MapXplore:

Linked Data in the App Store

Csaba Veres

University in Bergen, Norway
csaba@csabaveres.net

Abstract. MapXplore is an attempt to build a mainstream, useful, and easy to use application that uses linked data at its core. As such, it will be one of very few such apps on the Apple, or for that matter any of the mobile app stores. The purpose of the application is to allow users to browse any part of the globe, and identify points of interest from DBPedia articles. They can then drill down into traditional as well as linked data sources to get a comprehensive view of the points of interest. We note some difficulties in working with current linked data resources, and suggest some methods to help pave a future rich in popular, easy to use mobile semantic applications. By using these guidelines we aim to keep refining MapXplore to make it a showcase application for the power of linked data.

Keywords. Mobile, linked data, iPad, iOS, dbpedia, wikipedia, tourism

1 Introduction

Mobile computing is perhaps one of the most ubiquitous and fastest growing technological innovations in recent years. Only sixteen years ago, [1] predicted that the then rapidly expanding technology of cellular communication would eventually give rise to a future in which millions of people would carry portable computing devices which would not need a physical network connection. Their vision soon became a reality as wireless technologies continued to improve, which prompted the release of innovative hardware platforms like the Apple inc.® iPhone® in 2007, which was perhaps the first device to bring a general, user friendly computing platform into widespread use. Apple inc.® subsequently opened the "App store" where native applications for their iOS platform could be downloaded for free or for a relatively low price (compared to PC based applications). The App store opened on July 10, 2008, and as of June 2012, reported over 30 billion downloads from the 365 million iOS devices sold through March 30, 2012.[1] Following Apples's phenomenal success many other manufacturers opened their own application stores, and Google inc.® released their Android® platform which copied many of Apples's usability features. Together, the two platforms dominate the mobile platform landscape [2].

It is possible that the runaway success of smartphone devices is due in no small part to the native apps. The story goes that Steve Jobs was initially opposed to third party applications on the iPhone, preferring instead Web2.0 and Ajax apps that run on the Safari engine built into the iPhone. However, web apps never became popular, so in 2008 developers were given access to the native SDK, and the wildly successful app store opened in conjunction with the release of the new iPhone 3G.[2] The usability guru Jakob Nielsen has shown that native apps on mobile devices are far superior to web apps, and the main motivation for preferring the development of web apps is more based on strategy and finance than usability.[3]

The rise in popularity of mobile "smart phones" has become most staggering in just the last couple of years. A Nielsen report from Q3 2011 shows that 44 percent of U.S. mobile subscribers own a smartphone device as of Q3 2011, compared to 18 percent just two years ago, and in the 25-34 age group the figure is 64 percent. Moreover, the number of smartphone subscribers using the mobile Internet has grown 45 percent since 2010 [2].

The ubiquity of smartphone devices has been used to advantage by app developers who think of new ways to exploit location based mobile use. A prominent example is the so called SOCIAL + LOCAL + MOBILE or "So-Lo-Mo" trend, which

---

[1] http://events.apple.com.edgesuite.net/126pihbedvcoihbefvbhjkbvsefbg/event/index.html

[2] http://www.cultofmac.com/125180/steve-jobs-was-originally-dead-set-against-third-party-apps-for-the-iphone/

[3] http://www.useit.com/alertbox/mobile-sites-apps.html

is a term for marketing efforts to target their information through a localized, social platform which is sensitive to users' environmental context. [2] reports that in the United States, 87 percent of people use deal-of-the-day sites like Groupon[4] which offers current deals in the user's immediate vicinity.

Apart from these lucrative marketing efforts, feature rich mobile devices have also given rise to a large number of applications that help users explore their immediate environment for information, entertainment, food, companionship and many other opportunities. The major search engines have responded to the need for people to know "where am I and what is around me?" with services like Google Places.[5]

In addition to the real world applications, there has also been a huge amount of academic work on the topic of Mobile Computing and Context-aware systems e.g. [3]. We will not review these in any depth, because the focus of this paper is the experience of developing a usable, mobile semantic application.

