Manuscript Number: JWS-D-09-00080

Title: SemaPlorer---Interactive Semantic Exploration of Data and Media based on a Federated Cloud Infrastructure

Article Type: Special Issue: SemanticWebchallenge 2008

Keywords: Heterogeneous Semantic Data, Real-time Exploration and Visualization, Linked Open Data, Faceted Browsing, Amazon EC2, Amazon S3, Billion Triple Challenge

Corresponding Author: Dr. Ansgar Scherp,

Corresponding Author's Institution:

First Author: Simon Schenk

Order of Authors: Simon Schenk; Carsten Saathoff; Steffen Staab; Ansgar Scherp

Abstract: SemaPlorer is an easy to use application that allows end users to interactively explore and visualize a very large, mixed-quality and semantically heterogeneous distributed semantic data set in real-time. Its purpose is to acquaint oneself about a city, touristic area, or other area of interest. By visualizing the data using a map, media, and different context views, we clearly go beyond simple storage and retrieval of large numbers of triples.

The interaction with the large data set is driven by the user. SemaPlorer leverages different semantic data sources such as DBpedia, GeoNames, WordNet, and personal FOAF files. These make a significant portion of the data provided for the billion triple challenge. It intriguingly connects with a large Flickr data set converted to RDF.

SemaPlorer's storage infrastructure bases on Amazon's Elastic Computing Cloud (EC2) and Simple Storage Service.

We apply NetworkedGraphs as additional layer on top of EC2, performing as a large, federated data infrastructure for semantically heterogeneous data sources from within and outside of the

cloud. Therefore, the application is scalable with respect to the amount of distributed components working together as well as the number of triples managed overall. Hence, SemaPlorer is flexible enough to leverage for exploration almost arbitrary additional data sources that might be added in the future.

SemaPlorer won the first prize at the Billion Triple Challenge of the International Semantic Web Conference in Karlsruhe, 2008.

# SemaPlorer—Interactive Semantic Exploration of Data and Media based on a Federated Cloud Infrastructure

Simon Schenk[a], Carsten Saathoff[a], Steffen Staab[a],
Ansgar Scherp[a,*]

[a]*University of Koblenz-Landau, ISWeb, Germany,*
*http://isweb.uni-koblenz.de*

**Abstract**

SemaPlorer is an easy to use application that allows end users to interactively explore and visualize a very large, mixed-quality and semantically heterogeneous distributed semantic data set in real-time. Its purpose is to acquaint oneself about a city, touristic area, or other area of interest. By visualizing the data using a map, media, and different context views, we clearly go beyond simple storage and retrieval of large numbers of triples. The interaction with the large data set is driven by the user. SemaPlorer leverages different semantic data sources such as DBpedia, GeoNames, WordNet, and personal FOAF files. These make a significant portion of the data provided for the Billion Triple Challenge. It intriguingly connects with a large Flickr data set converted to RDF. SemaPlorer's storage infrastructure bases on Amazon's Elastic Computing Cloud (EC2) and Simple Storage Service. We apply NetworkedGraphs as additional layer on top of EC2, performing as a large, federated data infrastructure for semantically heterogeneous data sources from within and outside of the cloud. Therefore, the application is scalable with respect to the amount of distributed components working together as well as the number of triples managed overall. Hence, SemaPlorer is flexible enough to leverage for exploration of almost arbitrary additional data sources that might be added in the future. SemaPlorer won the first prize at the Billion Triple Challenge of the International Semantic Web Conference in Karlsruhe, 2008.

*Key words:* Heterogeneous Semantic Data, Real-time Exploration and Visualization, Linked Open Data, Faceted Browsing, Amazon EC2, Amazon S3, Billion Triple Challenge

# 1 Introduction

Informing oneself about cities, touristic regions, and other areas of interest is a task often performed on the Internet. Today's applications supporting users in this task are centralized and monolithic such as travel sites like Tripadvisor (`http://www.tripadvisor.com`) and Wikitravel (`http://wikitravel.org`) and knowledge platforms like Freebase (`http://www.freebase.com`). With our novel infrastructure and application, SemaPlorer, we target a web of networked data spaces. Such systems, services, and data stores are easily and seamlessly integrated into a federated infrastructure in order to enable generic access to semantic multimedia data. The different data spaces may be located remotely, provided over SPARQL end points that can be queried and connected over a distributed infrastructure. (Almost) arbitrary data sources may be added ad hoc at any later point in time to extend the data infrastructure of SemaPlorer.

