named "Carlo's Pizzeria" in the first one and "Da Carlo Trattoria" in the second one, visualized by different cartographic symbols. The goal is to consider them as the same object. [Fig. 1]

Many techniques should be undertaken to solve all the uncertainty issues and ensure the integration of homologous objects among all the heterogeneous ones to be overlaid on a unique base map.

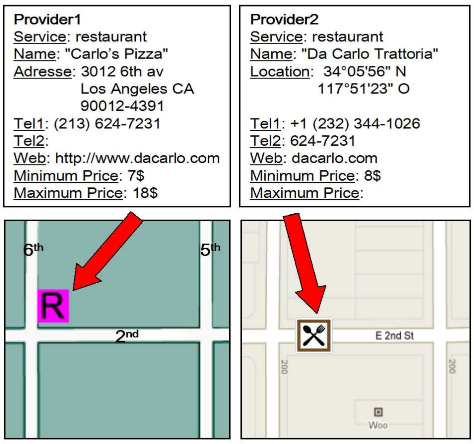


Fig.1. Example of the same LBS restaurant from two providers (candidates for integration)

Section II will list all the limitations and uncertainty problems related to LBS, location and cartographic' integrations. Section III will detail all the solutions that are implemented in our MPLoM (Multi Providers LBS on Mobile) framework [1] and proposed via our building/matching application with OWL extension (CartOWL) [2]; Finally, Section IV will recap and highlight some perspectives for future work.

II. Uncertainty issues in LBS integration

In the scope of geographical databases (GDB) interoperability, we can define five different types of conflicts: [20, 21]

- Conflicts related to the data source used to constitute the geographical databases (e.g. satellite images, raster, etc.)
- Conflicts based on the models and metadata
- Conflicts of class and attribute definition
- Conflicts of data measures
- Conflict of positioning

Yet users need to fusion various information coming from these databases. In order to integrate databases, redundancy and inconsistency between data should be identified. Many steps are required to finalize the databases integration. In particular, we will focus on solving the conflict of data/metadata of LBS among many providers at the application layer. We can distinguish three types of integration related to the location of objects: geographic, place names and semantic details.

1. Geographic integration

It consists of matching the geographical components by their position and representation. Same services could be retrieved as points (0D) from the first provider or line (1D) / polyline (2D) / volume (3D) from the second one. Besides, differences in location due to GPS tracking device precision and the conflict in positioning rules would interfere against a smooth integration (e.g. postal address v/s longitude and latitude).

2. Place Name integration

As per the use case scenario described in the introduction, the same restaurant could be saved as "Carlo's Pizzeria" in the first GDB and "Da Carlo' Trattoria" in the second one, even though they are belonging to the same restaurant. Place names' differences are mainly related to GDB lack of "real time" updates for their data/metadata. This may cause duplication thus uncertain representation of the geographical information on the mobile screen.

3. Semantic integration

Details about the same restaurant (Telephone, Website, etc.) could be different from one provider to another as described in Fig.1. This was due to lack of updates and common agreements on the rules for saving email addresses, websites URL, etc. This may cause duplication of unnecessary or inaccurate data/metadata on the mobile screen.

4. Cartographic symbols integration

Through a conventional point of view, the use of many LBS providers will imply several maps, one for each, with its specific legend (visual attributes: icon, color, texture, etc.), where as a cartographic integration will generate a unique base-map whose components will come from various LBS providers.

Fig.2 shows how the same service is represented differently from three different providers. Which cartographic symbol should be prioritized for the same integrated service? This can cause lot of ambiguity based on symbol's selection, user's profile, geographical zone, graphical semiology constraints, the background map used and finally the generalization and the adaptation of maps on the device due to its limitations in memory, resources and display.

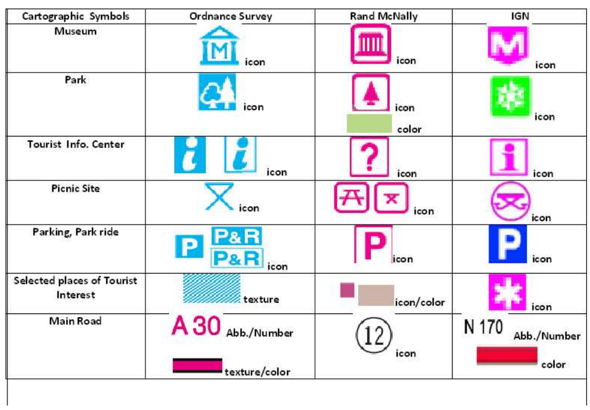


Fig.2 Excerpts from three different legends: Ordnance Survey, Rand McNally and IGN

III. IMPLEMENTED SOLUTIONS

The platform MPLoM is developed to test the feasibility of the location and map symbols integrations into a unique visual portal on mobile devices.

