



## LarKC

*The Large Knowledge Collider:*

*a platform for large scale integrated reasoning and Web-search*

FP7 – 215535

---

# D6.11 – 5th periodic report on data and performances

---

**Coordinator: Daniele Dell’Aglio (CEFRIEL)**

**With contributions from:**

**Irene Celino and Emanuele Della Valle (CEFRIEL),**

**Ioan Toma (Softgress),**

**Silviu Bota and Ionel Giosan (UTCN),**

**Seon-Ho Kim and Tony Lee (Saltlux),**

**Yi Huang and Volker Tresp (SIEMENS)**

**Quality Assessor: Axel Tenschert (HLRS)**

**Quality Controller: Emanuele Della Valle (CEFRIEL)**

|                      |                                   |
|----------------------|-----------------------------------|
| Document Identifier: | LarKC/2008/D6.11                  |
| Class Deliverable:   | LarKC EU-IST-2008-215535          |
| Version:             | 1.0                               |
| Date:                | September 30 <sup>th</sup> , 2011 |
| State:               | Final                             |
| Distribution:        | Public                            |



## EXECUTIVE SUMMARY

This document updates D6.7 – “3rd periodic report on data and performances” with a description of additional data sources we analyzed in order to consider them in the development of our activities and our scenarios. Specifically, we describe the datasets used in the development of the Location-based Social Media Analysis application named BOTTARI.

We also continued to test the Urban Computing workflows built on LarKC in order to obtain some indications about the performance of the applications we are developing over the platform, especially in consideration of the latest updates to the LarKC platform. The document contains a further testing of the so called “Urban LarKC” application, which was ported to the platform version 2.5<sup>1</sup> specifically for its performance evaluation and the evaluation of some new plug-ins used in the workflows for the BOTTARI application.

---

<sup>1</sup> The newest version at the time we started the evaluation



## DOCUMENT INFORMATION

|                           |  |                |       |
|---------------------------|--|----------------|-------|
| <b>IST Project Number</b> | FP7 - 215535   | <b>Acronym</b> | LarKC |
| <b>Full Title</b>         | The Large Knowledge Collider: a platform for large scale integrated reasoning and Web-search |                |       |
| <b>Project URL</b>        | http://www.larkc.eu/   |                |       |
| <b>Document URL</b>       |  |                |       |
| <b>EU Project Officer</b> | Stefano Bertolo  |                |       |

|                     |               |      |              |  |
|---------------------|---------------|------|--------------|--|
| <b>Deliverable</b>  | <b>Number</b> | 6.11 | <b>Title</b> | 5th periodic report on data and performances |
| <b>Work Package</b> | <b>Number</b> | 6    | <b>Title</b> | Urban Computing                              |

|                            |  |     |               |     |
|----------------------------|--|-----|---------------|-----|
| <b>Date of Delivery</b>    | <b>Contractual</b>   | M42 | <b>Actual</b> | M42 |
| <b>Status</b>              | version 1.0  |     | final ■       |     |
| <b>Nature</b>              | prototype <input type="checkbox"/> report ■ dissemination <input type="checkbox"/> |     |               |     |
| <b>Dissemination level</b> | public ■ consortium <input type="checkbox"/>                                       |     |               |     |

|                           |  |                    |               |  |
|---------------------------|--|--------------------|---------------|--|
| <b>Authors (Partner)</b>  | Daniele Dell'Aglio, Irene Celino and Emanuele Della Valle (CEFRIEL), Ioan Toma (Softgress), Silviu Bota and Ionel Giosan (UTCN), Seon-Ho Kim and Tony Lee (Saltlux), Yi Huang and Volker Tresp (SIEMENS) |                    |               |  |
| <b>Responsible Author</b> | <b>Name</b>  | Daniele Dell'Aglio | <b>E-mail</b> | <a href="mailto:daniele.dellaglio@cefriel.it">daniele.dellaglio@cefriel.it</a> |
|                           | <b>Partner</b>   | CEFRIEL            | <b>Phone</b>  | +390223945243  |

|                                     |  |
|-------------------------------------|--|
| <b>Abstract (for dissemination)</b> | This document is the fourth and final step of reporting about the collection of data sources and the evaluation results of the performance of the early Urban Computing demonstrators. |
| <b>Keywords</b>                     | data sets, measurements, tests, performances, stress tests, evaluation, periodic report, use case, urban computing   |

| <b>Version Log</b> |                 |               |  |
|--------------------|-----------------|---------------|--|
| <b>Issue Date</b>  | <b>Rev. No.</b> | <b>Author</b> | <b>Change</b>                            |
| August 23          | 1               | Irene         | Initialization of the document           |
| September 7        | 2               | Daniele       | Contribution to Chapter 3                |
| September 9        | 3               | Seonho        | Contribution to Chapter 2                |
| September 13       | 4               | Yi            | Contribution to Chapter 3 and Appendix B |
| September 20       | 5               | Ioan          | Contribution to Chapter 3 and Appendix A |
| September 23       | 6               | Irene         | Contribution to Chapter 1 and 4          |
| September 26       | 7               | Daniele       | Addressed Axel's comments                |
| September 29       | 8               | Ioan          | Addressed Axel's comments                |



## PROJECT CONSORTIUM INFORMATION

| Participant's name   | Partner  | Contact  |
|--|--|--|
| Semantic Technology Institute<br>Innsbruck, Universitaet Innsbruck   | <br> | Dieter Fensel,<br>Semantic Technology Institute (STI),<br>Universitaet Innsbruck,<br>Innsbruck, Austria,<br>E-mail: dieter.fensel@sti-innsbruck.at |
| AstraZeneca AB   |   | Bosse Andersson,<br>AstraZeneca<br>Lund, Sweden<br>Email:<br>bo.h.andersson@astrazeneca.com  |
| CEFRIEL - SOCIETA<br>CONSORTILE A<br>RESPONSABILITA LIMITATA         |   | Emanuele Della Valle,<br>CEFRIEL - SOCIETA CONSORTILE<br>A RESPONSABILITA LIMITATA,<br>Milano, Italy,<br>Email: emanuele.dellavalle@cefriel.it     |
| CYCROP, RAZISKOVANJE IN<br>EKSPERIMENTALNI RAZVOJ<br>D.O.O.          |   | Michael Witbrock,<br>CYCROP, RAZISKOVANJE IN<br>EKSPERIMENTALNI RAZVOJ<br>D.O.O.,<br>Ljubljana, Slovenia,<br>Email: witbrock@cyc.com               |
| Höchstleistungsrechenzentrum,<br>Universitaet Stuttgart              |   | Georgina Gallizo,<br>Höchstleistungsrechenzentrum,<br>Universitaet Stuttgart,<br>Stuttgart, Germany,<br>Email: gallizo@hlrs.de                     |
| MAX-PLANCK GESELLSCHAFT<br>ZUR FOERDERUNG DER<br>WISSENSCHAFTEN E.V. |   | Lael Schooler,<br>Max-Planck-Institut für<br>Bildungsforschung<br>Berlin, Germany<br>Email: schooler@mpib-berlin.mpg.de                            |
| Ontotext AD  |   | Atanas Kiryakov,<br>Ontotext Lab,<br>Sofia, Bulgaria<br>Email: naso@ontotext.com   |
| SALTLUX INC.   |   | Kono Kim,<br>SALTLUX INC,<br>Seoul, Korea,   |



|   |  |  |
|---|--|--|
|   |  | Email: kono@saltlux.com  |
| SIEMENS<br>AKTIENGESELLSCHAFT   |  | Volker Tresp,<br>SIEMENS<br>AKTIENGESELLSCHAFT,<br>Muenchen, Germany,<br>E-mail: volker.tresp@siemens.com                        |
| THE UNIVERSITY OF SHEFFIELD   |  | Hamish Cunningham,<br>THE UNIVERSITY OF SHEFFIELD<br>Sheffield, UK,<br>Email: h.cunningham@dcs.shef.ac.uk                        |
| VRIJE UNIVERSITEIT<br>AMSTERDAM   |  | Frank van Harmelen,<br>VRIJE UNIVERSITEIT<br>AMSTERDAM,<br>Amsterdam, Netherlands,<br>Email: Frank.van.Harmelen@cs.vu.nl         |
| THE INTERNATIONAL WIC<br>INSTITUTE, BEIJING<br>UNIVERSITY OF TECHNOLOGY |  | Ning Zhong,<br>THE INTERNATIONAL WIC<br>INSTITUTE,<br>Mabeshi, Japan,<br>Email: zhong@maebashi-it.ac.jp                          |
| INTERNATIONAL AGENCY FOR<br>RESEARCH ON CANCER                          |  | Paul Brennan,<br>INTERNATIONAL AGENCY FOR<br>RESEARCH ON CANCER,<br>Lyon, France,<br>Email: brennan@iarc.fr                      |
| INFORMATION RETRIEVAL<br>FACILITY                                       |  | John Tait,<br>INFORMATION RETRIEVAL<br>FACILITY<br>Vienna, Austria<br>Email : john.tait@ir-facility.org                          |
| UNIVERSITATEA TEHNICA<br>CLUJ-NAPOCA                                    |  | Sergiu Nedeveschi,<br>UNIVERSITATEA TEHNICA<br>CLUJ-NAPOCA,<br>Cluj-Napoca, Romania,<br>Email:<br>Sergiu.Nedeveschi@cs.utcluj.ro |
| SOFTGRESS S.R.L.  |  | Ioan Toma,<br>SOFTGRESS S.R.L.,<br>Cluj-Napoca, Romania,<br>Email: ioan.toma@softgress.com                                       |





