

LOWERING THE BARRIER TO OSM CONTRIBUTION AND CITIZEN MAPPING SERVICES DEVELOPMENT FOR CITY GOVERNMENTS

Chao-Kuei Hung¹

¹Chaoyang University of Technology, ckhung@cyut.edu.tw

Abstract

This article uses a mapping program potluckmap to propose two ways for city governments to improve tourist experiences. By focusing on nodes only, OpenStreetMap editing can be reduced to collaborative spreadsheet editing. Hopefully lowering the barrier to contribution will bring more local business owners to advertise themselves on OSM. By separating the real-time bus position information from the map drawing, more bus companies can easily make their information available to tourists, and tourists can use a single map application to navigate in many cities across the globe.

Keywords: OpenStreetMap, tourism, local businesses, public transportation

1. BACKGROUND AND THE PROBLEMS

More and more city governments learn to open up their data to the public and make use of the results from public participation projects such as OpenStreetMap and software projects with Free/Libre and Open Source Software (FLOSS) licenses. The released data feed FLOSS projects such as mobile apps and/or websites for transportation and tourism information, thereby benefiting residents and local businesses without taxing city governments on their software development resources.

In OSM, area- and street-information is relatively complete in many cities, but Point-of-Interest (POI) information such as restaurants, shops, and entertainment spots is typically quite incomplete or even sparse even in big cities. Even with the advancement of OSM editors such as JOSM and iD, the entrance barrier to OSM editing contribution is still considerably higher than that to Wikipedia contribution for example.

At the application end, we see a proliferation of public transportation mobile apps for tourists. However, having to install and learn a different app for each different city consumes phone storage as well as human learning efforts.

In this article we address the above two challenges with a github project potluckmap. This javascript program is quite simple and does not yet provide many fancy functions. The emphasis is more on its potential social impact than on the technological aspects.

As a long time FLOSS advocate, the author sees the importance of lowering the barrier to contribution. Ubuntu launchpad and the .po files infrastructure makes it for translators to participate. Github projects employ wikis to enlist more document writers. In both cases, the development hassles (compilers, build managers, version control, ...) are separated from the simple text contributions. In the same spirit, potluckmap makes it possible to separate the mapping display and interaction from the preparation of textual geographic data, much like the idea of "separation of content from presentation" in the word processing or web publishing world.

2. THE TOOL

The idea behind potluckmap is to have a map page or application that allows the user to specify multiple data sources (possibly in various different formats) for showing markers and/or additional map features on her own custom map. (Hung, 2016) There are existing projects such as umap which include a similar but far more limited feature. (Boniface, 2012)

The data sources can be either local files or public URL's. Each of the data sources can be in the leaflet default geojson format or in one of the formats supported by the leaflet-omnivore project, including csv, gpx, kml, wkt, etc. (Boniface, 2012) Potluckmap was originally called chorusmap and later changed to potluckmap to

honour this underlying powerful geographic-file-reading engine. Additionally, potluckmap also accepts the output of OverpassTurbo requests. (Raifer, 2012)

Data sources can be instructed to automatically refresh themselves at fixed time intervals. Moreover, potluckmap rotates each marker according to its "Azimuth" field. Together, these two features make it suitable for displaying moving vehicles such as buses, cars, and bikes.

3. FOCUSING ON AND DIVIDING THE OSM POI-EDITING LABOR

In the few mapping parties the author has participated in, he has observed some participants daunted simply by seeing the rich set of tool buttons of JOSM or iD editor. To address the barrier problem for collaborative POI-editing, a logical solution is to abandon all full-fledged OSM editors when we are mapping areas where the ways information (streets, rivers, etc.) are already relatively complete. Of the the three types of data in OSM (nodes, ways, and relations), we are thus only interested in nodes, which can be easily tabulated in a spreadsheet.

Conceivably far more people have used google sheets to collaborate than those who have ever used JOSM or iD. In stark contrast to the required efforts for learning/teaching JOSM and iD editors, collaborative spreadsheet editing requires practically no training. We choose google sheets' FLOSS alternative ethercalc for our workflow. (Tang, 2011) More similar to google docs and different from wikis, it also allows real-time collaborative editing without having to lock down the document. Moreover, simply appending ".csv" at the end of the URL of a ethercalc page gives us a csv version of the same page, which can be fed to potluckmap. Hence we propose a new form of mapping party following this workflow:

1. A small group of relatively newbie OSM mappers walk through an area and add nodes into OSM with only a primary tag ("shop", "amenity", etc.) and its name.
2. A data expert extracts nodes data from that area using for instance the OverpassTurbo service, imports the data into an ethercalc page, and keep/add only fields of interest to this particular mapping party.
3. Untrained first time OSM contributors fill in these fields on the ethercalc page by googling, making phone calls, etc.