The application presented in this paper is predicated on the feeling that, since the integration of information is a defining characteristic of linked data, there should be a large number of linked data applications for contextually rich information integration in the various application stores. Sadly, this is not the case. The "mobile semantics" blog[6] is collating a list of iOS and Android apps which claim to make use of semantic technologies in some form. The number of semantic apps at this moment is vanishingly small, and most of them provide functionality that is esoteric, or does not provide clear advantages over competing applications.

In spite of the somewhat disheartening status quo on the mobile app stores, we believe that mobile semantic apps have a bright future. In order to achieve success, however, future apps need to fulfill a number of requirements. The author of this paper is working on defining a set of desirable characteristics that a successful mobile semantic application must possess.[7] Currently there are four conditions:

- The semantics should help rather than hinder the user. The app should not simply present 100 possible links for the user to follow.
- The application should present some clear advantages over non semantic versions. It should be able to do some clever and useful things that simply cannot be done by competing apps of similar functionality, without semantics. The semantics should make it the go to app in its domain.

---

[4] http://www.groupon.com/

[5] http://www.google.com/places/

[6] http://mobilesemantics.blogspot.no/2012/05/mobile-semantic-apps-where-are-they.html

[7] http://mobilesemantics.blogspot.no/2012/05/mobile-semantic-apps-where-are-they.html

- It should be usable. Nobody (almost nobody) wants to type SPARQL queries into a 4 inch touchscreen!
- The semantics should be non trivial.

There are currently two prominent applications which fulfill these requirements. The first is Siri, Apple's personal assistant for iOS.[8] Siri is able to carry out simple inferences on users' questions, in some domains. This is accomplished by, what Tom Gruber calls it, "semantic snap to grid".[9] By modelling the user's behavior in a task oriented, contextually constrained information space, Siri is able to make a guess about intended meanings of questions.

A second application is news360° which is a news aggregator that uses semantic technologies for selecting and ranking related news stories in a personalized manner.[10] Its use of semantics is non trivial and completely transparent, and its selection of related stories is impressive.

MapXplore aims to be a user friendly app which introduces users to the world of linked data, and is a first step in the attempt to bring semantics into popular use. In this first release, the focus is on fulfilling requirements two and three, to create a user friendly application which showcases the usefulness of linked data. The semantics is currently limited to simple semantic categorization. The application also contains an implementation of Google analytics allowing us to track its use and ascertain which features are used, so we can see whether or not the features are found useful.

In the remainder of the paper we describe a use case which prompted the development of MapXplore. We look at some existing apps with related functionality, and discuss why a linked data application might be needed to better fit the requirements. Then we report on challenges as well as successes of developing the application with linked data.

2 Use case and Existing Applications

The use case centers around the idea of exploring points of interest (POIs) in a particular geographical area. This could be in situ, as in the case of a tourist who is in a foreign location, or in exploration mode where a person can navigate to any part of the globe on a map and explore points of interest. The application should highlight these points, and provide some means for selecting potential points for further investigation. That is, there should be some system by which the POIs are grouped

---

[8] http://www.apple.com/iphone/features/siri-faq.html

[9] http://vimeo.com/5424527

[10] http://news360app.com/

into coherent categories. Once a POI is selected, there should be ways to find additional information about it. The information should be comprehensive, and include visual and factual information. The POIs should be of broad interest, not focusing on just a few targeted categories like “restaurants” and “hotels”. Above all the application should be easy to use, performing quickly and providing information in an intuitive fashion.

There are existing applications which provide broad, location based information content rather than specialized services such as hotel bookings, for example.

Localscope[11] is a “Location Browser for your iPhone”, which uses the current GPS location to retrieve information from a large number of services. However, even though the breadth of data is large, the information is not integrated in any sense beyond the fact that they share a common geospatial bounding box. For example, Wikipedia and Google Places might have information about the same resource in a given area, but there is no connection between the two information sources. Altogether the experience feels somewhat disconnected because each service returns an overlapping set of independent results. However, it does provide a very comprehensive search interface, and users can search by address, facility (gas station, wifi, cafe), and so on, and returns informative results.