A major step forward towards accomplishing this visionary objective is presented in this paper with the SemaPlorer application and underlying data infrastructure. SemaPlorer is an interactive application that gives end users a usable tool to explore and visualize a very large, mixed-quality and semantically heterogeneous distributed semantic data set in real-time. For Sema-Plorer, we pursue a blended browsing and querying approach [8] to retrieve and visualize information. Users can navigate through almost arbitrary data sets using different facets (cf. [5]) such as location, time, people, and tags. When the user interacts with the application, multiple queries are sent to and executed by the underlying storage infrastructure to retrieve the appropriate results. The results are visualized using a map, media, and different context views representing the different facets.

For SemaPlorer, we have integrated and leveraged different semantic data sources such as DBpedia (`http://dbpedia.org`), GeoNames (`http://geonames.org`), WordNet (`http://wordnet.princeton.edu`), and personal FOAF files contained in the Swoogle (`http://swoogle.umbc.edu`) crawl of Semantic Web data. These make a significant portion of the data provided for the Billion Triple Challenge. Further, we have incorporated a partial crawl of Flickr (`http://flickr.com`) as a very large non-semantic data set that has been converted to 700 million RDF triples. Together, they form a very large, semantically heterogeneous and mixed-quality data set that sums up to more than 1 billion triples. Linking this data set requires a flexible and scalable storage infrastructure. The SemaPlorer infrastructure in its configuration for

———
\* Corresponding author. Tel: +49 261 287-2717

   *Email addresses:* `schenk@uni-koblenz.de` (Simon Schenk), `saathoff@uni-koblenz.de` (Carsten Saathoff), `staab@uni-koblenz.de` (Steffen Staab), `scherp@uni-koblenz.de` (Ansgar Scherp).

the Billion Triple Challenge has consisted of a set of 25 RDF stores[1]. The stores are hosted on virtual machines on Amazon's Elastic Computing Cloud (EC2, `http://aws.amazon.com/ec2/`). Amazon's Simple Storage Service (S3, `http://aws.amazon.com/s3/`) is used to store the EC2 virtual machine images and the semantic datasets. The stores can be transparently accessed as a single, virtual RDF store through a federator. The federator uses NetworkedGraphs [11], a SPARQL-based distributed view mechanism for RDF, and distributed evaluation of SPARQL queries [10,13]. Lightweight inferencing is done using NetworkedGraphs at runtime, e.g., for integrating semantically heterogeneous data. Thus, adding new data sources becomes extremely easy by extending the federator's configuration while being fully transparent to the SemaPlorer application.

## 2 SemaPlorer Application

Collecting information about an area of interest such as a city or touristic region is a task often performed on the Internet. The more complex such queries get, the harder today's search engines and platforms can fulfill these information requests. For example, a person interested in Berlin can easily find information about the city using standard search such as Google. However, finding places where there is some street art in the city of Berlin is almost impossible. Changing this context to another city such as Paris puts an additional challenge to the application that traditional approaches cannot solve. With the SemaPlorer application, we support the users in conducting such complex data exploration tasks. The application uses data federated from different sites using faceted, blended browsing and querying. We have defined four facets of general interest in SemaPlorer, namely location, time, people, and tags. Other facets can be easily configured and added. A facet provides a filtering on a large data set. For example, SemaPlorer can present the sights of a certain city or area using the location facet. Blended browsing and querying means that while users interact with SemaPlorer, different queries are constructed in the background and forwarded to the underlying storage infrastructure and their results are visualized on the screen. This approach allows for a user-driven visualization and interactive experience of the semantic data provided on the Web today. In SemaPlorer, the users initially state a simple text query to the system as depicted in the top left corner of Fig. 1. The result list contains different places, people, and tags matching the query. When the user clicks on the city of Berlin, the SemaPlorer application updates the center part of Fig. 1 showing a map of the city. Concurrently, a query is executed filling the