## **TABLE OF CONTENTS**

|  |           |
|--|-----------|
| <b>LIST OF FIGURES .....</b>                                   | <b>8</b>  |
| <b>LIST OF TABLES .....</b>                                    | <b>9</b>  |
| <b>LIST OF ACRONYMS.....</b>                                   | <b>10</b> |
| <b>1. INTRODUCTION .....</b>                                   | <b>11</b> |
| <b>2. PERIODIC REPORT ON DATA .....</b>                        | <b>12</b> |
| 2.1. TWITTER DATASET USED IN BOTTARI.....                      | 12        |
| 2.2. POIS DATASET USED IN BOTTARI .....                        | 14        |
| <b>3. PERIODIC REPORT ON PERFORMANCES .....</b>                | <b>15</b> |
| 3.1. URBAN LARKC INSTRUMENTATION AND TESTING .....             | 15        |
| 3.2. BOTTARI RECOMMENDATIONS USING SUNS .....                  | 26        |
| <b>4. CONCLUSION .....</b>                                     | <b>29</b> |
| <b>A. APPENDIX A – URBAN LARKC WORKFLOWS DESCRIPTIONS.....</b> | <b>30</b> |
| <b>B. APPENDIX B - NDCG.....</b>                               | <b>32</b> |
| <b>5. REFERENCES .....</b>                                     | <b>33</b> |



## List of Figures

|   |    |
|---|----|
| Figure 1 - Free and allocated memory for the platform .....   | 18 |
| Figure 2 - User memory by the platform and its uptime .....   | 19 |
| Figure 3 - The query total response time and the number of garbage collection operations performed .....        | 20 |
| Figure 4 - Memory used by the UrbanPathFinder workflow before and after workflow execution .....                | 21 |
| Figure 5 - CPU usage vs average CPU usage of the platform.....  | 21 |
| Figure 6 - Used vs allocated memory for the platform .....  | 22 |
| Figure 7 - Average total response time of the plugins that are part of UrbanEvent workflow .....                | 22 |
| Figure 8 - Average use of memory of the plugins that are part of UrbanEvent workflow.....                       | 23 |
| Figure 9 - ThreadUserCPUTime of the plugins that are part of UrbanEvent workflow .....                          | 23 |
| Figure 10 - System network traffic received and sent.....   | 24 |
| Figure 11 - System total free vs total used memory .....  | 24 |
| Figure 12 - TotalCPUTime used by platform.....  | 24 |
| Figure 13 - Free memory available for the platform .....  | 24 |
| Figure 14 - Total response for UrbanMonument plugins when running the query with the biggest response time..... | 25 |
| Figure 15 - Query total response .....  | 25 |
| Figure 16: Left: Number of ratings of each POI. Right: Number of ratings of each user. ....                     | 27 |
| Figure 17: Left: nDCG scores. Right: Accuracy values at top N. ....   | 28 |



## List of Tables

|   |    |
|---|----|
| Table 1 - Sample UrbanPathFinder query in SPARQL..... | 16 |
| Table 2 - Sample UrbanEvent query in SPARQL.....      | 16 |
| Table 3 - Sample UrbanMonument query in SPARQL.....   | 16 |
| Table 4 - The statistics of the data set .....        | 27 |



## List of Acronyms

| <u>Acronym</u> | <u>Description</u>             |
|----------------|--------------------------------|
| RDF            | Resource Description Framework |
| LarKC          | Large Knowledge Collider       |
| POI            | Point Of Interest              |



## 1. Introduction

This deliverable represents the fourth periodic report on data and performance of the Urban Computing use case in LarKC. It is based on the templates provided in D6.2 [1] and it follows the results presented in D6.4 [2], D6.6 [3] and D6.7 [4]. It includes the description of the newly acquired data sources and it describes the performance tests on the developed Urban Computing demonstrators.

The objective of this deliverable, as its title suggests, is twofold.

On the one hand, this report is aimed at providing an update on the list of data sources related to the Urban Computing use case. In particular, we describe the new data sources which are being used in the development of the Korean BOTTARI application, a Location-based Social Media Analysis demonstrator. More details about BOTTARI, the developed workflows and the new plug-ins used in its development are given in the companion deliverable D6.10 [10].

On the other hand, this deliverable reports on the tests we performed on the Urban Computing demonstrators built on the available LarKC platform released and developed within WP6. Those tests are useful to understand possible bottlenecks, to suggest possible improvements, to derive scientific or technological challenges to propose to other LarKC technical work-packages. Specifically, we report the performance assessment of the “Urban LarKC” application – specifically ported to the platform version 2.5 for this testing campaign – and the BOTTARI demonstrator. We executed the tests on the platform 2.5 instead the 2.6 because we started the development in April (when 2.5 was released) and we have been following a plan with tight schedule; thus, we kept working with 2.5 in order to minimize the delays caused by potential new bugs introduced by the concurrent development of version 2.6.

The deliverable is structured as follows. Chapter 2 describes the new data sources as per the template used also in the previous deliverables. Chapter 3 presents our tests in terms of the adopted methodology, the tested demonstrators, the results of this testing and the interpretation we can give to those results. Finally, the conclusions are offered in Chapter 4.



## 2. Periodic report on data

This chapter describes the datasets used in the BOTTARI Location-based Social Media Analysis application built on the LarKC platform. Details on the application scenario are given in D6.10 [10].

### 2.1. Twitter Dataset used in BOTTARI

|   |  |                                |
|---|--|--------------------------------|
| <b>Data Source: Twitter</b>                   |  |                                |
| <b>Report ID</b>                              |  |                                |
| <b>Section 1</b>                              | <b>Data source metadata</b>  |                                |
| <b>Name</b>                                   | Twitter  |                                |
| <b>Producer/Owner</b>                         | Twitter  |                                |
| <b>Description</b>                            | 1. Data gathered from twitter which contains: <ul style="list-style-type: none"> <li>• tweets</li> <li>• twitter users</li> </ul> 2. Reputations of POIs mentioned in tweets |                                |
| <b>Namespace/Web Address</b>                  | <a href="http://www.twitter.com">http://www.twitter.com</a>  |                                |
| <b>Availability</b>                           | Anyone can freely access to the twitter data but 350 authenticated calls (when signed in) are permitted per hour, and a search can only return 1,500 results at a time.      |                                |
| <b>Download/Upload/Acquisition date</b>       | Continuing from April, 2011  |                                |
| <b>Version</b>                                | -  |                                |
| <b>Physical size</b>                          | Increasing steadily  |                                |
| <b>Nature of data type</b>                    | Social data stream   |                                |
| <b>Quality of the data source</b>             | -  |                                |
| <b>Section 2</b>                              | <b>“semantics” of the data source</b>  |                                |
| <b>Typology of data</b>                       | OWL Ontology   |                                |
| <b>Geographic coverage of data</b>            | -  |                                |
| <b>Applied systems</b>                        | SOR repository   |                                |
| <b>Existence of schema/ontology</b>           | <a href="http://svn.larkc.eu/wp6/LB_SMA_SampleData/Lyon_LB_SMA/Ontology/">http://svn.larkc.eu/wp6/LB_SMA_SampleData/Lyon_LB_SMA/Ontology/</a>                                |                                |
| <b>Existing links with other data-sources</b> | -  |                                |
| <b>Possible linkage to other data-sources</b> | -  |                                |
| <b>Scale of data</b>                          | over 0.6 billion (increasing steadily)   |                                |
| <b>Section 3</b>                              | <b>Data source format</b>  |                                |
| <b>Format of data</b>                         | Social data stream   |                                |
| <b>Generation method</b>                      | using JENA, SOR APIs to generate triples   |                                |
| <b>Support query language</b>                 | Twitter API  | <b>Total no. of statements</b> |
| <b>Support triple type</b>                    | -  |                                |
| <b>No. of explicit statements</b>             | -  |                                |



|   |  |
|---|--|
| <b>Noise, Uncertainty and inconsistency of data</b> | -  |
| <b>Remarks</b>                                      |  |
|   | <ol style="list-style-type: none"><li>1. We are gathering twitter data continually and transforming the data to triples.</li><li>2. Data of reputations are being added upon increasing of twitter data</li><li>3. This dataset is stored in the same repository with POIs dataset</li></ol> |



**2.2. POIs Dataset used in BOTTARI**

| <b>Data Source: POIs (Point of Interest)</b>        |   |                                |               |
|---|---|--------------------------------|---------------|
| <b>Report ID</b>                                    |   |                                |               |
| <b>Section 1</b>                                    | <b>Data source metadata</b>   |                                |               |
| <b>Name</b>   | POIs  |                                |               |
| <b>Producer/Owner</b>                               | Saltlux   |                                |               |
| <b>Description</b>                                  | details of POIs constructed by surveying within specific area called Insa-dong in Seoul, Korea  |                                |               |
| <b>Namespace/Web Address</b>                        | <a href="http://www.saltlux.com/geospatial">http://www.saltlux.com/geospatial</a>   |                                |               |
| <b>Availability</b>                                 | Available to access via SPARQL endpoint   |                                |               |
| <b>Download/Upload/Acquisition date</b>             | March, 2011   |                                |               |
| <b>Version</b>                                      | -   |                                |               |
| <b>Physical size</b>                                | 15MB  |                                |               |
| <b>Nature of data type</b>                          | Static data   |                                |               |
| <b>Quality of the data source</b>                   | Good  |                                |               |
| <b>Section 2</b>                                    | <b>“semantics” of the data source</b>   |                                |               |
| <b>Typology of data</b>                             | Topology  |                                |               |
| <b>Geographic coverage of data</b>                  | Part of Seoul (Insa-dong)   |                                |               |
| <b>Applied systems</b>                              | SOR repository  |                                |               |
| <b>Existence of schema/ontology</b>                 | <a href="http://svn.larkc.eu/wp6/LB_SMA_SampleData/Lyon_LB_SMA/Ontology/">http://svn.larkc.eu/wp6/LB_SMA_SampleData/Lyon_LB_SMA/Ontology/</a> |                                |               |
| <b>Existing links with other data-sources</b>       | -   |                                |               |
| <b>Possible linkage to other data-sources</b>       | -   |                                |               |
| <b>Scale of data</b>                                | 319 POIs and each of them has 44 attributes   |                                |               |
| <b>Section 3</b>                                    | <b>Data source format</b>   |                                |               |
| <b>Format of data</b>                               | Triple repository (SOR)   |                                |               |
| <b>Generation method</b>                            | SOR APIs  |                                |               |
| <b>Support query language</b>                       | SPARQL  | <b>Total no. of statements</b> | Around 21,200 |
| <b>Support triple type</b>                          | RDF, OWL, Ntriple   |                                |               |
| <b>No. of explicit statements</b>                   | Around 21,200   |                                |               |
| <b>Noise, Uncertainty and inconsistency of data</b> | -   |                                |               |
| <b>Remarks</b>                                      |   |                                |               |
|   | This dataset is stored in the same repository with Twitter dataset  |                                |               |