Roamz[12] is an application which focuses more on the socially oriented services. It provides filtered content from Facebook, Foursquare, Instagram and Twitter, based on a personal profile. While the breadth of information on Roamz is smaller than on Localscope, it is more connected because clicking on a result will retrieve more results about the same entity. However, none of the results are designed to give factual information as they are all from the social media. The intent of the app is to facilitate spontaneous discovery based on catchy photographs and comments rather than informative content.

DBPedia mobile [4], [5] is an application built to explore the “geospatial semantic web”. It is based on DBPedia entries that have a geospatial property, so the articles can be retrieved based on a location of interest. The POIs are retrieved from a SPARQL endpoint and placed on the map with an icon corresponding to their YAGO category[6]. Selecting any icon will pop up a Fresnel based Linked Data browser[7], which integrates information from various linked data sources and gives access to photographs from ‘flickr(™) wrappr’.[13] The application is a brilliant demonstrator which includes the ability to filter results by their properties, as well as execute full SPARQL queries. Unfortunately the author has had very little success in interacting with the service. Its interface is quite rough and can be difficult to navigate, partly due to the fact that it is not a native app. The application fails to retrieve any POIs around

---

[11] http://itunes.apple.com/us/app/localscope/id409869453?mt=8

[12] http://itunes.apple.com/app/roamz/id459343660

[13] http://www4.wiwiss.fu-berlin.de/flickrwrappr/

the author's location in Bergen, Norway, forcing him to chose one of the few predefined locations. Browsing POIs with the Fresnel browser is patchy and unreliable at best, since results are often not retrieved. In fact, this complete reliance on an external service for integrating information is the single biggest usability issue, since a failure in that service renders the entire application useless.

WorldExplorer[14] is an application for iOS as well as a version for OSX, which also shows POIs from WikiPedia, but in a more user friendly manner. For a given location it shows Wikipedia POIs with a thumbnail, which can be ranked by distance from current location or by rating. The bottom left panel shows Google maps, which can optionally be viewed in streetview. On the right one can read the Wikipedia article about the selected POI. There is also current weather information. The application is very highly rated in both the iOS and OSX app stores.

MapXplore can be regarded as a combination of DBPedia mobile and WorldExplorer. It is an easy to use and attractive application which includes most of the existing functionality of DBPedia mobile, with an immersive browsing interface. It includes many more information sources about each resource than WorldExplorer by virtue of linked data sources, and includes a much richer interface for selection of locations of interest, by virtue of semantic classification.

3 MapXplore Overview

MapXplore is a native iPad application. Its main window is a map view which zooms to the user's current location when the app launches (fig. 1). It retrieves surrounding POIs from DBPedia, within a certain distance of the current location. Currently this is set at 2KM., but the distance is user configurable. The reason it is set at such a close distance is that some places, for example the city of London contain very many POIs per square kilometer, and the application makes no attempt to filter the results before presentation. The defined distance returns a reasonable number of results for most locations, and the user has the option to adjust the range after they see the returned results. The location can also be selected in the search bar, or by moving the map to any location and "dropping the pin" with a long press on the iPad screen.

