We Compare

Project Report

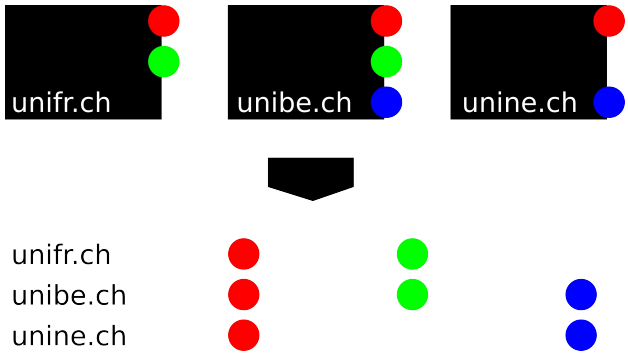
Michael Luggen
Jörg Schenker
Mariusz Wisniewski

University of Fribourg, December 2011

Idea

WeCompare is a system allowing to make comparisons between different things on the web.

When people usually want to compare three different restaurants; they go on the web, find the website of each of the restaurants and write on a piece of paper the different attributes they are interested in. Once they have the attributes they are interested in for each restaurant, they make a comparison between all three of them and make their choice.



This goes for anything, not just restaurants. Travels, cars, clothing, apartments or computers could be another example.

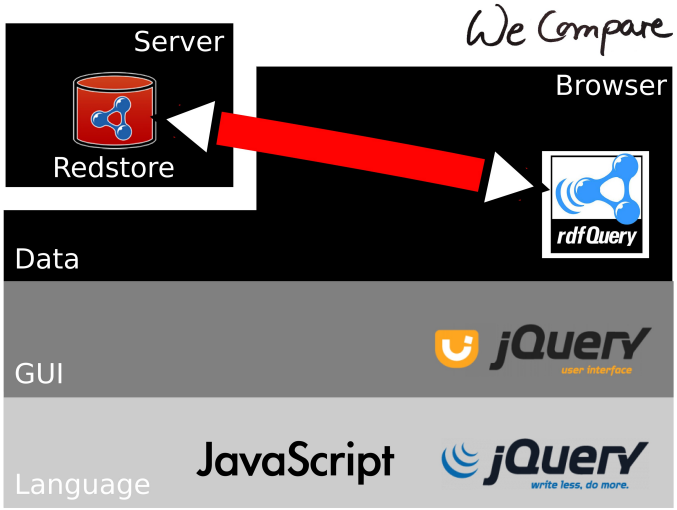
With WeCompare we wanted to speed up this process by allowing the user to have some sort of interface allowing while surfing on websites, to take the information that is interesting and to put it in some sort of list, so that a user can retrieve it easily when he is ready to make a comparison. We also wanted to display the information in a way that helps the user make the comparison; and the main goal is that all of this should be easier and faster using our system than doing it the old way with paper and pencil.

Another interesting thing we wanted to do with this project, is to take the information chosen by the user and turn it into RDF triples that we would store on a database, so that it could be reused by our or by other systems.

Implementation

Our system is implemented as a client side application. Only the data storing is handled by the server.

We have decided that our system will be accessible by users as a webpage allowing them to surf on the internet while having a menu on the right side for the information retrieval. A user can drag and drop any information he finds useful into tags, for which he can choose a title, and once he is done with a page, he can save the information and go on another website. Once he feels he has enough material for a comparison, he can just click a button which will open a window with all the information regarding each website displayed in a way that will help him make the comparison.



All the information chosen by the user will also be turned into RDF triples and stored on our server for later use.

Since our system is a client side application, we decided to use Javascript as the language in which we coded the behavior of our system. The JQuery library also seemed to be a good way to improve the possibilities and to do the graphical part of our system.

For the database, we chose to use Redstore, which is a lightweight triplestore allowing us to store RDF triples and make queries on the stored data.