### 3. Periodic report on performances

#### 3.1. Urban LarKC instrumentation and testing

##### 3.1.1. Test goals

To assess the performance of the WP6 plugins and workflows, as well as the performance of the LarKC platform, a set of experiments were performed using the instrumented versions of some WP6 plugins and workflows. WP11 tools were used to instrument and collect monitoring information about WP6 plugins, workflows and the platform. Furthermore WP11 visualization enabled WP6 plugins and workflows developers to quickly analyse how their components perform and, based on this, to adjust and tune their components in order to increase their performance.

Execution time, CPU usage, number of used threads are some of the dimensions for which data was measured and collected from WP6 plugins, workflows and the platform. The exact set of the considered WP11 metrics are described in the results section.

##### 3.1.2. Instrumented test case

The experiments were performed using the following WP6 workflows:

1. *UrbanPathFinder workflow* which includes the following plugins: SourceSplitter, RemoteGraphLoaderIdentifier and OpResPathFinderReasoner
2. *UrbanEvent workflow* which includes the following plugins: SourceSplitter, SparqlToCityQueryTransformer, EventIdentifier, XML2RDFTransformer and SparqlQueryEvaluationReasoner
3. *UrbanMonument workflow* which includes the following plugins: SourceSplitter, UrbanSindiceIdentifier and SparqlQueryEvaluationReasoner

The RDF description of each workflow for the LarKC platform release 2.5 is given in Appendix A.

##### 3.1.3. Methodology

This section describes the methodology used to performed the experiments for each workflow we tested. For each workflow we indicate the number of queries executed. Queries are sent to the LarKC platform where the workflows are deployed in a sequential order.

For the UrbanPathFinder workflow 1300 queries were executed. A dedicated script was used to submit the queries automatically. An example of query for UrbanPathFinder workflow is given in Table 1. The queries are similar in structure. The parameters that were varied are the node ids (in the following example node8252 and node8253). Different combinations of pairs of nodes have been generated using the complete list of nodes. Metrics were collected for about one hour interval.

```
1. PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns/>
2. PREFIX lud: <http://www.linkingurbandata.org/onto/ama/>
3. SELECT ?p ?w ?n1 ?l ?n2
4. WHERE {
5.     ?p rdf:type lud:Paths.
6.     ?p                                     lud:pathFrom
       <http://seip.cefriel.it/ama/resource/nodes/node8252>.
7.     ?p                                     lud:pathTo
       <http://seip.cefriel.it/ama/resource/nodes/node8253>.
8.     ?p lud:contain ?l.
9.     ?l lud:from ?n1.
10.    ?l lud:to ?n2.
11.    ?p lud:pathWeight ?w.
12. }
13. ORDER BY ?w
```



**Table 1 - Sample UrbanPathFinder query in SPARQL**

For the UrbanEvent workflow 100 queries were executed. An example of query for UrbanEvent workflow is given in Table 2. The queries are similar in structure. The parameter that were varied is the name of the city (in the following example "Milan").

```

1. PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
2. PREFIX rdfcal: <http://www.w3.org/2002/12/cal/icaltzd#>
3. PREFIX addr: <http://schemas.talis.com/2005/address/schema#>
4. PREFIX geo: <http://www.w3.org/2003/01/geo/wgs84_pos#>
5. PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
6. PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
7. PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
8. SELECT ?e ?s ?summary ?cat ?desc ?l ?lab ?lat ?lng
9. WHERE{
10.   ?e rdf:type rdfcal:Vevent.
11.   ?e rdfcal:summary ?summary.
12.   ?e skos:subject ?cat.
13.   ?e rdfcal:description ?desc.
14.   ?e geo:location ?l.
15.   ?l rdfs:label ?lab.
16.   ?l geo:lat ?lat.
17.   ?l geo:long ?lng.
18.   ?e rdfcal:dtstart ?s .
19.   ?l addr:localityName "Milan".
20.   FILTER(?s > xsd:dateTime("2011-03-30T00:00:00Z")
21.     && ?s < xsd:dateTime("2011-04-04T23:59:59Z")).
22. }
    
```

**Table 2 - Sample UrbanEvent query in SPARQL**

For the UrbanMonument workflow one query was run multiple times. The query is given in Table 3.

```

1. PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
2. PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
3. PREFIX foaf: <http://xmlns.com/foaf/0.1/>
4. PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
5. PREFIX georss: <http://www.georss.org/georss/>
6. PREFIX dcterms: <http://purl.org/dc/terms/>
7. SELECT DISTINCT ?monument ?geopoint ?img ?wiki ?name ?desc
8. WHERE{
9.   {
10.    {?monument dcterms:subject ?subject.
11.     ?subject skos:broader
12.     <http://dbpedia.org/resource/Category:Visitor_attractions_in_Milan>
13.    }
14.   UNION
15.    {?monument dcterms:subject
16.     <http://dbpedia.org/resource/Category:Visitor_attractions_in_Milan>.
17.    }
18.   }
19.   ?monument georss:point ?geopoint.
20.   ?monument foaf:depiction ?img.
21.   ?monument foaf:page ?wiki.
22.   ?monument rdfs:label ?name.
23.   ?monument rdfs:comment ?desc.
24.   FILTER ( lang(?name) = "en" && lang(?desc) = "en" )
25. }
    
```

**Table 3 - Sample UrbanMonument query in SPARQL**



In the experiments the following metrics for query, plugin, workflow, platform and the system(s) where the platform is running are considered. These metrics have been defined in D11.1.2 [9]. The most up-to-date formal definition of the metrics is available at: <https://github.com/semantic-im/sim-server/blob/master/src/main/resources/IMOntology-new.owl>

1. *Platform metrics*: PlatformTotalCPUTime, PlatformThreadsCount, PlatformThreadsStarted, PlatformTotalThreadsStarted, PlatformGccTime, PlatformTotalGccTime, PlatformFreeMemory, PlatformUnallocatedMemory, PlatformAllocatedMemory, PlatformUsedMemory, PlatformUptime, PlatformCPUTime, PlatformAvgCPUUsage, PlatformTotalGccCount, PlatformCPUUsage, PlatformGccCount, PlatformCPUUsage,
2. *Query metrics*: QueryThreadGccTime, QueryThreadGccCount, QueryTotalResponseTime, QueryThreadBlockCount, QueryThreadBlockTime, QueryThreadBlockCount, QueryResultSize, QueryThreadTotalCPUTime, QueryThreadSystemCPUTime.
3. *Workflow metrics*: WorkflowTotalResponseTime, WorkflowUsedMemoryBefore, WorkflowUsedMemoryAfter, WorkflowUnallocatedMemoryBefore, WorkflowUnallocatedMemoryAfter, WorkflowFreeMemoryBefore, WorkflowFreeMemoryAfter.
4. *Plugin metrics*: PluginUnallocatedMemoryBefore, PluginUnallocatedMemoryAfter, PluginUsedMemoryBefore, PluginUsedMemoryAfter, PluginAllocatedMemoryBefore, PluginAllocatedMemoryAfter, PluginThreadWaitCount, PluginThreadWaitTime, PluginThreadTotalCPUTime, PluginThreadSystemCPUTime, PluginThreadGccTime, PluginThreadGccCount, PluginThreadCount, PluginThreadBlockCount, PluginThreadBlockTime, PluginProcessTotalCPUTime, PluginInputSizeInTriples, PluginOutputSizeInTriples, PluginTotalResponseTime.
5. *System metrics*: SystemCPUTime, SystemIdleCPULoad, SystemIORead, SystemIdleCPUTime, SystemIOWrite, SystemIrqCPULoad, SystemIrqCPUTime, SystemNetworkReceied, SystemNetworkSent, SystemOpenFileDesrCount, SystemProcessCount, SystemRunningProcessCount, SystemSwapIn, SystemSwapOut, SystemTcpInbound, SystemTcpOutbound, SystemThreadCount, SystemTotalFreeMemory, SystemTotalUsedMemory, SystemTotalUsedSwap, SystemUserCPULoad, SystemUserCPUTime.