4. Up-to-date collaborated editing status can be previewed in potluckmap.
5. At the end the data expert merges the ethercalc page back to OSM.

It is hard to imagine inviting local business owners into a traditional OSM mapping party, where they have to be trained first about how to use JOSM or the iD editor. Yet it could be a good opportunity for them to advertise their businesses. In this new form of "spreadsheet mapping party", however, non-technical people may very well participate just once and forget about it after making considerable amount of editing contributions with much less training.

4. ONE TRANSPORTATION MOBILE APP FOR MANY CITIES ACROSS CONTINENTS

As stated in their website, CitySDK is a "service development kit" for cities and developers that aims at harmonizing application programming interfaces (APIs) across cities. (Varteva, 2014) The homepage seems to be quiet for the past one year or two but at least one github project is still active. (Boozallen, 2015) Unfortunately the author of this article has been unable to locate a CitySDK API for public transportation.

Potluckmap can be a simple starting point for public transportation in the spirit of CitySDK. Bus companies can simply provide a URL that returns, in response to a query containing a line number, a geojson or gpx file detailing the real-time positions and orientations of that particular line. A user can then put a few such query URL's into potluckmap and set refresh intervals to see a dynamic map of running buses of those bus lines, as demonstrated at the project homepage. Of course it's even easier for the bus companies to publish and for the user to include static information such as bus stops and bus routes into the map.

Admittedly potluckmap does not yet have many functionalities. The more important point, however, is that participating city governments and bus companies can leave the work of producing a mapping application to the developers, and a small group of developers can enhance the tourist experiences in all of the participating cities all at once.

More likely, other competitive application programs will be created once the infrastructure is in place. Eventually there will again be many city transportation programs or mobile apps across globe, except that this time each user can choose

the one to her or his liking and use just that single app as s/he travels among the participating cities. One particularly interesting possibility is to have such a plugin for the offline map app osmand. (vshcherb, 2010)

5. CONCLUSIONS

Even in the time at its inception, a wiki program was not an advanced technology. The new way of collaboration it brought about, however, resulted in noticeable efficiency improvements in many areas of our daily lives and created one of the most important encyclopaedia of our time.

Using potluckmap the simple mapping application as a demonstration, this article advocates a new form of OSM mapping party in relatively well-mapped areas, plus a new way of providing public transportation information. With a significant reduction in OSM editing training, hopefully we can help the OSM project grow stronger by "thickening" the long tail of the editing contribution. By removing the need for bus companies and city governments to write their own real-time bus mapping programs, hopefully more of them would be willing to devote a smaller amount of energy and resources to publishing such information in textual form for the world's programmers to play with.

As a traveller, the author looks forward to the day when he can use a single mobile app in most foreign cities to browse interesting restaurants, shops, etc., and to look up real-time bus information about how to get there.

6. REFERENCES

- Boniface, Y. et al. (2012), umap, at <https://github.com/umap-project/umap>, [accessed 1 July 2016]
- Boozallen, J. T. et al. (2015). CitySDK. at <https://github.com/uscensusbureau/citysdk>, [accessed 1 July 2016]
- Hung, C. K. (2016). Potluckmap, at <https://github.com/g0v/potluckmap>, [accessed 1 July 2016]
- MacWright, T. et al. (2014). leaflet-omnivore, at <https://github.com/mapbox/leaflet-omnivore>, [accessed 1 July 2016]
- Raifer, M. et al. (2012). Overpass-Turbo, at <https://github.com/tyrasd/overpass-turbo>, [accessed 1 July 2016]

Tang, A. et al. (2011). EtherCalc, at <https://github.com/audreyt/ethercalc>,
[accessed 1 July 2016]

Varteva, A. et al. (2014). CitySDK, at <http://www.citysdk.eu/> , [accessed 1 July 2016]

vshcherb et al. (2010). Osmand, at <https://github.com/osmandapp/Osmand>,
[accessed 1 July 2016]