---

[1] Given the scalability of today's RDF stores, a smaller number would certainly suffice. However, this higher number illustrates the scalability of our approach with regard to federation.
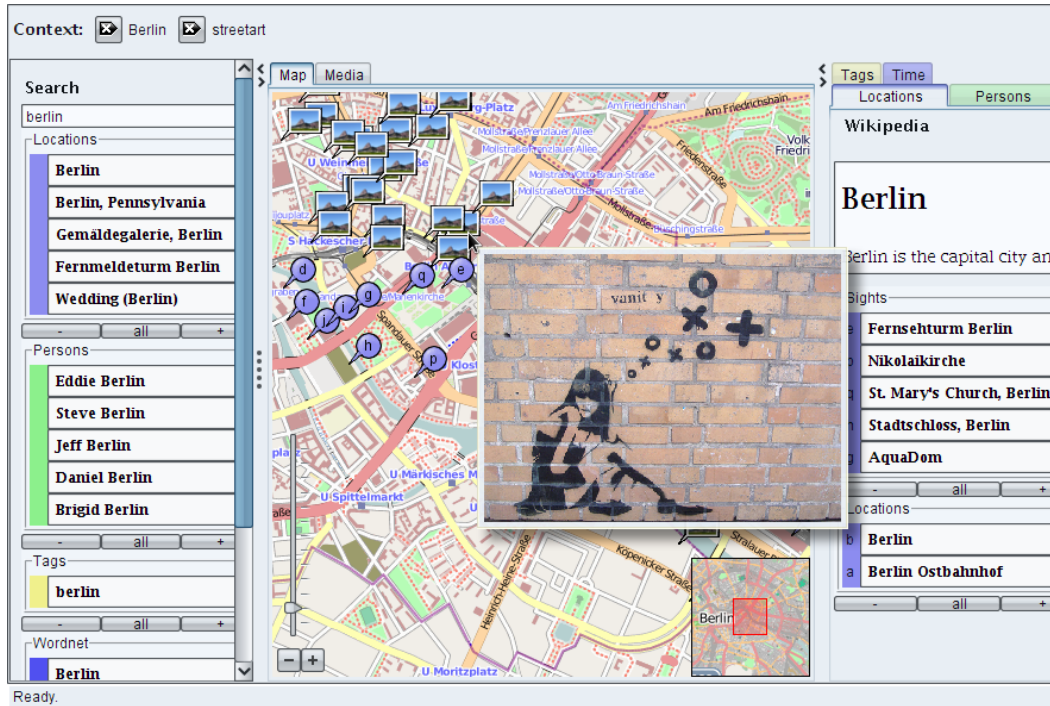
Fig. 1. Screenshot of the SemaPlorer application showing street art in Berlin

map view with interesting places and sights, represented by pins. At the same time, further queries are executed based on what is currently seen on the map to fill the context view in the right hand side of Fig. 1.

For each facet, a context view is defined in the SemaPlorer application. For example, the location view provides information from DBpedia such as population, country, and others. It lists sights and shows nearby places. The people view contains celebrities associated with that place, Flickr users who have uploaded geo-referenced images from that region, and Internet users living in that area according to their FOAF files. The time view allows for selecting a specific time period such as from-to-date and seasons like summer and winter. In the tag view, the tags from Flickr are shown as cloud. All elements in the context views such as sights, nearby places, celebrities, tags, and others are interactive. This means that the users can click on it to continue the blended browsing and querying. For example, when the map view shows the city of Berlin, one can click on the tag *street art*. Instantaneously, the map view is updated and locations of Flickr photos tagged as street art are shown. By stating another query for Paris, the user can switch from the current context of street art photos in Berlin and compare them with Paris.

4

## 3   SemaPlorer Dataset and Interlinking of the Data

To provide blended browsing and querying about areas of interest in Sema-Plorer, different kinds of semantic data are combined. We use a significant portion of the dataset provided for the challenge, namely DBpedia (120M triples), GeoNames (70M triples), WordNet (2M triples), and Swoogle (175M triples). In addition, we use a crawl of Flickr covering several months in 2005-2006 (700M triples), which has been translated to RDF. As described in Sec. 2, we have defined different facets for our SemaPlorer application. These facets are provided by different parts of the data. In the following, we describe the data used for the different facets and how they are connected.

**Location.** Elements of this facet refer to geographic coordinates. We employ GeoNames for cities, countries, and others. Images are displayed based on geo-tagged pictures on Flickr. For sights, we use a combination of full-text search on DBpedia article labels and SKOS category labels. In order to identify sights, we use the SKOS categories that are available in DBpedia. We assume that *skos:broader* is transitive and precompute the transitive closure of all resources. Subsequently, we perform a full text search on the category labels and constrain the results to resources that are connected to *dbpedia:Visitor_attractions* via *skos:subject* and the transitive closure of *skos:broader*. For displaying nearby places and sights, we select all siblings of a chosen location element and rank them based on the geo location distance. For example, when selecting the Arc de Triomphe in Paris, nearby places computed include Eiffel Tower and Notre Dame. This has to be done, because *nearbyPlace* information is missing from the GeoNames export.

**Time.** For the time facet, there is no explicit data set defined. We rather provide the possibility to filter content from a certain time period, e.g., select pictures of a specific month from Flickr. In addition, we allow filtering of content from a particular season like winter and summer.

**Person.** From the datasets introduced above, we have identified three types of persons. First, we select "celebrities" from DBpedia. Second, we select users that posted images on Flickr. Finally, we search for Internet users that published their FOAF files from Swoogle. For any of these types of persons, we use a different combination of the data. For celebrities, we find images depicting the selected celebrity based on a full-text search on the Flickr tags. With respect to a Flickr user, we search for content posted by the user. For Internet users, we look at their FOAF profile's geo location (if available) and connect it with images of that location from Flickr.