Phase 1 covers the location integration from two providers offering pull services (hotel and restaurant finders) and push service (weather forecast) while phase 2 covers the cartographic integration especially with other suggestions related to web application and geo web services standards for multi providers' interoperability.[1]

In MPLoM, the pull services which are the nearest hotels and restaurants are visualized on a 2D background Google map and the components are overlaid as Google markers(R for restaurants and H for hotels); the details for each clickable restaurant marker or hotel are presented textually on the mobile device. A user interface is created to get all the preferences of the clients (e.g. name, age, nationality, major, email, credit card, language, etc.) and save them into a middleware admin database. Thus, the client request with all the needed parameters will be forwarded via servlet to the concerned tables in the providers' databases. Both providers DB are created In Postgresql with PostGIS feature for spatial usage. Each request will be then subdivided in two sub requests, one for each provider in order to collect the available data. Output data for each requested service type, will be saved as GML(Geographical Markup Language) file, precisely in cGML format (compact for mobile device). All Cgml output files, one per provider, will be collected in the middleware admin database. XQuery is used to parse these cGML files, integrate the details of homologous objects and keep heterogeneous ones towards a unified Cgml file response. The platform source code can easily match each Cgml tag with the correspondent field on the S60 Nokia emulator mobile screen with LBS middleware. Push service like weather forecast is integrated as web service and represented as textual output on the mobile screen.

Fig.3. Nearest LBS Restaurants on Nokia Emulator with integration

An improvement of phase 1 is currently running to ensure the cartographic integration in order to visualize on the screen a unique base map whose components will come from the various providers. Visual ontology of concepts with an extension of OWL standard (Web Ontology Language) will be proposed to include all the visual attributes of each cartographic symbol. Each provider will have its own dictionary or visual ontology of icons to describe its services.

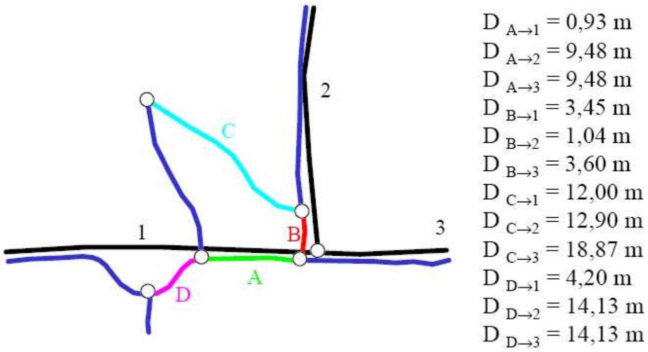
The local visual ontology of each provider will be created via our building/matching application. In order to facilitate the automatic build of local ontologies and their integration towards domain reference ontology, we propose CartOWL (Cartographic OWL) as an extension to the Web Ontology Language OWL. Building visual ontologies would become easier by generating the corresponding CartOWL file through our application. The full prototype would be able to parse the CartOWL output files, unify them into one file towards one reference knowledge base (domain ontology) so that we can ensure map conflation result on a mobile device for example. [2]

The cartographic symbols of the spatial ontologies are detailed below:

- an icon, described by its URL, e.g., http://www.example.net/TouristInfoCenter.png
- a color, described by its RGB model value, e.g., #FF8000
- a texture, described by its URL, e.g., http://www.example.org/ParkTexture.png
- an abbreviation, e.g., "info"
- a number, e.g., "A 30"
- a font, described by its name, e.g., "Times"

To ensure that two objects for the same location based service, candidates for integration, are certainly homogenous and should be visualized once on the mobile screen, many solutions were adapted:

3.1 In our MPLoM framework, to decide if two objects are the same, we choose a threshold of five meters between the candidates. For the integration of two punctual objects, the Euclidian distance dE is used. To integrate two linear objects, three types of distances could be used (average distance, Hausdorff distance, and Frechet distance). So as far as the distance between object 1 and object 2 is less than a threshold of 5 m, we can suggest that the two objects are homologous. However, the choice of the threshold is very important. A large threshold (9 m distance) leads to many doubtful candidates for integration and a small one (3 m) can neglect many solutions. To reach the good compromise, Stricher technique is used by eliminating with successive thresholds, the doubtful points. For that reason, the threshold of 5 meters was adopted.



15 m	A, (1,2,3)	B, (1,2,3)	C, (1,2)
12 m	A, (1,2,3)	B, (1,2,3)	C, ()
9 m	A, (1)	B, (1,2,3)	C, ()
6 m	A, (1)	B, (1,2,3)	C, ()
3 m	A, (1)	B, (2)	C, ()

Fig.4. Choice of adequate threshold (Stricher technique)

3.2 For the place names' integration, the fusion technique uses the Levenshtein distance dL to compare the place name (String of characters) of two objects from two different providers. This distance will increase if the number of differences between characters increases as well thus increasing the probability of heterogeneity.