[14] http://itunes.apple.com/us/app/world-explorer/id381581095?mt=8

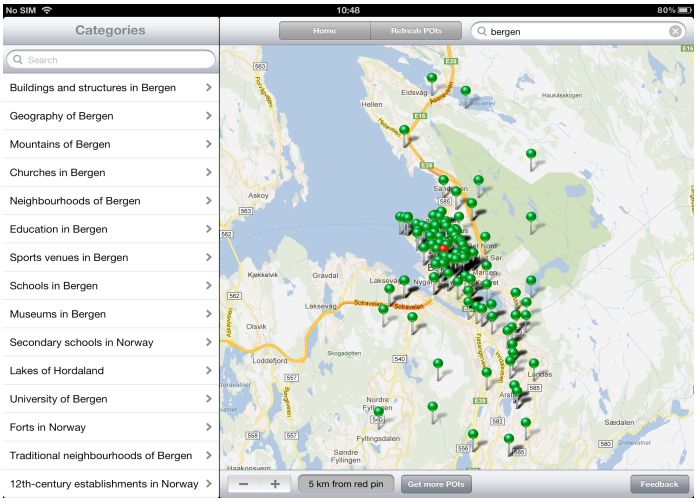

Fig. 1. MapXplore main window

The left table view shows the WikiPedia categories that describe the POIs, with the labels ordered by the number of POIs in each category. The interesting side effect of this is that the "character" of a city is quickly apparent just by looking at the categories. For example Bergen is clearly famous for its geography and in particular its mountains.

POIs can be selected in various ways, including by choosing a category which reveals the contained POIs, which can then be selected. In fig. 2 the user has selected Bergen Museum, which triggers a popup with the available data sources. The sources are broadly listed under "Pictures" or "Information". The information is retrieved individually from the services, so there is no single point of failure, even though individual services might intermittently fail to return a result.

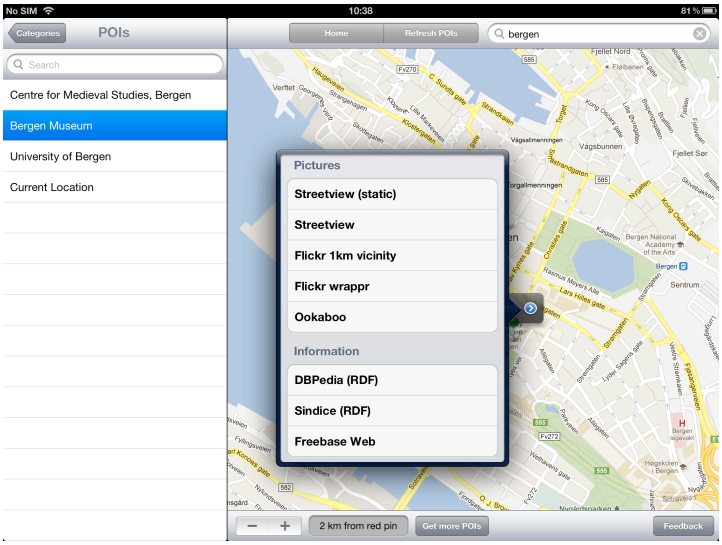

Fig. 2. Data sources for the selected resource

3.1 Pictures

There are currently five image sources. The first three are purely location based, and the remaining two are semantic services in which the picture is linked to the resource itself. This mixture serves two purposes. First it provides a richer experience to the user who can investigate the resource from more perspectives. Second, it enables a quantitative evaluation of the coverage and quality of the semantic services.

- *Google streetview image API*.[15] provides static images taken outward in a radial direction from the specified location. MapXplore retrieves four images at 0, 90, 180 and 270 degrees from the central location and displays them in a popup image viewer.
- *Google Maps JavaScript API v3*[16] embeds Google maps with full streetview functionality in an embedded popup browser window.
- *Flickr*[17] the original photo sharing site, which was partly responsible for the inception of Web2.0. Users upload photographs and add freeform tags. Many of the photos are taken with mobile phones and contain GPS location data. The Flickr API has several options for returning photographs, and in

[15] https://developers.google.com/maps/documentation/streetview/

[16] https://developers.google.com/maps/documentation/javascript/streetview

[17] http://www.flickr.com/

MapXplore we retrieve photos ranked by "interestingeness" in a 1Km. radius around the location of the resource itself. No effort is made to ensure that the photograph is actually a depiction of the resource itself.