#### 3.1.4. Environment

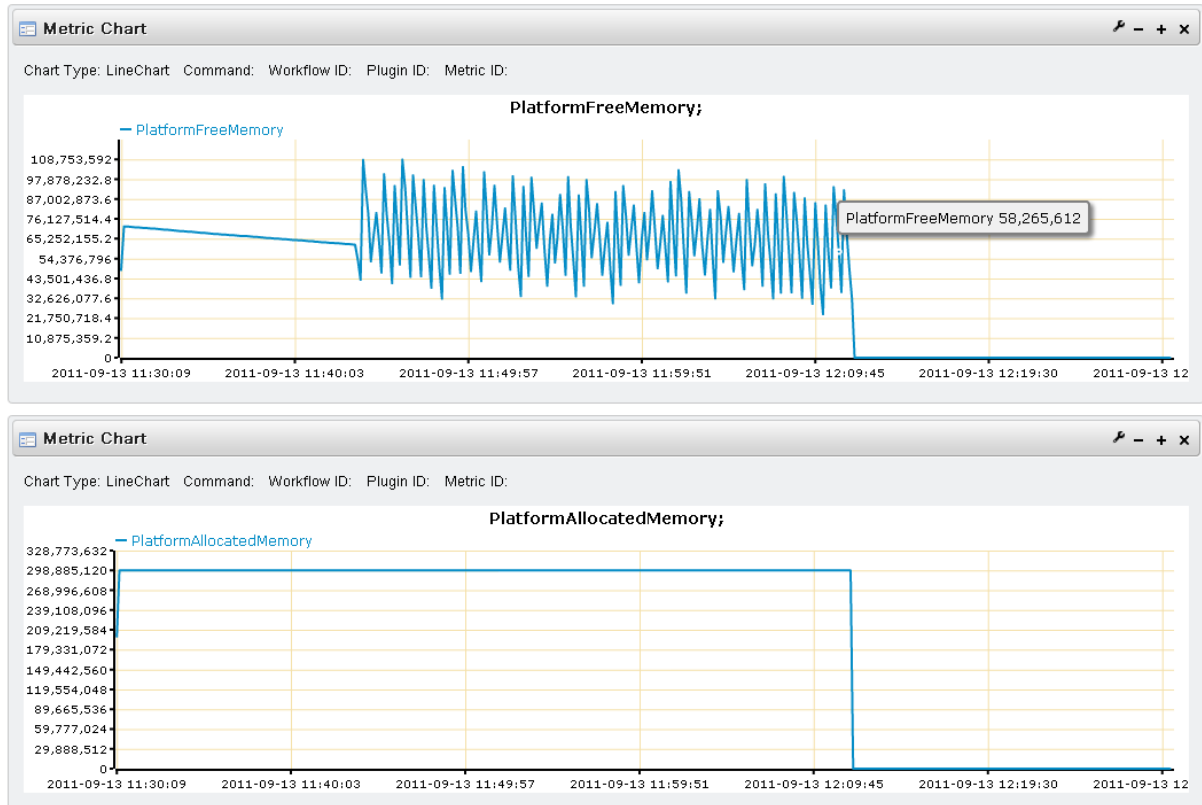
All the queries and the workflows have been run on version 2.5 of the platform, commit revision 1797. We run the experiments on a 32-bit Windows machine, Core 2 Quad@2.66 GHz, with 4GB RAM.

#### 3.1.5. Instrumentation results

In the rest of the section we visualize some of the metrics mentioned before for each of performed experiment.

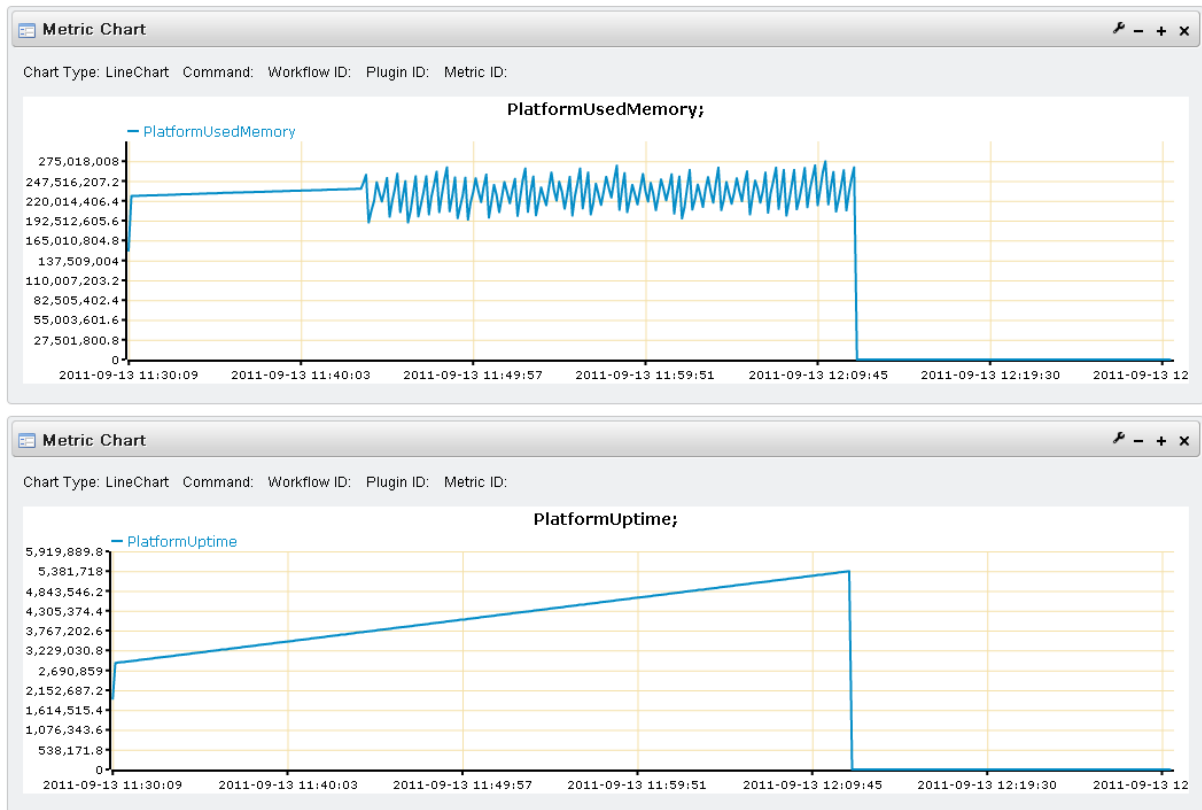
For the UrbanPathFinder experiments, platform measurements are illustrated in Figure 1 and Figure 2. The memory usage illustrated in all the figures in Section 3.1 is measured in bytes.

Figure 1 shows the platform free memory and its variation (min, max) in time. The free memory is oscillating around a constant value that basically means there are no memory leaks.



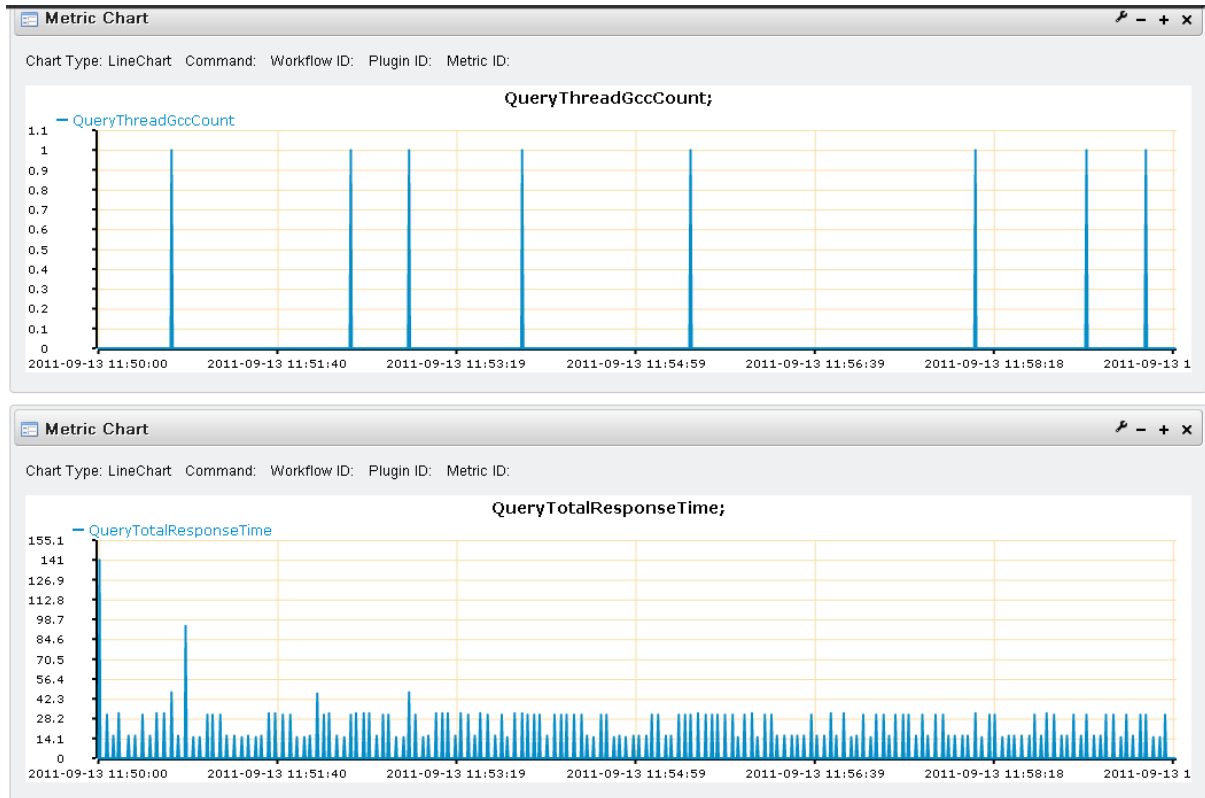
**Figure 1 - Free and allocated memory for the platform**

Figure 2 shows how much memory the JVM has allocated in time. The free memory, correlated with the used memory gives an indication of how loaded the JVM memory is. Figure 2 shows as well the uptime of the platform. If it grows linearly that means the platform is running continuously, if it drops to zero, that means the platform was stopped.