3.3 Semantic' integration between these two objects is related to their metadata/data. To avoid duplication of the service details from two different providers, a matching table had been used in our framework MPLoM and a semantic ontology-driven approach had been implemented via Protégé [14]. For example, if a navigator wants to know what restaurants can offer "Hamburger", the platform should list all the restaurants of American type or Fast food.

3.4 We can assume that location integration ambiguities had been partially solved by the above solutions. The final decision for homologous objects is relying on the output result of the belief function with Dempster operator [5, 6]. The Belief theory is developed in three steps [18]:

- The first step consists on initializing the belief weights for each candidate and for each source
- The second step consists on combining the masses per candidate (geographic, place names and semantic details)
- The third step is based on the combination of all results from step 2 towards final candidates' integration.

Geographic, place names and semantic details reasoning are assigned a certain weight, reflecting the degree of homogeneity of the candidates for integration. Dempster operator [5, 6] will combine the three different weights (the first mass m for the geographic decision, the second mass for the place names decision and the third mass for the semantic details decision). As far as the final weight which is the sum of the three weights is high, the probability to consider both objects as homologous will be higher. [3] The order among the three decisions' types is not obligatory.

3.5 On the other side, cartographic symbols ambiguities are solved by a new building/matching application Fig.5 with an extension of OWL named CartOWL. Our application is responsible to build local visual ontology for each provider by inserting their visual service attributes based on their legends.

As OWL can handle only textual concepts, it needs to be revised in order to describe visual aspects as well. In the <Class> tag, the human readable textual representation of the concerned class is done with the <Label> tag.

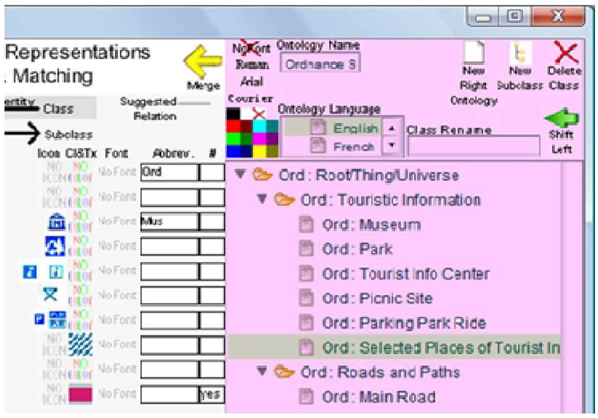


Fig.5. Ontology Building/Matching Application

We suggest to extend this <Label> with new attributes or to add a new <Symbol> tag that includes two parameters:

-the symbol type: the parameter value may be "icon URL" or "color" or "texture" or "abbreviation" or "number" or "font".

-the symbol value: the parameter value may be one of the following string values: the Icon URL, the RGB color, the texture URL, the abbreviation's text, the presence of a number indicated by the Boolean values "yes" or "no", or the font name. Below is an example where we define the class "Tourist Information Center" and its three attached symbols; an icon , a color and an abbreviation> The first part of the example shows the DTD definition of the new <symbol> tag.

CartOWL output file including tags for visual concepts will be generated by the application as shown below:

```
<!DOCTYPE rdf:RDF [
<!ELEMENT cartowl:Symbol EMPTY >
<!ATTLIST cartowl:Symbol cartowl:symbolType
(iconURL|color|texture|abbreviation|number|font)#REQUIRED
cartowl:symbolValue CDATA  #REQUIRED
xmlns:cartowl CDATA  #FIXED
"http://www.example.net/CartOWL.owl# ... >
]>
<rdf:RDF ... ..
```

```
xmlns:cartowl="http://www.example.net/CartOWL.owl#
" >

...

<owl:Class rdf:ID="Tourist Information Center">

<cartowl:Symbol cartowl:symbolType="iconURL"
cartowl:symbolValue="http://www.example.net/TouristIn
foCenter.png" />

<cartowl:Symbol cartowl:symbolType="color"
cartowl:symbolValue="FF8000" />

<cartowl:Symbol cartowl:symbolType="abbreviation"
cartowl:symbolValue="info" />

</owl:Class>
```

The full prototype will be able to parse the CartOWL output files and align them towards a unified one for the visual domain ontology. In that case, we can ensure map conflation results on mobile devices by referring to this file.