- *flickr TM wrappr*[18] provides images from Flickr, but attempts to return only highly relevant images by matching DBPedia metadata against flickr tags and titles. The pictures will therefore most often be of the resource itself rather than something else in the vicinity. The images are interlinked with DBpedia using the foaf:depiction relationship type.
- *Ookaboo*[19] "is a collection of free pictures, indexed by precise terms from the semantic web." The semantic API allows retrieval by RDF Freebase identifiers, DBpedia Resource identifiers, or Wikipedia pages. MapXplore retrieves photos with the DBPedia identifier.

3.2 Information

The three information sources contain linked data which the user can explore to enrich their understanding of the domain.

- *DBPedia* links provide is a web view of the set of RDF triples about a given resource, for example http://dbpedia.org/page/Bergen. The web view contains clickable links), so users can follow the information graph in any way they choose. While this may not be the most user friendly way to browse the data, we feel that it is not a bad first experiment. Many of the links have clear and descriptive names, so users should have no trouble finding interesting facts like populationMetro, postalCode, octHighC, octLowC, and so on.
- *Sindice*[20] is a service which extracts and indexes semantic markup in the form of RDF, RDFa, Microformats or Microdata from web sites, in the form of RDF triples. It provides several ways to search and query, including free text searches, exact matches, predicates, triples, and even has a SPARQL endpoint. It returns web sites matching the query, as well as the RDF triples which were scraped from the web sites. MapXplore performs a simple query with the name of the resource, and presents the Sindice search results in a popup browser window.
- *Freebase*[21] is "an open, Creative Commons licensed repository of structured data of almost 23 million entities".

---

[18] http://www4.wiwiss.fu-berlin.de/flickrwrappr/

[19] http://about.ookaboo.com/a/about/what_is_ookaboo

[20] http://sindice.com/

[21] http://www.freebase.com/

Altogether, the information sources provide a rich mashup of information about a particular resource. But the mashup is a collection of semantic and non semantic data. We think this is important because it lowers the barrier for entry by providing familiar components alongside a new set of novel features. Our expectation is that the familiar components will encourage users to explore the unfamiliar territory, and this clam will be tested once usage data becomes available.

The data sources are a mixture of transparent and not so transparent semantics. In particular, DBpedia and Sindice present an interface that is less than ideal, since it requires some investment of effort to navigate. One of the major design plans for future releases is to streamline the presentation and exploration of linked data from these sources, in particular by making them contextually sensitive to respond to individual user needs and profiles.

4 Technical issues and challenges

One of the major advantages of building the application with semantic technologies is that the retrieved POIs can be modified without any changes to the application itself, since queries are made via SPARQL to a selected triplestore. The query simply retrieves the labels and category names for all resources within the specified geolocation. If the content of the triplestore is modified by the inclusion of any resource whatsoever, it will show up on MapXplore.

An unfortunate problem, however, is in the reliability of the data sources. During the development process it became painfully obvious that the server at http://dbpedia.org/sparql would not reliably return results. Many times it would simply time out without a response. Our solution was to set up a second server with replicated data, and switch over if dbpedia.org failed to respond within five seconds.

A second problem was the response speed, which could be slow because the SPARQL queries used FILTER clauses to retrieve just the resources in the specified range. Some triplestores provide a special mechanisms for geo data. For example Virtuoso has some built in functions (e.g. bif:st_intersects) that can be used to filter by location.[22] Unfortunately one can't rely on using these functions in general because they have to be specially set up by the administrator. For example, a second Virtuoso instance at http://live.dbpedia.org/sparql does NOT support the functions. The impact of this issue is exacerbated by the previous problem, because the queries have to be written with sufficient generality to work with potentially any triplestore.

Another solution might be to consider an alternative query method for retrieving the Wikipedia articles, and generating a DBPedia identifier from that. While the Wikipedia API does not support query by location, there are several third party solutions. The most straightforward is WikiLocation[23]. However this only

[22] http://docs.openlinksw.com/virtuoso/rdfsparqlgeospat.html

[23] http://wikilocation.org/

returns the Wikipedia article identifier, so an extra lookup would be required to gather information about semantic category, etc.