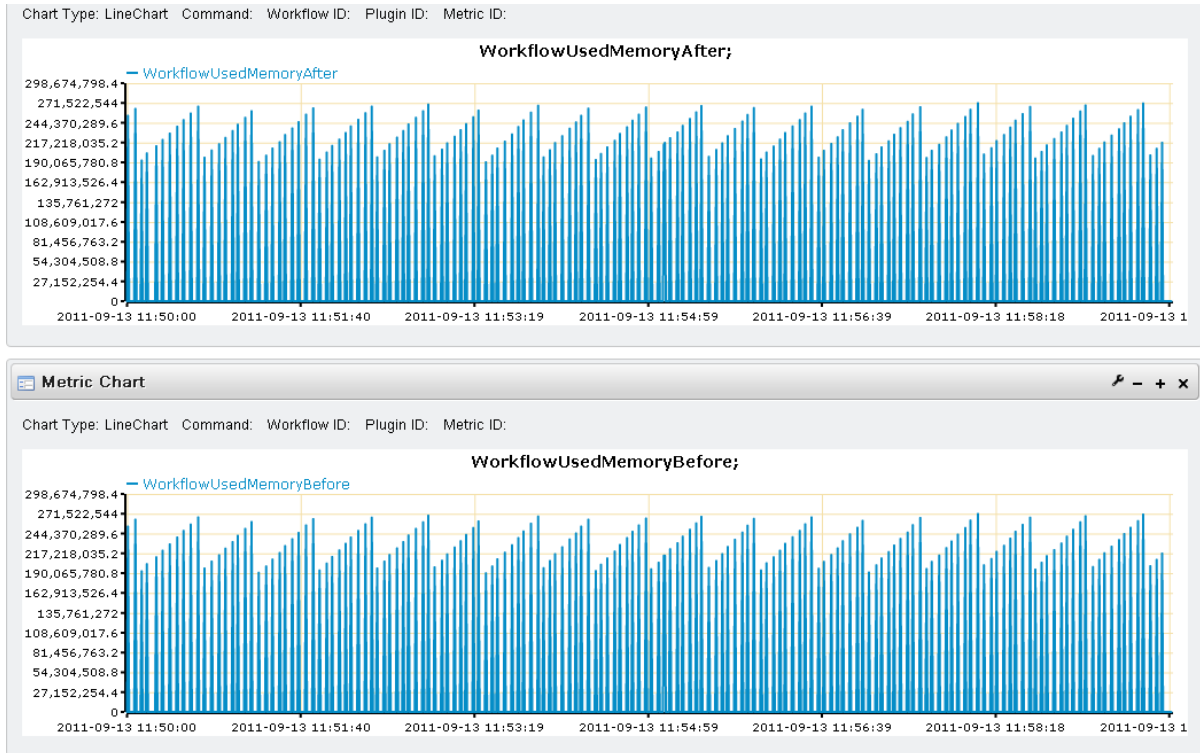
**Figure 2 - User memory by the platform and its uptime**

Query measurements are available in Figure 3. A spike in QueryThreadGccCount indicates that there is lost time on garbage collection process. In the same figure we can observe the QueryTotalResponseTime.



**Figure 3 - The query total response time and the number of garbage collection operations performed**

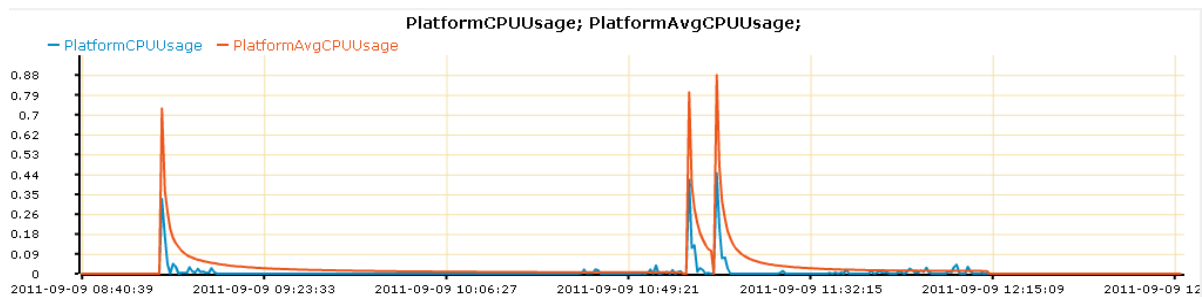
Workflow measurements are shown in Figure 4. It shows the memory used before and after the execution of the workflow. The values of these metrics grow before a garbage collection is performed.



**Figure 4 - Memory used by the UrbanPathFinder workflow before and after workflow execution**

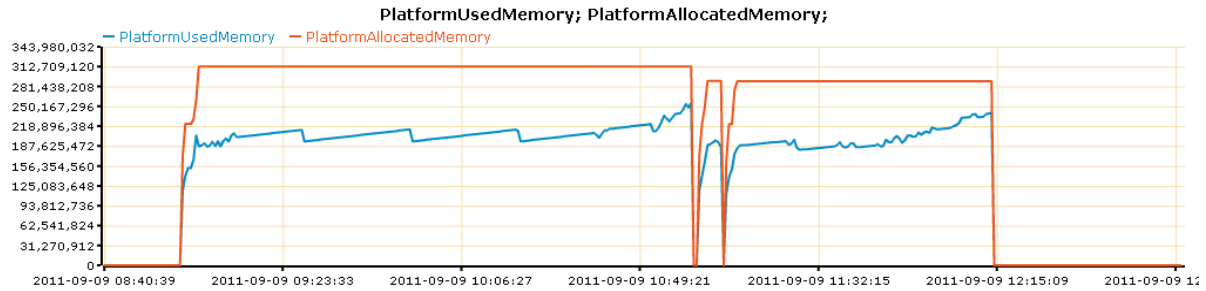
For the UrbanEvent experiments, platform measurements are available in Figure 5 and Figure 6.

In Figure 5 the relative low CPU usage of the platform. There are however spikes when the platform is started or stopped.



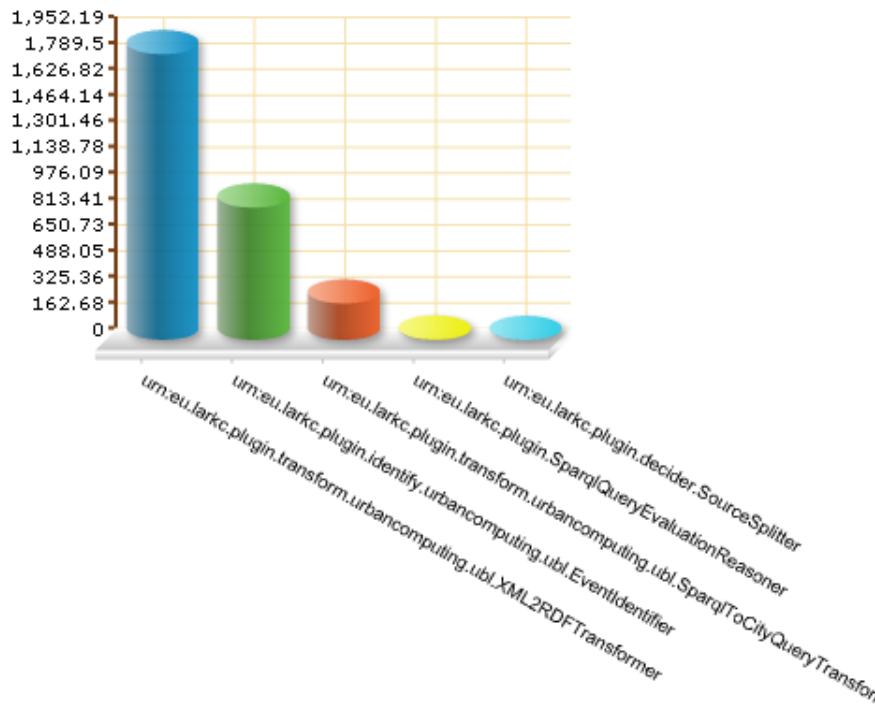
**Figure 5 - CPU usage vs average CPU usage of the platform**

Similarly, as shown in Figure 6, the memory usage of the platform is constant, thus no leaks. Variations can be observed when the platform is started or stopped.



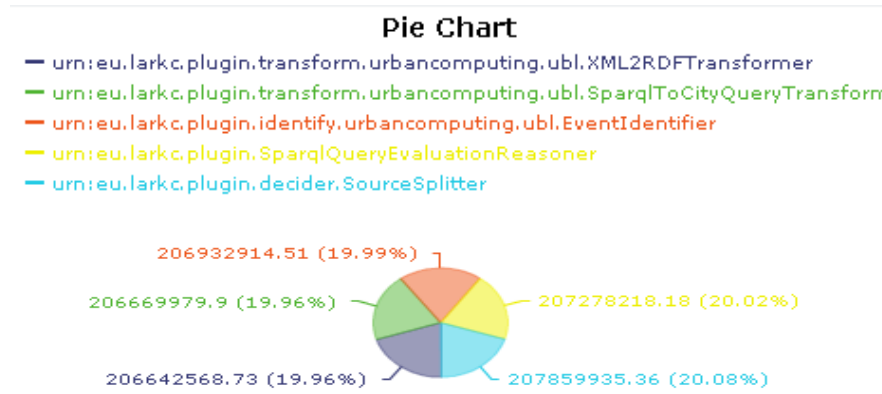
**Figure 6 - Used vs allocated memory for the platform**

Plugin measurements are available in Figure 7, Figure 8 and Figure 9. In Figure 7 we can observe which plugins from the UrbanEvent workflow are fast and which are slow. In this case XML2RDF and EventIdentifier are the most time to respond.