**Tags.** Tags are directly associated with the Flickr content. We provide full-text search over the tags. When a tag is selected by a user, we show related tags from Flickr and WordNet.

**Complexity of Queries.** For filling the facets described above, multiple queries are executed at the same time. For the initial search by keyword as described in Sec. 2, three simultaneous queries are performed for retrieving locations, persons, and tags. When clicking on one of the retrieved items in the search results, eight simultaneous queries are executed filling the media view and map view, calculating nearby places, selecting sights, celebrities, Flickr users, Internet users, tags, and retrieving the DBpedia abstract. The same queries are performed when the context of the current view is changed, e.g., when the location is changed by clicking on a sight or nearby place or when a specific person or tag is selected in the corresponding context view. The SPARQL queries make use of the full expressiveness of SPARQL, including UNION, OPTIONAL and various FILTER expressions. Additionally, Lucene queries are included in the SPARQL queries using predicate functions and Sesame LuceneSail (`http://dev.nepomuk.semanticdesktop.org/wiki/LuceneSail`). We have extended the Lucene Sail to allow for range queries and queries for geographic proximity. The queries have a standard length of 4 to 9 joins. On average, 2 to 3 joins connect multiple repositories with up to 4 datasets in a single query. As the GeoNames and Flickr datasets have been distributed over multiple repositories, a varying number of distributed unions are executed. However, these are less critical as they can easily be parallelized. Depending on the context the user selects, the queries can grow, e.g., by selecting images tagged with multiple tags in a certain time period in a certain geographic area.

**Achievements and Experiences.** When designing the dataset for our SemaPlorer application and working on it, we found out that the data sets are often not complete and sometimes the semantics are not explicit enough. For example, GeoNames is missing information on sights and nearby places. Nevertheless, we were able to retrieve this information by intriguingly connecting the heterogeneous data sets as described above. Considering the data set, we further observe that the data is heterogeneous even within a solitary dataset. For example, there is no clear approach for specifying the place of birth of a person in DBpedia. Sometimes it is *dbpedia:cityofbirth* and sometimes *dbpedia:birthPlace*. In SemaPlorer, we solve such ambiguities by mapping the two properties and unifing the result sets. While Linked Open Data makes progress in linking the metadata, it is still an open issue how to exploit it for managing resources such as Flickr images. As SemaPlorer shows, mapping of Linked Open Data and the RDF conversion of the Flickr data is feasible and it works well, e.g., with GeoNames. However, instead of tagging images with keywords and mapping these tags with Linked Open Data, it would be more beneficial to directly use Linked Open Data to annotate the images. For

example, an image depicting the Eiffel Tower could be annotated with the corresponding DBpedia concept.

## 4  SemaPlorer Architecture

The architecture of SemaPlorer is depicted in Fig. 2. It is divided into two subsystems: The first subsystem consists of the K-Space Annotation Tool (KAT, `https://launchpad.net/kat`) and it's SemaPlorer specific extensions, the KAT Plugins. It is deployed to the client's computer and provides the user interface and application logic of the SemaPlorer application described in Sec. 2. The second subsystem implements the federated data infrastructure and comprises an Administration Component for RDF repositories, the NetworkedGraphs-based Federator, and the different RDF Stores for the semantic data and Literal Stores for the DBpedia abstracts and Flickr tags. The Administration Component and the Federator are hosted on our local computing infrastructure. All other components, i.e., RDF Stores and Literal Stores providing the billion triple data set are hosted on Amazon EC2 nodes. The architecture of SemaPlorer and the single components are described in more detail in the following.

The first subsystem, provided by KAT and its plugins is a generic architecture designed to develop applications for browsing and (semi-automatically) annotating multimedia data. It can be extended by generic functionality such as an interactive map or access to Flickr images. The functionality is provided via a Messaging Bus to more application specific plugins such as the depicted SemaPlorer plugin. KAT provides a Plugin Manager for managing application specific extensions. Furthermore, it provides some GUI Tools and a GUI Layouter. Finally, KAT possesses a local storage infrastructure for multimedia annotations based on COMM [1] and Sesame 2 (`http://openrdf.org`). This storage is designed for annotations made by (semi-automatic) annotation plugins or manual annotations by the users. It will become an interesting feature for future extensions of our SemaPlorer application.

