GEO-MAPPING MASHUPS USING OPEN APIS: THE STATE-OF-THE-ART AND AN APPLICATION FOR GEO-SPATIAL WEB

Yong Jae Park and Kiwon Lee

Dept. of Information System Engineering, Hansung University
E-mail: kilee@hansung.ac.kr

ABSTRACT: Since the mid-2000, so-called Web 2.0 paradigm was emerged and has been widely extended to actual Web markets with supporting Web technologies. Among Web 2.0 services, mashups with Open APIs is regarded as one of core technologies. Web mapping is also the most dominant mashup application in Open API-based applications, so that various mashups with Google Map API and other Open APIs are developing. In this paper, current technological status is explained with web GIS file formats of KML. While, a prototype is web-based GIS layer authoring system for general users, without any kinds of GIS tools and external spatial databases. This mashup application will be used web-based feature extraction for geo-spatial web for public users.

KEY WORDS: GeoJSON, Google Maps API, Mashup, Web 2.0

1. INTRODUCTION

In most web service systems with query functionality in Web 1.0, contents providers deliver them from their database server or accessible account to its server, as long as client users request data or information what they need. Whereas Web 1.0 is on provider-centric and server-oriented service, Web 2.0 paradigm pursues users' participation, information sharing, and interface openness. In facts, Web 2.0 can be realized in the form in combination with several web computing technologies such as AJAX (Asynchronous Javascript and XML), Open API (Application Programming Interface), REST (Representational State Transfer), XHTML/CSS, RSS (Rich Simple Syndication) and so forth.

Mashup, what Wikipedia formerly defined as a web application that combines data from more than one source into a single integrated tool, is a kind of realization of Web 2.0, which is a key item in this study.

Now that mashup using open map APIs is one of the popular and affordable ways to web mapping or web map service based on Web 2.0, there are various types of mashup applications. As for open map APIs, enterprise portals provide their own APIs: Google maps API, Yahoo maps API, Microsoft Virtual Earth API and Naver maps API.

An implementation for mashup application is performed in this study. The implemented result using Google maps API is web application so that user on web browser can manually digitize or extract and store geo-spatial feature directly from geo-based resources within Google mapping server, without any other GIS tools or geo-spatial database.

2. GEO-MAPPING MASHUP: STATE-OF-THE-ART

2.1 Geo-spatial Web

Geo-spatial web, shown in Fig. 1, is somewhat a new paradigm, compared with conventional web-GIS. Geo-spatial web focuses on contexts situated in both real world and digital space, and can be realized with ubiquitous computing and GIS contents. Main components in geospatial web are Geobrowsers, social software, and web 2.0. In facts, these three components are not separated ones, and eventually closely-related ones.

Especially, web 2.0 is a kind of paradigm for web services, which needs some web technologies such as open API, light weighted applications, and mashups.

Concerning these mapping APIs, Programmable.com as a specialized portal for web computing and mashup has reported the status of mashup usability and the rank of used open APIs (Fig. 2).
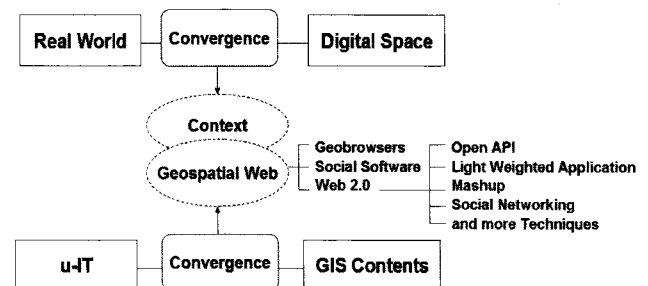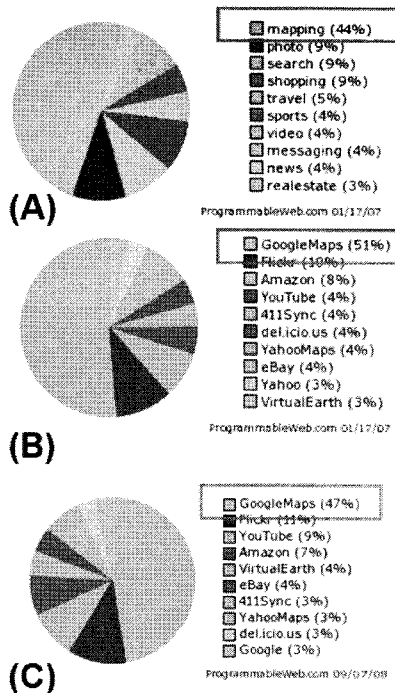

Fig. 1. General view of Geo-spatial web.

