HDTourist: Exploring Urban Data on Android

Elena Hervalejo[1], Miguel A. Martínez-Prieto[1], Javier D. Fernández[1,2], Oscar Corcho[2]

[1] DataWeb Research, Department of Computer Science, Univ. de Valladolid (Spain)
elena.hervalejo@gmail.com, {migumar2,jfergar}@infor.uva.es
[2] Ontology Engineering Group (OEG), Univ. Politécnica de Madrid (Spain)
{jdfernandez,ocorcho}@fi.upm.es

1 Introduction

The Web of Data currently comprises ≈ 62 billion triples from more than 2,000 different datasets covering many fields of knowledge[3]. This volume of structured Linked Data can be seen as a particular case of Big Data, referred to as *Big Semantic Data* [4]. Obviously, powerful computational configurations are traditionally required to deal with the scalability problems arising to Big Semantic Data. It is not surprising that this "data revolution" has competed in parallel with the growth of *mobile computing*. Smartphones and tablets are massively used at the expense of traditional computers but, to date, mobile devices have more limited computation resources.

Therefore, one question that we may ask ourselves would be: *can (potentially large) semantic datasets be consumed natively on mobile devices?* Currently, only a few mobile apps (e.g., [1, 9, 2, 8]) make use of semantic data that they store in the mobile devices, while many others access existing SPARQL endpoints or Linked Data directly. Two main reasons can be considered for this fact. On the one hand, in spite of some initial approaches [6, 3], there are no well-established triplestores for mobile devices. This is an important limitation because any potential app must assume both RDF storage and SPARQL resolution. On the other hand, the particular features of these devices (little storage space, less computational power or more limited bandwidths) limit the adoption of semantic data for different uses and purposes.

This paper introduces our *HDTourist* mobile application prototype. It consumes urban data from DBpedia[4] to help tourists visiting a foreign city. Although it is a simple app, its functionality allows illustrating how semantic data can be stored and queried with limited resources. Our prototype is implemented for Android, but its foundations, explained in Section 2, can be deployed in any other platform. The app is described in Section 3, and Section 4 concludes about our current achievements and devises the future work.

2 Managing RDF in Mobile Devices

Our approach for managing RDF is inspired by the role played by SQLite[5] in Android devices. SQLite is a self-contained SQL engine which is deployed as

[3] Stats reported by LODStats: http://stats.lod2.eu/.
[4] http://dbpedia.org/.
[5] http://www.sqlite.org/.

an internal component of the application program. This way, the app itself can read and write data directly from the database files without requiring a separate process running as a DBMS (Database Management System).

Similarly, our only requirement is to hold properly serialized RDF files and a standardized interface to operate on them. Both responsibilities are provided by the RDF/HDT [5] format, which serializes RDF using up to 15 times less space than other syntaxes [4], while allowing basic SPARQL queries to be efficiently resolved on the serialized file [7].Thus, including RDF/HDT as a library[6] of the app, allows it to manage and query semantic data in compressed space.

3 HDTourist

HDTourist is a proof-of-concept app[7] built on top of RDF/HDT. It is designed as a lightweight app to provide tourists with information when they are in a foreign place. In these cases, people are more reluctant to connect to Internet because of the potentially expensive costs of roaming. Thus, our mobile device will be useful to keep compressed semantic information and query it offline.

Use case. Let us suppose that we plan our trip to Riva del Garda to attend ISWC'2014, and our flight arrives to Verona. Fortunately, we have a day to visit the city and decide to use HDTourist. Before leaving home, or in a Wi-Fi hotspot (*i.e.* in the hotel), we use our Internet connection to download the RDF/HDT file with relevant information about Verona. Currently, these data are obtained by exploring different categories related to the DBpedia entity modeling the city: `http://dbpedia.org/page/Verona`. In addition to semantic data, we can download multimedia: images, maps of the region, etc. to improve the user experience. We download them and HDTourist is ready to be used in our visit.