The data set described in Sec. 3 is provided through the second subsystem, the NetworkedGraphs-based federated data infrastructure leveraging Amazon's EC2. The Administration Component of this data infrastructure controls the virtual machines running on EC2. Using a simple web GUI, EC2 nodes for specific parts of the data or the entire dataset can be started and stopped. New datasets can be created by adding a description of the dataset to a configuration file and starting the new node. Whenever nodes are started or stopped, the Administration Component updates the Federator configuration accordingly. The Federator is the single SPARQL endpoint offering SemaPlorer unified access to the whole dataset in a virtual RDF repository.
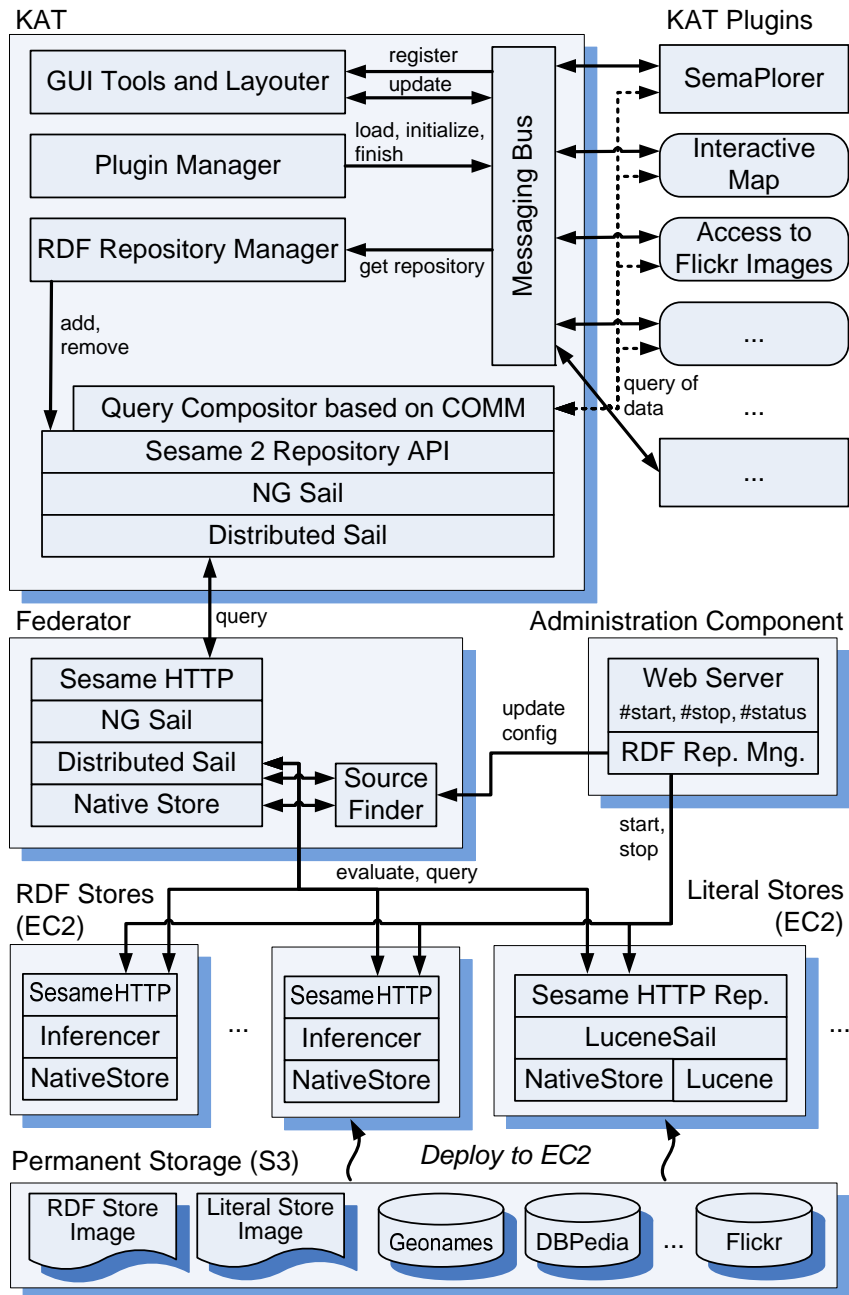
Fig. 2. Architecture of SemaPlorer

Queries against the Federator are analyzed to determine, which endpoints can be used to evaluate parts of the query. Subsequently, the query is split into subqueries that are evaluated at the actual data sources [10,13].

The dataset is stored at storage nodes in EC2 using S3. We use three different configurations for EC2 nodes: The first one stores RDF data without any inferences. It is used, e.g., for DBpedia infobox data. It also serves as basis for the other two node types. The second one uses LuceneSail and additionally provides full-text indexes over the RDF literals. It is used, e.g., for tags, DBpedia articles, and category labels. For the SemaPlorer application, we do not need full RDFS inferencing. In contrast, transitivity in SKOS hierarchies is needed, which is not provided by RDFS. Hence, we use inferencing with custom rules in the third configuration of S3 nodes. As the custom rules inferencer of Sesame does not scale to the dataset used, we precompute the transitive closure of *skos:broader* for DBpedia categories.