(A)

- mapping (44%)
- photo (9%)
- search (9%)
- shopping (9%)
- travel (5%)
- sports (4%)
- video (4%)
- messaging (4%)
- news (4%)
- realestate (3%)

ProgrammableWeb.com 01/17/07

(B)

- GoogleMaps (51%)
- Flickr (10%)
- Amazon (8%)
- YouTube (4%)
- 411Sync (4%)
- del.icio.us (4%)
- YahooMaps (4%)
- eBay (4%)
- Yahoo (3%)
- VirtualEarth (3%)

ProgrammableWeb.com 01/17/07

(C)

- GoogleMaps (47%)
- Flickr (11%)
- YouTube (9%)
- Amazon (7%)
- VirtualEarth (4%)
- eBay (4%)
- 411Sync (3%)
- YahooMaps (3%)
- del.icio.us (3%)
- Google (3%)

ProgrammableWeb.com 09/07/08

Fig. 2. Mashup types and the rank of open APIs used.

Fig. 2 shows a part of their results as of the year 2007 and 2008; as shown in Fig. 1(A), mapping is the most popular one among many applications. Moreover, it shows that Google maps API is the most-adopted open API for mashup during the past two years. Though Google or Yahoo is not the GIS software company or mapping company, just for portal services, their open APIs contribute to increase the worthy value within spatial data and information and to extend the scope of web mapping business.

A funny web site of gmapsmania reported a case examples named '100 Things to do with Google Maps Mashups' In those, most applications are for public users on world coverage with satellite images and map layers provided from Google mapping servers (Ref URL: http://gmapsmania.googlepages.com/100thingstodo withgooglemapsmashups).

Further, according to Turner (2008) and Wilson (2008), OGC (Open Geospatial Consortium, Inc.) officially accepted KML, XML-based file structure, of Google map and earth as the standard exchangeable geo-format on web environment, in the mid 2008.

While, Fig. 3 shows user interface of OpenStreetMap, one of the distinguished and wide-used mashup applications. This mashup is for editing of mapping data with users' GPS data and file exporting in the form of PNG, JPEG, and other image formats.

Fig. 3. Mashup case: OpenStreetMap 2008
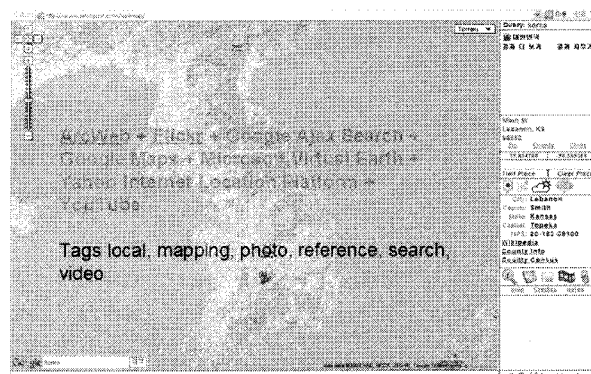(http://www.openstreetmap.org/)

Fig. 4. Mashup case: SAKMAP 2008.

Fig. 4 is another mashup cases named SAKMAP, and this provides the various web mapping linked with flicker-images and YouTuBe animation image sets with all kinds of mapping APIs such as Google, Virtual Earth, and Yahoo.

2.2 An Implementation in this study and Technological Consideration

The Google maps API (http://www.google.com/apis/maps/) provides several dozens of function sets for web mapping application using huge geo-spatial data resources in Google mapping server (Davis, 2006, 2007; Purvis et al., 2006). Functions can be classified into several parts: core object, map control, user data, AJAX, and Event. Classes with NAVER maps API is not enough to implement mashup for web mapping, compared to those of Google maps API.

Classes and objects in core objects are mainly for display of user-defined or searched zone. Those in map control are for manipulation of displayed scene. User data contains several classes and objects to mark data sets to user's purpose. AJAX

classes are to enable interactive web pages without requesting the browser to refresh. This plays a role in mashup application development for web mapping, and this make Google map be useful and accessible to web developers. No special development tools are required in order to take advantage of Google maps API; all that is necessary is a text editor, web browser, and a public web server from which the scripts can be served.

Web service with the basis of Web 2.0 needs a strategic design process for good performance of better data building or contents creation. AJAX dealing with XMLHTTPRequest Objects is used for a mashup which is a web application that combines data from more than one source into a single integrated tool. Normally, mashup site can be divided into four parts: content provider, API provider, Client web browser, and mashup site.

KML (Keyhole Markup Language) is an XML language for expressing geographical annotation and visualization which is used by Google maps,

Google earth and Google mobile, and it has become ubiquitous in the geospatial web, with support for import and export from commercial mapping APIs such as Google Maps, Microsoft's Virtual Earth, and other web mapping APIs. Nowadays, KML is more widely used in web mapping mashup.

Fig. 5 represents schematic diagram with respect to an implementation in this study.