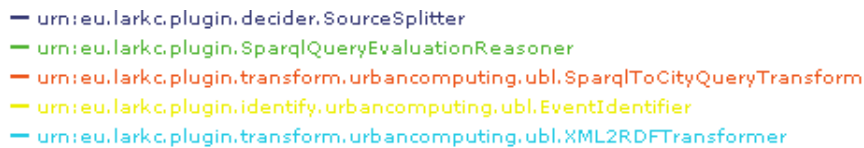
**Figure 7 - Average total response time of the plugins that are part of UrbanEvent workflow**

Figure 8 is about the average use of memory of the plugins that are part of UrbanEvent workflow. In this case all the plugins consume roughly the same amount of memory. . For the total response time, as well as other time related metrics, the measurement unit is microseconds.



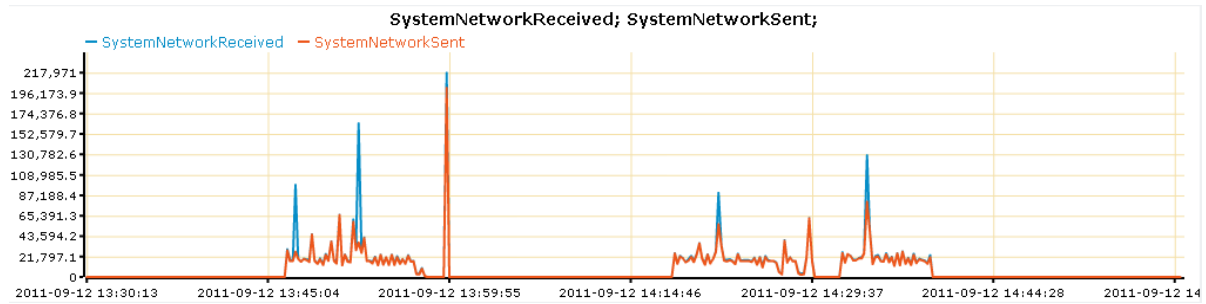
**Figure 8 - Average use of memory of the plugins that are part of UrbanEvent workflow**

In Figure 9 one can see the distribution of ThreadUserCPUTime and implicitly the distribution amount plugins of the CPU usage.



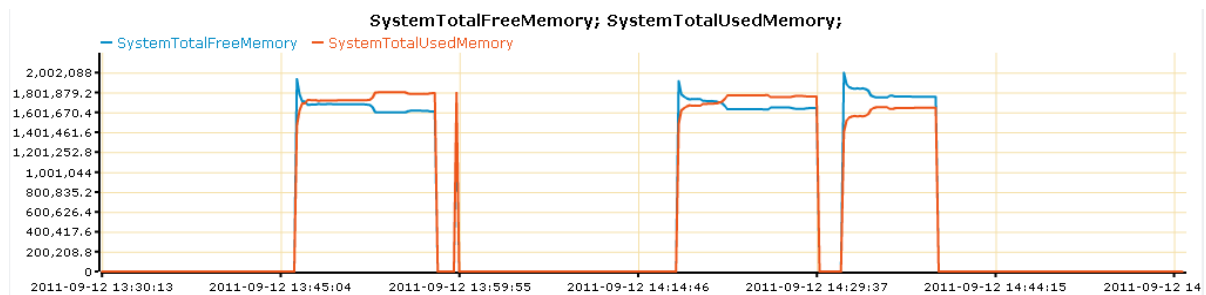
**Figure 9 - ThreadUserCPUTime of the plugins that are part of UrbanEvent workflow**

Figure 10 shows the variation in network traffic, which in this case is relatively small. . In the figure one can notice picks in network traffics that are due to some of the plugins that do network traffic in a discontinuous way during the execution of the workflow.



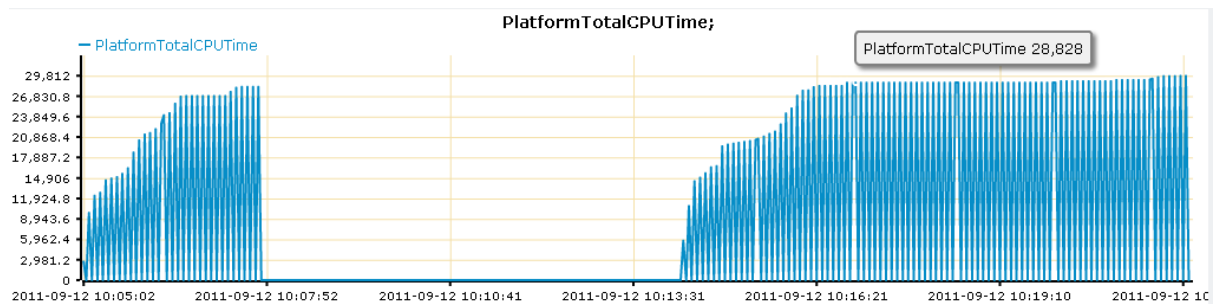
**Figure 10 - System network traffic received and sent**

Figure 11 shows the system total free and used memory.

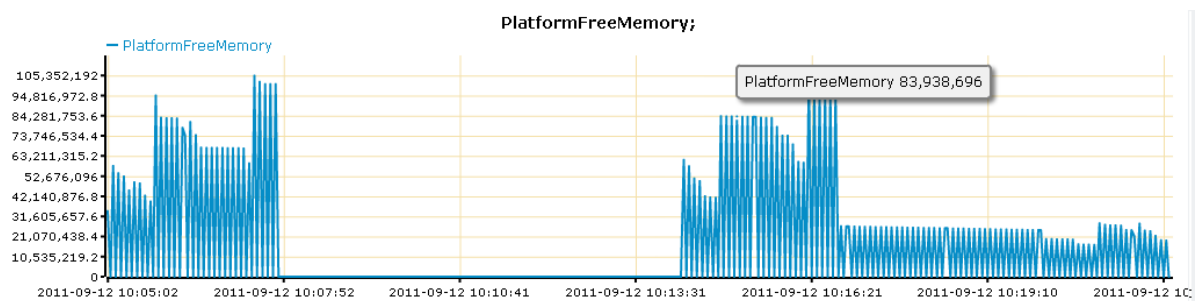


**Figure 11 - System total free vs total used memory**

For the UrbanMonument experiments, platform measurements are available in Figure 12 and Figure 13. In these figures one can notice gaps that correspond to inactivity of the platform.

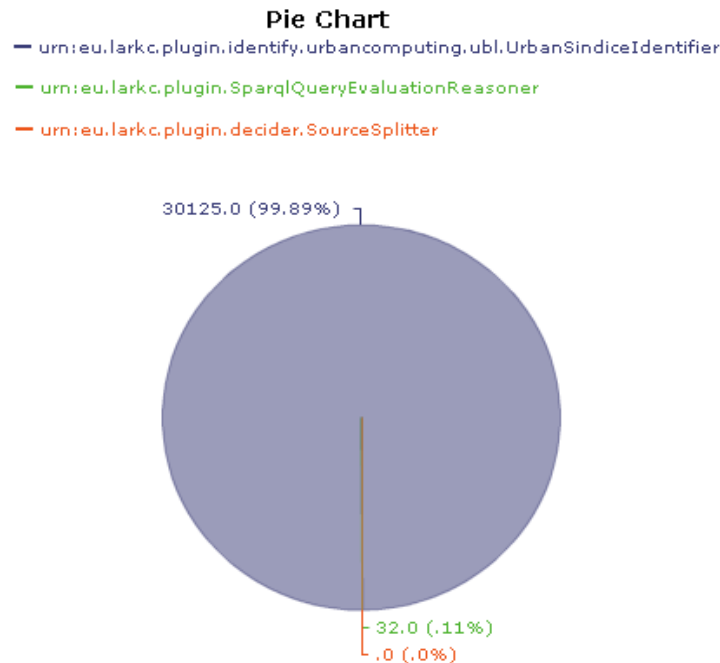


**Figure 12 - TotalCPUTime used by platform**



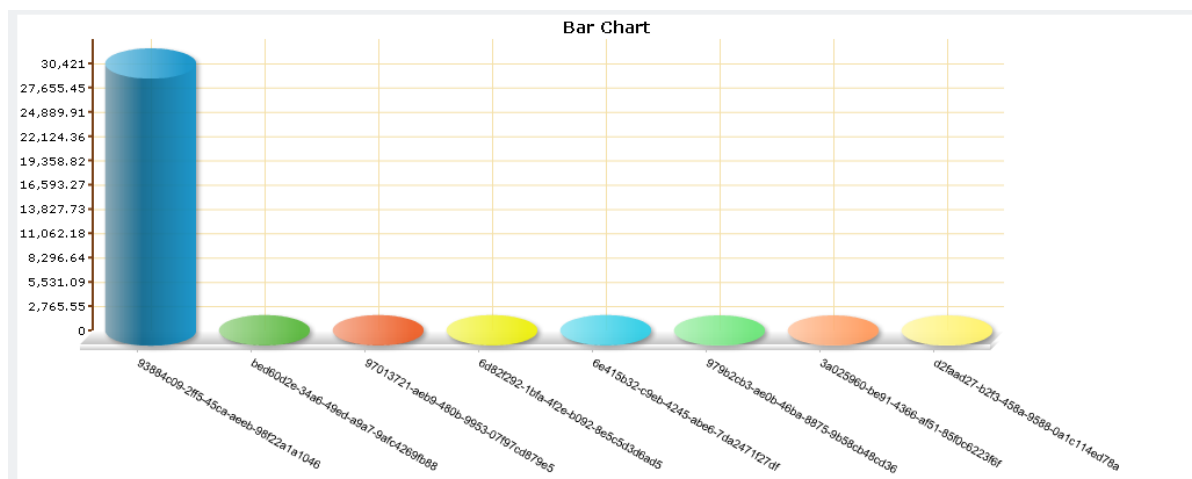
**Figure 13 - Free memory available for the platform**

Figure 14 shows that the total response time of UrbanMonument workflow is highly influenced by the total response time of the UrbanSindiceIdentifier. The figure just gives a comparative overview in terms of response time for all the plugins that are part of the UrbanMonument workflow.