In addition to SPARQL federation, the Federator performs simple schema mappings to homogenize representations from the various data sources used for SemaPlorer. This schema mapping is done at run time using Networked-Graphs. For example, for persons we have three different representations: FOAF files using the FOAF vocabulary, DBpedia using a (Living)Person category, and Flickr users. Similar challenges arise from the modeling of geographic entities and annotation of images and for providing access to properties without a clear schema such as place of birth in DBpedia. In order to allow the SemaPlorer application to abstract from these differing representations, we map them to a canonical form. In the case of Persons, the FOAF vocabulary is used. As a result, we can add any dataset for which a mapping to the FOAF vocabulary is possible.

## 5 Related Work

The principle idea of faceted, blended browsing and querying is intriguing, but well-known, e.g., [12,7]. The winner of the Semantic Web challenge 2006, /facet [6], has brought this idea into the arena of semantic data. Recently, the faceted application Freebase Parallax (`http://mqlx.com/~david/parallax`) emerged, a faceted browser for exploring and visualizing the structured data of Freebase (`http://www.freebase.com`). The largest disadvantage of /facet and Freebase Parallax is that they are built on a centralized infrastructure that does not allow for scalable use of a large set of data coming from many different data sources. With the SemaPlorer application based on KAT and NetworkedGraphs, we have achieved this and provide for a faceted, blended browsing and querying over a very large, mixed-quality and semantically heterogeneous distributed semantic data.

9

Various systems providing highly scalable management of RDF data have been provided, e.g. YARS2 [4]. These systems aim at managing a large volume of RDF data in a single, albeit federated, repository. In contrast, our infrastructure aims at integrating multiple semantically heterogeneous repositories across the Semantic Web into a single virtual repository infrastructure. DARQ [9] is a related approach aiming at querying multiple SPARQL endpoints. In contrast to our system, it is based on manually maintained statistics about remote endpoints, which we do not assume to be available. Additionally, severe limitations are imposed on the structure of queries by DARQ. In the context of the Linked Open Data effort, challenges similar to our setting arise with respect to storage requirements. However, querying is not addressed. DynaQuest [3] aims at a web-scale distributed virtual relational database. However, relational databases do not cope well with semi-structured, semantically heterogeneous data.

## 6  Evaluation

Goal of the Billion Triple Challenge was to demonstrate the scalability of Semantic Web technologies to up to more than a billion triples and doing something useful with these triples. As such, the SemaPlorer application and its underlying infrastructure has been designed as a technical demonstrator rather than an application running in a real productive environment. In order to gain insight of the usability and usefulness of an application in such an early stage and to get feedback on how to improve it, a formative evaluation [2] of the application has been conducted. To evaluate SemaPlorer, we asked 20 test persons (11 PhD students and 9 graduate students) from the computer science department of the University of Koblenz to use our application. The test persons are between 21 to 36 years old and have good or very good knowledge using a computer. 18 test persons already use map-based applications for information exploration and visualization like Google Maps (`http://maps.google.com`) and others. The test persons use these applications for trip planning (75%), gathering information about a location (55%), and for business search (25%). Thus, the test persons are typical users that our SemaPlorer application aims at and are good candidates to provide relevant feedback to it.

The evaluation of SemaPlorer has been conducted in three phases, namely introduction, test, and feedback. In the introduction phase, the participants were familiarized with the SemaPlorer application and its features. The participants were told that the evaluation is not about tracking and measuring their performance but gaining feedback how to improve SemaPlorer. In the subsequent test phase, the actual evaluation is carried out. Each test person executes a predetermined set of tasks. Having such a common course of tasks

is important to ensure comparability between the single test persons and receiving valid feedback. In the feedback phase, the participants have filled in a questionnaire and were able to express further subjective feedback.

For the test phase, the test persons could take as much time as they need and like to conduct the tasks defined. The duration of the evaluation sessions were between 10 to 60 minutes (average 30, median 25). Thus, the test persons have spent a reasonable amount of time using SemaPlorer. The tasks conducted were searching for the city of Berlin and looking for sights in Berlin using the "sights" feature. Then, the result set should be narrowed down to show pictures of street art only by adding the "streetart" tag and exploring street art pictures around the transmission tower around Berlin using the "nearby places" feature. A special kind of street art are space invaders found by adding the "space invaders" tag. The test candidates were asked to explore space invaders in Berlin. Subsequently, the location context should be changed to Paris to explore the space invaders there. To further explore Paris, the test persons were asked to search for specific Flickr users and looking for interesting pictures the users took. In addition, the test persons should search for Celebrities in Paris from DBpedia and navigate along the entities with semantic relation to Paris found in WordNet.