As user-side view, a user is on IE (Internet Explorer), and freely accesses this mashup. Then user inquires ones interested area, and that area is shown at the center location in the main frame. In this frame, user digitizes vector-typed data such as point, line, and polygon on screen, and can check their geographic coordinates on other frame (Fig. 6). After on-screen vectorizing and editing, these data can be stored in the form of KML or GeoJSON (Fig. 7) into user's local disk. This newly generated and stored file is also overlay on Google Earth for the further geo-processing linked with web mapping (Fig. 8).
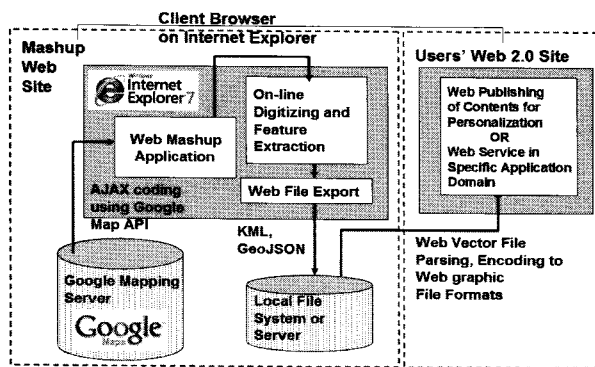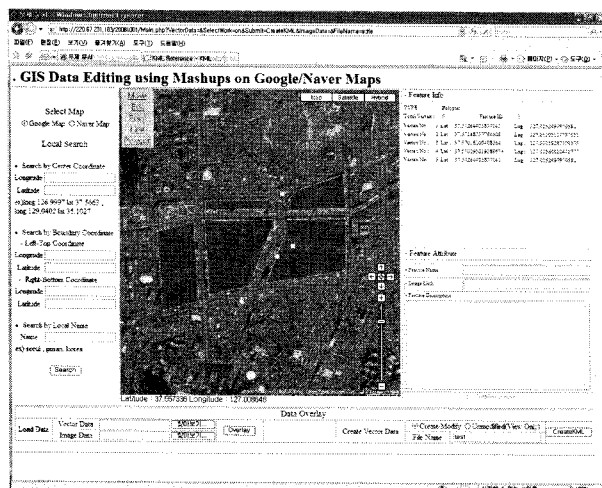

Fig. 5. Implementation scheme and workflow.

Fig. 6. User interface of mashup application implemented in this study.

Fig. 7. Exported KML from this mashup application, the actual data set shown in Fig. 6.
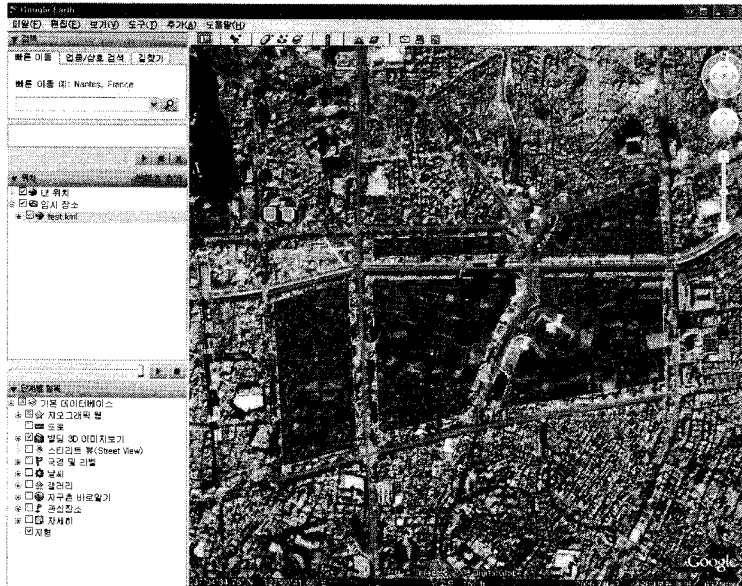
Fig. 8. KML, newly generated in Figs. 6 and 7, overlaid on the same area of Google Earth.

3. CONCLUDING REMARKS

In this study, a prototype for mashup application using AJAX computing of open API is presented as web mapping service based on web 2.0 paradigms and computing schemes. Google map composed of open API for mashup and huge map resources covering all over the world is an important milestone for web mapping. For public uses, the number of web sites for web mapping or mapping mashup is gradually increased. But most mashup applications are for geo-spatial data browsing or viewing. Current, there are no mashup applications for geo-spatial feature generation, being implemented using Google maps API. So, this is the first mashup application for the purpose of feature generation directly from Google mapping server. By using this mashup application, public users and GIS experts on web client of MS internet explorer can create their own geo-spatial data sets in the form of vector typed feature, without help of any GIS tools or accessible account to proprietary databases.

References

Davis, S., 2007. GIS for Web Developers: Adding Where to Your Web Applications, The Pragmatic Bookshelf, 254p.

Davis, S., 2006. Google Maps API v2: Adding Where To Your Applications, The Pragmatic Bookshelf, 75p.

Purvis, M., J. Sambells, and C. Turner, 2006. Beginning Google Maps Applications with PHP and Ajax, APRESS, 358p.

Turner, A., 2008. Emerging Mass Market Geo Standards, OGC Technical Committee Meeting: 1-88.

Wilson, T. (Ed.), 2008. OGC-KML, OGC 07-147r2.

Acknowledgements