3.6 Belief functions must be applied as well through CartOWL in order to achieve the best compromise between the domain ontology and other constraints that may interfere such as the user's profile (nationality, map preference, age, etc.), the scope of the geographical zone, the graphical semiology rules and color contrasts v/s visibility, the device limitations and the need for generalization, adaptation and dynamic maps, etc. So, in order to prioritize visual attributes from one provider among others, highest masses will be assigned to them as per the belief theory via the CartOWL tags [4]

IV. Conclusions and Perspectives

In this position paper, we have presented our framework MPLoM and our building/matching application with visual ontology approach and CartOWL language. Belief theory/Dempster operator is applied to location and cartographic ontology alignments to solve most of the ambiguities in geographic, place names, semantic details and OWL concepts. Future work with Belief function can carry about the properties of the visual symbol tags and the matching relations in CartOWL. Other solutions should be found out to overcome the limitations in GPS and mobile technologies.

References

[1] R. Karam, F.Favetta, R.Laurini and ,R.Kilany ,"Integration of Similar Location Based Services Proposed by Several Providers", in press, (2010).

[2] R. Karam, F.Favetta, R.Laurini and ,R.Kilany, "Concepts Are Not Only Textual: Towards an Ontology for Cartographic Symbols," unpublished, (2010).

[3] A.Raimond,"Appariement de données géographiques utilisant la théorie des croyances",(2007).

[4] A.Essaid and B.Yaghlane, "BeliefOWL:An Evidential Representation in OWL Ontology", (2009) .

[5] G.Shafer and J. Pearl, " Readings in Uncertain Reasoning",(1990).

[6] G.Shafer," Perspectives on the theory and practice of belief functions",International Journal of Approximate Reasoning,p1-40,(1990)..

[7] R. Laurini, « GeoWeb: Internet Géographique, Chapitre 5: Ontologies pour les applications géographique « ,(2009).

[8] J. Euzenat, P. Shvaiko: Ontology Matching, Springer edition (2007)

[9] N. Cullot, C. Parent, S. Spaccapietra, C. Vangenot: Des SIG aux ontologies géographiques, (2003)

[10] N. WeiBenberg, A. Voisard, R. Gartmann: Using Ontologies in Personalized Mobile Applications, (2004)

[11] F. Fonseca, M. Egenhofer, P. Agouris: Using Ontologies for Integrated Geographic Information Systems, (2002)

[12] J. Bertin : Sémiologie graphique: Les diagrammes -Les réseaux -Les cartes, (1967)

[13] M. Kavouras, M. Kokla: Ontology-Based Fusion of Geographic Databases, (2000)

[14] M. Horridge, H. Knublauch, A. Rector, R. Stevens, C. Wroe: A Practical Guide to Building OWL Ontologies Using the Protégé-OWL Plugin and CO-ODE Tools, (2004)

[15] E. Chesneau, A. Ruas, O. Bonin: Colour Contrasts Analysis For A Better Legibility Of Graphic Signs For Risk Maps, (2005)

[16] J. Itten : Art de la couleur, (1985). Original Edition (1961)

[17] C. Brewer, G. Hatchard, M. Harrower: ColorBrewer in Print: A Catalog of Color Schemes for Maps, (2003)

[18] J. Mersey: Colour and Thematic map Design: The Role of Colour Scheme and Map Complexity in Choropleth Map Communication,

[19] N. F. Noy, R. W. Fergerson, M. A. Musen: The knowledge model of Protege-2000: Combining interoperability and flexibility, (2000).

[20] A. Olteanu : fusion des connaissances imparfaites pour l'appariement de données géographiques, proposition d'une approche s'appuyant sur la théorie des croyances, (2008)

[21] T. Devogèle: Processus d'Intégration et d'Appariement de Bases de Données Géographiques - Application à une base de données routières multi-échelles, thèse de doctorat de l'Université Marne-la-Vallée, laboratoire COGIT, IGN-SR 97-025-S- THE-TD, (1997).

[22] M.Brown: Hacking Google Maps and Google Earth, (2006)

[23] E. De Vita, A. Piras, S. Sanna: Using compact GML to deploy interactive maps on mobile devices, (2003)

[24] J. Cardoso, A. Rocha, J. Lopes: M-GIS: Mobile and Interoperable Access to Geographic Information, (2004)

[25] N. Gesbert : Étude de la formalisation des spécifications de bases de données géographiques en vue de leur intégration, (2005)

[26] T. Badard : Propagation des mises à jour dans les bases de données géographiques multi-représentation par analyse des changements géographiques, thèse de doctorat de l'Université de Marne-la-Vallée, laboratoire COGIT, IGN-SR

[27] S. Mustière, N. Gesbert, D. Sheeren : Unification des bases de données géographiques, (2003)

[28] N.Grégoire, M.Bouillot: Hausdorff distance between convex polygons, (1998) [30] Wikipedia : Fréchet distance, http://en.wikipedia.org/wiki/Fr%C3%A9chet_distance

[29] .Davala : Distance de Levenshtein, http://pagesperso-orange.fr/jean-paul.davalan/lang/algo/lev/index.html

[30] Institut Géographique National, Laboratoire COGIT, www.ign.fr