In the last phase, the test persons where asked to fill out a questionaire to provide feedback on the features currently implemented in SemaPlorer and the application in general. Table 1 summarizes the questions asked and their ratings. The questions could be answered in a range from 1 to 5, where 1 means totally agree, 2 agree, 3 so-so, 4 disagree, and 5 totally disagree. The feedback shows that the search results very much meet the test persons expectations (S1). Also the separation of the search results into location, tags, and persons was welcomed (S2). However, changing the context using the search feature could be designed in a more intuitive way (S3). The usability of the map view and media view is considered good or better (V1 and V2). Considering the single features of the facets, all features were valued good or average such as selection of sights in the location facet (F1). Also interesting sights were found (F2). The nearby places feature was rated as good (F3). However, the quality of the nearby places determined (see Section 3) was only valued so-so (F4) and thus should be improved. Navigation along WordNet (F5) and selecting Celebrities from DBpedia (F6 and F7) are both also valued so-so. Here, we think that especially the feature of navigating along WordNet is questionable and might be removed. Only the feature of browsing along Flickr users was disliked by the test persons. Apparently only very few or no interesting users or photos of celebrities were found (F8 and F9).

In the last phase of our evaluation, the test persons could make suggestions of features they like to be added to SemaPlorer. These are a history feature to allow for scrolling backward/forward, selecting multiple locations to plan a

Table 1

Feedback on the search feature (S1-S3), map view and media view (V1-V2), as well as facets (F1-F10), and performance of the application (P1). For each question, the mean and standard deviation (StDev) is shown.

| Question | Mean | StDev |
|---|---|---|
| S1: Search results meet my expectations. | 1.7 | 0.9 |
| S2: Separation into location, tags, and persons is intuitive. | 2.2 | 0.7 |
| S3: Change of context using the search feature is intuitive. | 3.2 | 1.0 |
| V1: Map view is intuitive and easy to use. | 2.0 | 0.6 |
| V2: Media view is a good addition to the map view. | 1.8 | 0.8 |
| F1: Is the selection of sights a useful feature? | 1.6 | 0.5 |
| F2: Did you find any interesting sights? | 2.2 | 0.7 |
| F3: Is the nearby places feature useful? | 1.9 | 0.6 |
| F4: Did you find interesting nearby places? | 2.8 | 0.9 |
| F5: Is the navigation using WordNet useful? | 2.9 | 1.0 |
| F6: Did you find interesting celebrities in DBpedia. | 2.6 | 1.0 |
| F7: Is this feature useful? | 2.6 | 1.0 |
| F8: Is this Flickr users feature useful? | 3.3 | 1.0 |
| F9: Did you find interesting Flickr users? | 4.1 | 0.8 |
| P1: The response time meets my expectations. | 2.5 | 1.2 |

trip, and presenting a slideshow of pictures. One test person stated that there are already too many features. We also asked the test persons which additional information sources we should add to SemaPlorer. Here, among others the integration of satellite images, further media types like videos, news, other sights like subway stations, cafes, and cinemas, as well as meta-information about sights like opening hours were requested. Very interesting was the feedback asking for support to judge the trustworthiness of the information provided. Finally, the test persons could provide further subjective feedback if they liked. Here, 5 out of the 20 participants mentioned that they would like to see further improvement on the performance of the application. Although response times are in general good, some complex queries take longer than the test persons like to accept. In the questionnaire, the response-time of SemaPlorer is judged between good and so-so (P16). This rating might be surprising but we assume that the test persons had commercial systems like Google Maps in mind when rating the performance of SemaPlorer. Thus, it is important to note that SemaPlorer is not an application running in a real productive environment like Google Maps but a technical demonstrator designed to prove the scalability of Semantic Web technologies. Also some usability issues were

mentioned. For example, the change of location via the facets menu could be more intuitive.

## 7 Conclusions

In this paper, we have presented the SemaPlorer application and data infrastructure. As shown, the SemaPlorer application is an easy to use tool that allows end users to interactively explore and visualize a very large, mixed-quality distributed semantic data set in real-time. The interaction with the large data set is driven by the user and carried out by a faceted, blended browsing and querying. The application leverages a significant portion of the data provided for the Billion Triple Challenge, namely GeoNames, DBpedia, WordNet, and Swoogle. Further, a large Flickr data set converted to RDF is incorporated. However, the main focus of the SemaPlorer application remains on the use and integration of the different data sources provided for the challenge. The storage infrastructure underlying SemaPlorer allows for transparent access to arbitrary, distributed RDF repositories, in our case stored on EC2. By this, the application is scalable with respect to the amount of distributed components working together. In addition, arbitrary additional data can be added at a later point in time. Thus, using Amazon's EC2 and Networked-Graphs brings us closer to the vision of generic access to distributed semantic multimedia data. Particularly, we have shown that besides scaling centralized repositories, connecting many smaller repositories is a feasible and in many ways a more advantageous approach to scale with regard to organizational needs of autonomous contributors on the Semantic Web.