**Figure 14 - Total response for UrbanMonument plugins when running the query with the biggest response time**

Figure 15 shows the total query response time. It shows the total response time for multiple executions of the same query (8 execution instances of the same query). The response time is measured in microseconds. One can notice that the response time of first query is much higher than the response time of the following queries. This can be explained by the fact that the caching mechanism is enabled and then the following executions of the same query are almost instant.



**Figure 15 - Query total response**

### 3.1.6. Considerations

Using WP11 tools we performed evaluation of WP6 workflows, plugins and platform. We recorded a large set of metrics most characterizing the performance of the components.

For UrbanPathFinder workflow all the tests were executed successfully (no errors) and the workflow, plugins and platform performed well during the experiments. Average time execution for UrbanPathFinder workflow was below 1 second.



The UrbanMonument workflow allows execution of only a finite set of predefined queries. Time execution of a query for this particular workflow varies between one second to one minute. Average time execution for UrbanMonument queries is 6 seconds. For some of the queries the response time goes up to 1 minute. The bottlenecks in this case are not the workflow, plugins or the platform but external services invoked by the workflow (DBpedia in this case).

The UrbanEvent workflow supports execution of queries that make reference to cities for which the workflow has information available.

Running the queries for cities for which the system has no information returns no results. The average time for running a query for UrbanEvent is 5.5 seconds.

In all the experiments we observe no bottlenecks in terms of memory consumption, CPU usage, I/O and other metrics, related to workflows, plugins and platform. Overall, the performance of the system is good, with an acceptable usage of resources and response times.

### **3.2. BOTTARI recommendations using SUNS**

#### **3.2.1. Test goals**

One of the core functionalities of BOTTARI is to recommend interesting POIs to a particular user. The SUNS model estimates the probabilities with that this user would like to visit the POIs. SUNS is a latent variable based probabilistic model [6]. Here we will evaluate the quality of recommendations made by SUNS.

#### **3.2.2. Considered workflow**

Given a user, SUNS provides a list of POIs ranked by the probabilities with that the user might be interested in those POIs. The input query described in D6.10 [10]<sup>2</sup> is an example of the queries acceptable by the workflow where the ProbabilisticRDFTransformer plug-in [7] is applied on the LarKC platform v2.5. The interested reader is referred to D6.10 for more details on the workflow.

#### **3.2.3. Methodology**

In the evaluation we applied three baselines:

- random guess (random)
- item-based k-nearest neighbour (knnItem) and
- the most liked items (mostLiked)

The first baseline randomly recommends POIs. This one is very weak. However, its poor performance can prove whether other methods make sense or not. For the knnItem we defined the similarity measure as cosine function, i.e.,  $\text{sim}(i1, i2) = \langle i1, i2 \rangle / (\text{norm}(i1) * \text{norm}(i2))$ , where  $(i1, i2)$  represents any pair of POIs. We set  $k$  to the total number of the users. The baseline mostLiked takes all positive ratings over all time into account. It recommends the POIs most positively talked about to every user. Of course, one can restrict the time frame by using a time window, e.g., the most liked POIs in the last two weeks. It cannot cause loss in the fairness of the comparisons.

To evaluate the quality of recommendations we withheld one positive rating for each user and treated it as a test data point. We trained the models using the remaining ratings and then estimated the values of the test data points. This is a common method to split the data into the training and the test subsets.

We used the normalized discounted cumulative gain (nDCG) (see Appendix B for details) and the accuracy at the top N POIs (acc@topn) to evaluate an estimated ranking of POIs.

#### **3.2.4. Environment**

We implemented the baselines, the SUNS model and the evaluation methods in Matlab. We run the evaluation on a laptop with Windows XP, CPU 2.1 GHz and 3.24 GB RAM.

---

<sup>2</sup> The input query is also at <http://wiki.larkc.eu/LarKCProject/WP6/WorkInProgress/LBSMA/deploy>.

### 3.2.5. Test results

Before analysing the test results we take a look at some insights of the data set. Table 4 shows the numbers of the entities, i.e., users and POIs, and the numbers of their relations, namely positive / negative / neutral ratings. The data set has following characteristics:

- Very sparse: there are only 0.55% non-zeros entries available in the user-POI data matrix.
- Some POIs are not positively or negatively rated, while a lot of users do not have positive or negative ratings. E.g., 58% users do not have positive ratings. Figure 16 shows the distribution of positive ratings over POIs (left) and over users (right).
- Multi-rating problem: a user can rate the same POI for several times and differently. For instance, user  $u$  rated POI  $p$  positively twice, negatively once and neutrally five times. To solve this problem we decided to consider all ratings. We dealt with three kinds of ratings separately and then combined their results. Here we set positive rating(s) to 1, negative rating(s) to -1 and neutral rating(s) to 0.1, omitting the frequency. In the example we have  $\text{positive}(u,p)=1$ ,  $\text{negative}(u,p)=-1$  and  $\text{neutral}(u,p)=0.1$ . Note that we interpret neutral ratings as little positive ones.

|                |                  | #      | users | sparsity [%] |
|----------------|------------------|--------|-------|--------------|
| entity classes | user             | 31369  |       |              |
|                | poi              | 245    |       |              |
| relations      | positive ratings | 19045  | 213   | 0.29%        |
|                | negative ratings | 14404  | 181   | 0.25%        |
|                | neutral ratings  | 75941  | 245   | 0.99%        |
|                | total            | 109390 | 639   | 0.55%        |

Table 4 - The statistics of the data set

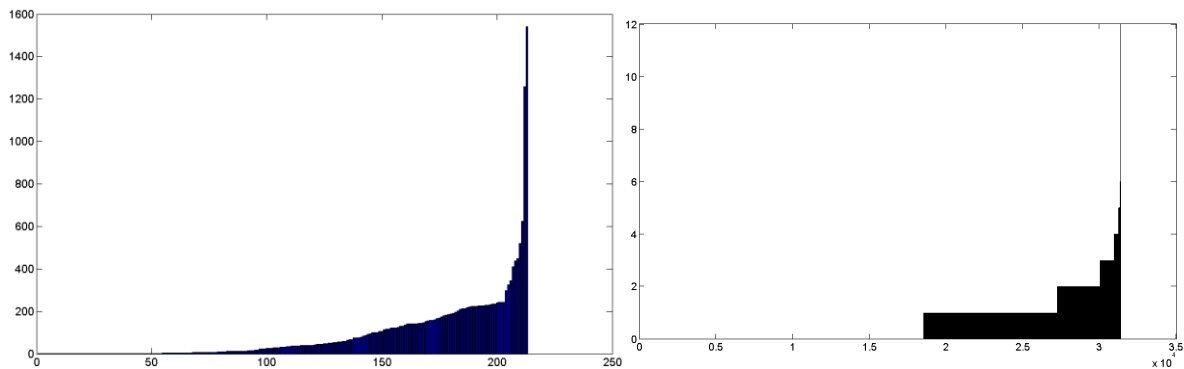
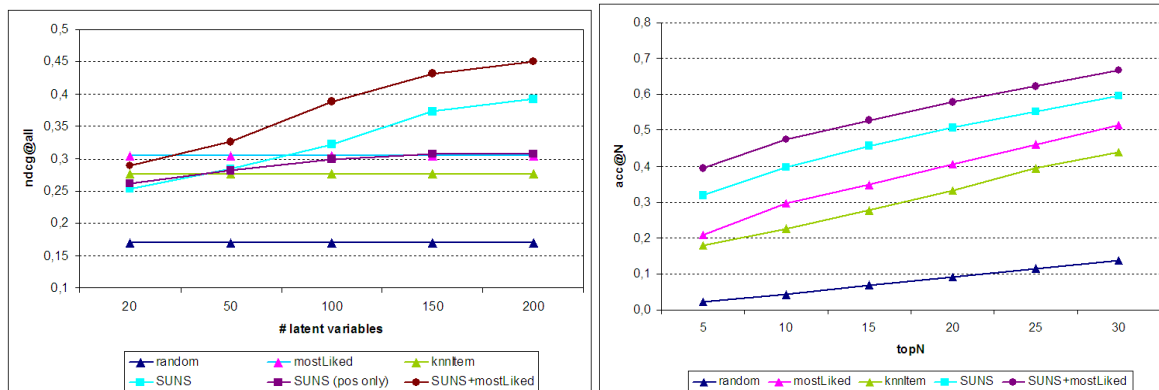


Figure 16: Left: Number of ratings of each POI. Right: Number of ratings of each user.