To communicate between our system and the database, we use rdfQuery, which is an easy-to-use Javascript library for RDF-related processing. We can use it to parse RDF embedded in our page, query over the facts it contains and work in concert with our server-side triplestore.

How does our system work?

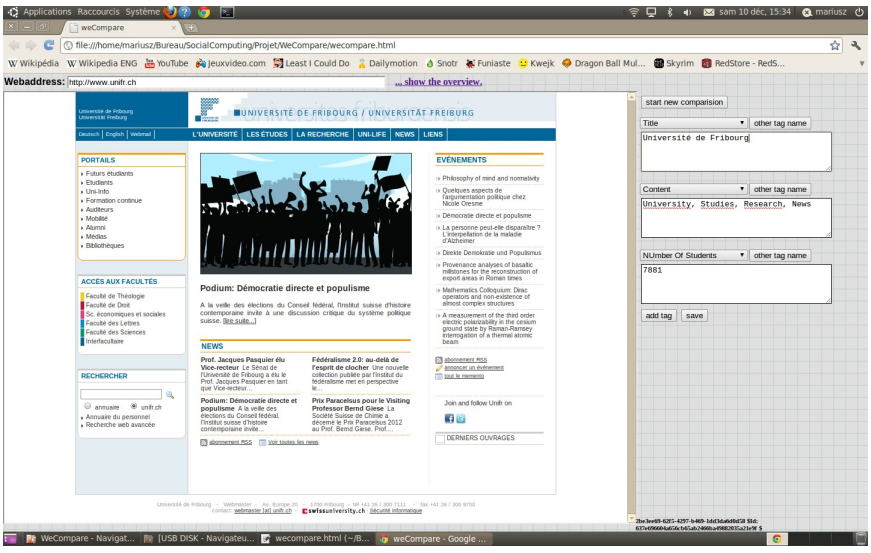
Basically, a user has to connect on our webpage:

http://arkania.dyndns.org/WeCompare/

Once he reaches the home page, he has to enter the first website he wants to visit for his comparison. When this is done, another page is loaded.

On the new page, the user has a browser on the left which is currently browsing the page he asked, and a textarea above the browser in case he wants to enter the address of another webpage.

On the right he has the possibility to create a new tag by clicking a button. The total number of tags that can be opened has been fixed to ten. Each tag when opened is empty. The user can choose a title for its tag in a list containing all the titles possible for the given webpage. He can also create his own title by clicking on a button.



The user should drag and drop all the information he needs from the website into the tags. Once this is done, he just has to click on a button which will open a new window with the list for the comparison or another which will open a new website and store the previous information. All the data put into the tags is transformed into RDF triples and stored on the database for further use.

Problems encountered

We had some problems coding in Javascript, since none of us ever really used it for a serious project before. We had to spend some time before getting comfortable coding in it and using the JQuery library.

One of the biggest problems we encountered was the connection between the database and the client sided application. Indeed, connecting and querying the database from Javascript was really problematic, and rdfQuery was not as useful as we expected it to be. RdfQuery helped us to create databanks to store the information for each webpage in regard of future comparison; it was however useless to connect to the query the database and we had to find another way using Ajax to do so.

Improvements

Our system offers basic functionalities; we could however improve many things.

The first thing is, except for the titles possible for each tag, we do not really reuse the data we stored on the triplestore. We thought about how to reuse the data another user already selected for a given webpage, but couldn't really agree between us on how this should be done. This data however should be reused in a way improving the quality of the service each time a user uses our system.

We could also improve the comparison window by allowing the user to select different options to rearrange data or to change the display of the result.

What we learned

We all got to learn how to code using Javascript, which is a good experience. We also learned how to manipulate and query RDF data on a database.

We now have gained some experience with each element we had to use in this project such as JQuery and Redstore.

One of the main things we learned is that it is not easy to create a system that will be pleasant and useful for people to use; and that even if the concept seems simple; it really isn't that simple to put it in practice.