In the long term, the preferred mode of operation will be the direct use of SPARQL endpoints run by the providers of the data. Switching to these live data sources can be easily conducted by changing the Federator's configuration and without modifying the SemaPlorer application or any other application that might use the federated data infrastructure. Thus, to save costs, we shut down the part of the SemaPlorer infrastructure that was provided by Amazon's EC2 and S3 after the Billion Triple Challenge ended. This part hosted among others the entire Flickr data set. Instead, we have developed a live SPARQL endpoint that translates SPARQL queries to calls against the Flickr API (`http://www.flickr.com/services/api/`). Hence, instead of hosting the Flickr dataset ourselves on payed EC2 nodes, we now access the live Flickr system from the web. As expected, replacing the corresponding SPARQL endpoint only required a small reconfiguration of the Federator, which was done at runtime and completely transparent to the SemaPlorer application. This again demonstrates the flexibility of our overall infrastructure. Once Swoogle is available through a SPARQL endpoint and DBpedia and GeoNames are

available through SPARQL endpoints that support fulltext search and efficient geo-range queries, we could also switch over to these live data sources.

# References

[1]  R. Arndt, R. Troncy, S. Staab, L. Hardman, M. Vacura, COMM: Designing a Well-Founded Multimedia Ontology for the Web, in: ISWC, 2007.

[2]  G. Gediga, K.-C. Hamborg, Evaluation of software systems, Encyclopedia of Computer Science and Technology 45.

[3]  M. Grawunder, F. Köster, The DynaQuest-Framework for Dynamic and Adaptive Source Selection, in: Collaborative Technologies and Systems, 2003.

[4]  A. Harth, J. Umbrich, A. Hogan, S. Decker, YARS2: A Federated Repository for Querying Graph Structured Data from the Web, in: ISWC, Springer, 2007. URL `http://iswc2007.semanticweb.org/papers/211.pdf`

[5]  M. A. Hearst, Design recommendations for hierarchical faceted search interfaces, in: SIGIR, Workshop on Faceted Search, 2006.

[6]  M. Hildebrand, J. van Ossenbruggen, L. Hardman, /facet: A Browser for Heterogeneous Semantic Web Repositories, in: ISWC, 2006.

[7]  m. c. schraefel, D. A. Smith, A. Owens, et al., The evolving mspace platform: leveraging the semantic web on the trail of the memex, in: Hypertext, 2005.

[8]  K. D. Munroe, B. Ludscher, Y. Papakonstantinou, Blending Browsing and Querying of XML in a Lazy Mediator System, in: Extending Database Technology, 2000.

[9]  B. Quilitz, U. Leser, Querying Distributed RDF Data Sources with SPARQL, in: ESWC, 2008.

[10] S. Schenk, J. Petrak, Sesame RDF Repository Extensions for Remote Querying, in: ZNALOSTI Conf., 2008.

[11] S. Schenk, S. Staab, NetworkedGraphs: a declarative mechanism for SPARQL rules, SPARQL views and RDF data integration on the web, in: WWW, 2008. URL `http://dblp.uni-trier.de/db/conf/www/www2008.html#SchenkS08`

[12] K.-P. Yee, K. Swearingen, K. Li, M. Hearst, Faceted metadata for image search and browsing, in: Human factors in computing systems, ACM, 2003.

[13] J. Zemanek, S. Schenk, V. Svatek, Optimizing SPARQL Queries over Disparate RDF Data Sources through Distributed Semi-Joins, in: ISWC 2008 Poster and Demo Session Proceedings, CEUR-WS, 2008.

**Figure 1**

# KAT

**GUI Tools and Layouter** — register ← / update ↔

**Plugin Manager** — load, initialize, finish →

**RDF Repository Manager** ← get repository

add, remove ↓

**Messaging Bus**

Query Compositor based on COMM ← query of data
Sesame 2 Repository API
NG Sail
Distributed Sail

# KAT Plugins

**SemaPlorer**

**Interactive Map**

**Access to Flickr Images**

...

...

...

# Federator

Sesame HTTP
NG Sail
Distributed Sail
Native Store

**Source Finder**

query ↕

# Administration Component

**Web Server**
#start, #stop, #status

**RDF Rep. Mng.**

update config

start, stop

evaluate, query

# RDF Stores (EC2)

SesameHTTP
Inferencer
NativeStore

...

SesameHTTP
Inferencer
NativeStore

# Literal Stores (EC2)

Sesame HTTP Rep.
LuceneSail
NativeStore | Lucene

...

*Deploy to EC2*

# Permanent Storage (S3)

RDF Store Image

Literal Store Image

Geonames

DBPedia

...

Flickr