**Figure 17: Left: nDCG scores. Right: Accuracy values at top N.**

Figure 17 left shows the nDCG scores of the tested methods against the number of the latent variables. Since the baselines are independent of this number, we see three lines each for random, mostLiked and knnItem. As expected, the random is the worst. The mostLiked is lightly better than the similarity-based method. This might indicate the Bandwagon effect that exists in many social communities. SUNS significantly outperformed all baselines after the number of the latent variables reaches 100. Here we also plot the results of SUNS which are only based on the positive ratings and which do not have any improvement compared with mostLiked. The best ranking ever was produced by the combination of both SUNS and mostLiked. These results confirm again the idea presented in the paper [8] – A combined approach of deductive and inductive reasoning.

Figure 17 right shows the accuracy of the top N POIs. We observe similar results: the quality of recommendation made by SUNS is much higher than that provided by the baselines and the combination of SUNS and mostLiked generates the overall best recommendations.

### **3.2.6. Considerations**

The most important issue of the SUNS-based recommendation engine is the scalability. First, the SUNS model guaranties by design the capability of dealing with large data sets. Secondly, in the evaluation the model training cost approximately 86 seconds by #latentVariables=200, while the calculation of the probabilities of POIs for a user needed on average less than 5 milliseconds. Another issue is ease of use. ProbabilisticRDFTransformer plug-in is very easily integrated in a LarKC workflow. In addition, it requires only two model specific parameters.

An interesting future work is to add additional information, e.g., category of POIs and age of users, if available, since this could probably improve the quality of recommendations. Such additional information should be easily integrated by defining appropriate SPARQL queries and it is not necessary to explicitly explore (potentially complex) ontologies.



## 4. Conclusion

In this deliverable we updated the list of data sources (continuing what we started in D6.2) that we analysed in previous months and integrated in our latest Urban Computing scenarios and prototypes. These data sources contain information from Korean social media and details about the points of interest in the Insa-dong area.

On the other side we continued to conduct tests to evaluate the performances of our various Urban Computing workflows. We continued our testing on the very first demonstrator we developed – the so-called Urban LarKC – which was specifically ported to the platform version 2.5 for this evaluation. We also included an evaluation of the latest urban application, BOTTARI, in terms of the recommendations we can provide to the application users; this last test not only assess the behaviour of the newly developed plug-ins and workflows, but also give an idea of the “meaningfulness” of this technical result from the final user point of view.

Our experience in developing Urban Computing applications to address final user needs by using the LarKC platform was overall satisfactory. Our demonstrators should be considered as proofs-of-concept; still, the evaluation campaigns demonstrate that the LarKC platform development proceeded in the right direction and that it is already possible to plan for the use of the platform in real-world applications.



## A. Appendix A – Urban LarkC Workflows Descriptions

### UrbanPathFinder: RDF workflow description

```
1. @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
2. @prefix larkc: <http://larkc.eu/schema#> .
3. @prefix uc: <http://larkc.cefriel.it/ontologies/urbancomputing#> .
4.
5. _:sourceSplitter a <urn:eu.larkc.plugin.decider.SourceSplitter> .
6. _:sourceSplitter larkc:connectsTo _:remoteLoader .
7. _:sourceSplitter larkc:connectsTo _:pathFinder .
8.
9. _:remoteLoader a <urn:eu.larkc.plugin.identify.urbancomputing.ubl.
10. RemoteGraphLoaderIdentifier> .
11. _:remoteLoader larkc:connectsTo _:pathFinder .
12. _:remoteLoader larkc:hasParameter _:remoteLoaderParams.
13.
14. _:remoteLoaderParams uc:location "path_to_ama-xml-
    milano_navigli_graph.rdf";
15. uc:graphName "http://seip.cefriel.it/ama".
16.
17. _:pathFinder a <urn:eu.larkc.plugin.reason.urbancomputing.ubl.
18. OpResPathFinderReasoner> .
19.
20. <urn:eu.larkc.endpoint.sparql.ep1> a <urn:eu.larkc.endpoint.sparql>
    .
21. <urn:eu.larkc.endpoint.sparql.ep1> larkc:links _:path .
22.
23. _:path a larkc:Path .
24. _:path larkc:hasInput _:sourceSplitter .
25. _:path larkc:hasOutput _:pathFinder .
```

### UrbanEvent: RDF workflow description

```
1. @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
2. @prefix larkc: <http://larkc.eu/schema#> .
3. @prefix uc: <http://larkc.cefriel.it/ontologies/urbancomputing#> .
4.
5. _:sourceSplitter a <urn:eu.larkc.plugin.decider.SourceSplitter> .
6. _:sourceSplitter larkc:connectsTo _:cityExtractor .
7. _:sourceSplitter larkc:connectsTo _:sparqlEvaluator .
8.
9. _:cityExtractor a
    <urn:eu.larkc.plugin.transform.urbancomputing.ubl.
10. SparqlToCityQueryTransformer> .
11. _:cityExtractor larkc:connectsTo _:eventIdentifier .
12.
13. _:eventIdentifier a
    <urn:eu.larkc.plugin.identify.urbancomputing.ubl.
14. EventIdentifier> .
15. _:eventIdentifier larkc:connectsTo _:xml2rdf .
16. _:eventIdentifier larkc:hasParameter _:eventIdentifierParams.
17.
18. _:eventIdentifierParams uc:hasApiKey "Md2mzZP7Tk4GkPhw";
19. uc:hasXsltTransformation
20. "http://seip.cefriel.it/urbanlarkc-public/evdb-event2rdf.xsl".
```



```

21.
22. _:xml2rdf a <urn:eu.larkc.plugin.transform.urbancomputing.ubl.
23. XML2RDFTransformer> .
24. :xml2rdf larkc:connectsTo :sparqlEvaluator .
25.
26. _:sparqlEvaluator a
   <urn:eu.larkc.plugin.SparqlQueryEvaluationReasoner>.
27.
28. <urn:eu.larkc.endpoint.sparql.ep1> a <urn:eu.larkc.endpoint.sparql>
   .
29. <urn:eu.larkc.endpoint.sparql.ep1> larkc:links _:path .
30.
31. _:path a larkc:Path .
32. _:path larkc:hasInput _:sourceSplitter .
33. _:path larkc:hasOutput _:sparqlEvaluator .

```

### UrbanMonument: RDF workflow description

```

1. @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
2. @prefix larkc: <http://larkc.eu/schema#> .
3. @prefix uc: <http://larkc.cefriel.it/ontologies/urbancomputing#> .
4.
5. _:sourceSplitter a <urn:eu.larkc.plugin.decider.SourceSplitter> .
6. _:sourceSplitter larkc:connectsTo _:monumentLocator .
7. _:sourceSplitter larkc:connectsTo _:sparqlEvaluator .
8.
9. _:monumentLocator a
   <urn:eu.larkc.plugin.identify.urbancomputing.ubl.
10. UrbanSindiceIdentifier> .
11. _:monumentLocator larkc:connectsTo _:sparqlEvaluator .
12.
13. _:sparqlEvaluator a
   <urn:eu.larkc.plugin.SparqlQueryEvaluationReasoner> .
14.
15. <urn:eu.larkc.endpoint.sparql.ep1> a <urn:eu.larkc.endpoint.sparql>
   .
16. <urn:eu.larkc.endpoint.sparql.ep1> larkc:links _:path .
17.
18. _:path a larkc:Path .
19. _:path larkc:hasInput _:sourceSplitter .
20. _:path larkc:hasOutput _:sparqlEvaluator .

```



## B. Appendix B - NDCG

NDCG is calculated by summing over all the gains in the rank list  $R$  with a log discount factor as

$$nDCG(R) = \frac{1}{Z} \sum_k \frac{2^{r(k)} - 1}{\log(1 + k)},$$

where  $r(k)$  denotes the target label for the  $k$ -th ranked item in  $R$ , and  $r$  is chosen such that a perfect ranking obtains value 1. To focus more on the top-ranked items, we also consider the  $nDCG@n$  which only counts the top  $n$  items in the rank list. These scores are averaged over all ranking lists for comparison.



## 5. References

- [1] Kim et al.: D6.2 “Templates of periodic report on data and performance”, LarKC deliverable, 2008
- [2] Della Valle et al.: D6.4 “1<sup>st</sup> Periodic report on data and performance”, LarKC deliverable, 2009
- [3] Dell’Aglío et al.: D6.6 “2<sup>nd</sup> Periodic report on data and performance”, LarKC deliverable, 2009
- [4] Celino et al.: D6.7 “3<sup>rd</sup> Periodic report on data and performance”, LarKC deliverable, 2010
- [5] <http://www.larkc.eu>
- [6] Volker Tresp, Yi Huang, Markus Bundschuh, and Achim Rettinger. Materializing and querying learned knowledge. In Proceedings of the First ESWC Workshop on Inductive Reasoning and Machine Learning on the Semantic Web, 2009.
- [7] <http://www.larkc.eu/plugin-marketplace/>
- [8] Davide Francesco Barbieri, Daniele Braga, Stefano Ceri, Emanuele Della Valle, Yi Huang, Volker Tresp, Achim Rettinger, Hendrik Wermser: Deductive and Inductive Stream Reasoning for Semantic Social Media Analytics. IEEE Intelligent Systems 25(6): 32-41 (2010)
- [9] Ioan Toma, Raluca Brehar, Silviu Bota, Mihai Negru, Andrei Vatavu, Mihai Chezan. D11.1.2 LarKC metrics ontology, LarKC deliverable, 2011.
- [10] Irene Celino, Daniele Dell’Aglío, Emanuele Della Valle, Seon-Ho Kim, Tony Lee, Yi Huang, Florian Steinke, Volker Tresp, Zhisheng Huang: D6.10 “Urban Computing environment v3”, LarKC deliverable, 2011.