A more flexible option is a new "big data" service at Infochimps, who have an API for Wikipedia data.[24] This API is far richer than WikiLocation. The problem with the API, however, is that it is not possible to retrieve selected category information. Infochimps appear to have inserted their own categorization, but this does not seem as useful or flexible as the categories retrieved through SPARQL from DBPedia and its various mappings.

5 Preliminary evaluation

As previously mentioned, MapXplore contains a Google analytics implementation, and we have some preliminary results based on one month's availability. The application has been downloaded in 37 countries, and received a total of more than 1000 page views.

The application is an excellent platform for comparing information sources linked from a unique identifier. It is easy to gather data about the coverage and content of the different sources. Freebase, for example, appears to have fairly good coverage of DBPedia resources, but the information does not seem to contribute significantly: much of the text is simply a copy of the Wikipedia article, but some of the structured information is useful. Flickr(™) wrappr returns photographs of the resource itself more reliably than Flickr itself, so the semantic algorithm is good. However, its coverage can be patchy. Similarly for Oookaboo. The overall subjective impression is that the linked data provide accurate additional information, but their coverage is patchy.

The preliminary results seem to support these observations. The most frequently used features of the app are listed in descending order: retrieving and searching for more POIs, static street view, Wikipedia, Oookaboo, flickr wrappr, street view, DBPedia, Freebase, Flickr, Sindice. Users seem to favour the traditional information sources, but are very keen to try the semantically enhanced versions. Oookaboo and flickr wrappr are more than twice as often consulted than Flickr itself. The more complex DBPedia browser is more popular than Freebase. On the other hand the Sindice interface appears too confusing at the moment. Clearly future development will involve streamlining the DBPedia and Sindice user experience.

6 Conclusion

The combination of mobile and semantic technologies is surely going to break into the popular market in the coming years, especially as the availability and reliability of open data expands. We proposed a set of development criteria to help

---

[24] http://www.infochimps.com/datasets/wikipedia-articles

focus designers' attention in making successful applications. We don't want to deceive anyone that it is going to be easy. In spite of the massive growth of open linked data and consequent exposure in mainstream technology circles, it can still be difficult to find reliable, extensive data for interesting applications that can capture the public imagination.

MapXplore is a first attempt at making a linked data explorer tool that is both feature rich as well as immersive and easy to use on a mobile platform. It uses a combination of semantic and traditional technologies to gather information about geo located points of interest. We hope that this combination will make a well received and useful addition to many people's app libraries. Subsequent releases will look to additional data sources, enhanced usability of the information contained in the sources, and a bigger role for semantic integration.

7 References

[1] T. Imielinski and H. F. Korth, "Introduction to Mobile Computing," in *The Kluwer International Series in Engineering and Computer Science*, vol. 353, no. 1, T. Imielinski and H. F. Korth, Eds. Boston, MA: Springer US, 1996, pp. 1–43.

[2] Nielsen, "State of the Media: The Mobile Media Report Q3 2011," nielsen, Dec. 2011.

[3] J. Hong, E. Suh, and S. J. Kim, "Context-aware systems: A literature review and classification," *Expert Systems with Applications*, vol. 36, no. 4, pp. 8509–8522, 2009.

[4] C. Becker and C. Bizer, "DBpedia Mobile: A Location-Enabled Linked Data Browser," presented at the 1st Workshop about Linked Data on the Web (LDOW2008), Beijing, China, April 2008., 2008.

[5] C. Becker and C. Bizer, "Exploring the geospatial semantic web with dbpedia mobile," *Web Semantics: Science*, 2009.

[6] F. Suchanek, G. Kasneci, and G. Weikum, "Yago: A Large Ontology from Wikipedia and Wordnet," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 6, pp. 203–217, 2008.

[7] E. Pietriga, C. Bizer, D. Karger, and R. Lee, "Fresnel: a browser-independent presentation vocabulary for RDF," presented at the ISWC'06: Proceedings of the 5th international conference on The Semantic Web, 2006.