Verona's HDT file has $18,208$ triples, with a size of ≈850 KB, more than 4 times smaller than the original NTriples file (≈3.6 MB). Beyond the space savings, this HDT file is self-queryable in contrast to the flat NTriples serialization.

3.1 Retrieving Urban Data from DBpedia

DBpedia contains a lot of descriptive data about cities, which we filter as follows: given the URI u of a city (*e.g.* `http://dbpedia.org/page/Verona`), we run a `CONSTRUCT` query on DBpedia which retrieves: i) all triples describing the city, *i.e.*, all triples comprising u as subject, and ii) all landmarks related to the city, *i.e.*, all resources (and their descriptions) linking to u. We restrict to resources of some kind of landmarks that we have manually identified, *e.g.* resources of type *Place* (`http://dbpedia.org/ontology/Place`), *Historical Buildings* (`http://dbpedia.org/ontology/HistoricPlace`), etc. Other types specifically related to the city are considered, for instance the *squares in Verona* (`http://dbpedia.org/class/yago/PiazzasInVerona`).

The RDF subgraph returned by this `CONSTRUCT` query is then converted to RDF/HDT and ready to be downloaded and queried by our mobile app.

[6] We use the Java RDF/HDT library: `https://github.com/rdfhdt/hdt-java`.

[7] Available at: `http://dataweb.infor.uva.es/project/hdtourist/?lang=en`.

3.2 Browsing Urban Data

HDTourist uses categories to organize and display data. The main menu comprises four categories: *description*, *demography and geography*, *attractions*, and *other interesting data*. Figure 1 (a) shows the *description* of Verona, which includes basic information about the city. The information showed in each category is defined as SPARQL templates in XML configuration files (one per category), such as the following one:

```
<?xml version="1.0" encoding="utf-8"?>
<category id="Attractions">
  <name>Attractions</name>
  <group>
    <name>Squares</name>
    <sparql>
    SELECT ?label
    WHERE
    { ?place rdf:type <http://dbpedia.org/class/yago/SquaresIn${CITY}> .
      ?place rdfs:label ?label .}
    UNION
    { ?place rdf:type <http://dbpedia.org/class/yago/PiazzasIn${CITY}> .
      ?place rdfs:label ?label .}
    }
    </sparql>
  </group>
  <group>
    <name>Buildings</name>
....
```

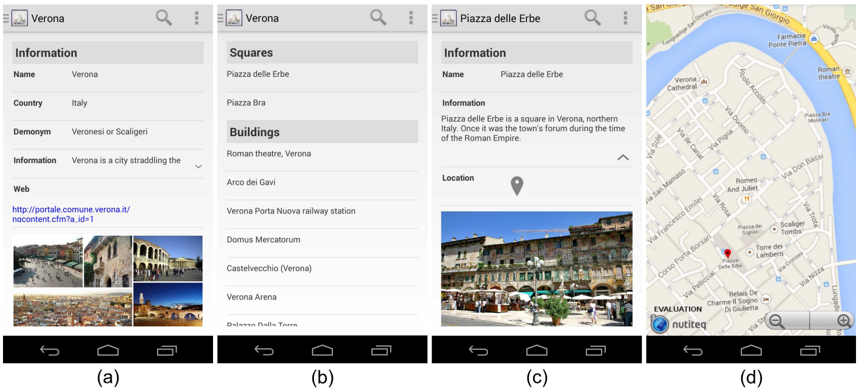
This XML excerpt corresponds to Figure 1 (b) showing the category "*Attractions*", which includes *Squares*, *Buildings*, etc. Each group retrieves the label of attractions with a SPARQL query which typically consists of a UNION of Basic Graph Patterns searching for certain types of resources, as shown in the excerpt. When parsing the XML, the template ${CITY} is converted to the appropriated name, *e.g. Verona*. Each SPARQL query is then resolved making use of the query API of RDF/HDT, retrieving the label shown in the screen layout.

As shown in Figure 1 (c), each landmark can be expanded, obtaining further information. In this screenshot, we choose the "Piazza delle Erbe" (within "*Squares*"), and the app retrieves the triples describing it. The concrete information to be shown in the landmark description is also configured by means of an XML file containing one SPARQL template per category, again resolved against the local RDF/HDT. As shown in the screenshot, pictures can be downloaded and stored offline. Finally, HDTourist is able to show geolocated landmarks in interactive maps, as shown in Figure 1 (d) for "Piazza delle Erbe". The app uses Google maps by default, but offline maps[8] can be downloaded beforehand.

4 Conclusions and Future Work

The offline capacities and structured information consumption possibilities of mobile devices are still several order of magnitudes below traditional devices. With our demo we show that RDF/HDT can be used as a self-contained engine to retrieve RDF information in mobile devices. To date, we have explored a given set of cities and certain query templates to build the screen layout. We are now exploring a spreading activation mechanism to automatically retrieve interesting

[8] In this prototype we use Nutiteq SDK Maps, available at `http://www.nutiteq.com/`.



(a)

Verona

Information

Name Verona

Country Italy

Demonym Veronesi or Scaligeri

Information Verona is a city straddling the

Web

http://portale.comune.verona.it/nqcontent.cfm?a_id=1

(b)

Verona

Squares

Piazza delle Erbe

Piazza Bra

Buildings

Roman theatre, Verona

Arco dei Gavi

Verona Porta Nuova railway station

Domus Mercatorum

Castelvecchio (Verona)

Verona Arena

Palazzo Dalla Torre

(c)

Piazza delle Erbe

Information

Name Piazza delle Erbe

Information

Piazza delle Erbe is a square in Verona, northern Italy. Once it was the town's forum during the time of the Roman Empire.

Location

(d)

Fig. 1. Some screenshots of HDTourist.

features of a city which are then converted to HDT on the server side. This also takes into account other datasets besides DBpedia.

Acknowledgments

This work has been funded by the European Commission under the grant PlanetData (FP7-257641) and by the Spanish Ministry of Economy and Competitiveness (TIN2013-46238-C4-2-R).

References

1. C. Becker and C. Bizer. DBpedia Mobile: A Location-Enabled Linked Data Browser. In *Proc. of LDOW,* CEUR-WS 369, paper 14, 2008.
2. A.E. Cano, A.S. Dadzie, and M. Hartmann. Who's Who–A Linked Data Visualisation Tool for Mobile Environments. In *Proc. of ISWC*, pages 451–455, 2011.
3. J. David, J. Euzenat, and M. Rosoiu. Linked Data from your Pocket. In *Proc. of DownScale,* CEUR-WS 844, paper 2, pages 6–13, 2012.
4. J.D. Fernández, M. Arias, M.A. Martínez-Prieto, and C. Gutiérrez. Management of Big Semantic Data. In *Big Data Computing*, chapter 4. Taylor & Francis, 2013.
5. J.D. Fernández, M.A. Martínez-Prieto, C. Gutiérrez, and A. Polleres. *Binary RDF Representation for Publication and Exchange (HDT).* W3C Member Submission, 2011. `http://www.w3.org/Submission/2011/03/`.
6. D. Le-Phuoc, J.X. Parreira, V. Reynolds, and M. Hauswirth. RDF On the Go: An RDF Storage and Query Processor for Mobile Devices. In *Proc. ISWC,* CEUR-WS 658, paper 19, 2010.
7. M.A. Martínez-Prieto, M. Arias, and J.D. Fernández. Exchange and Consumption of Huge RDF Data. In *Proc. of ESWC*, pages 437–452, 2012.
8. V.C. Ostuni, G. Gentile, T. Di Noia, R. Mirizzi, D. Romito, and E. Di Sciascio. Mobile Movie Recommendations with Linked Data. In *Proc. of CD-ARES*, pages 400–415, 2013.
9. G. Parra, J. Klerkx, and E. Duval. More!: Mobile Interaction with Linked Data. In *Proc. of DCI,* CEUR-WS 817, paper 4, 2010.