

Exploring the Netherlands on a “Semantic Path”

Michael Martin

Business Information Systems, University of Leipzig
o-mega@gmx.de
Postfach 100920
04009 Leipzig

Abstract: This poster gives an overview about the web application “vakantieland.nl”, a Dutch Internet portal in the tourism domain. The core functionality is to provide information about holiday destinations, accommodation and other tourism related points-of-interest as well as a corresponding visualization with a mapping service. Because of the underlying semantic data structures and alternatively generated RFD output, vakantieland.nl can be considered to be a Semantic Web application. Its realization and functionalities, social aspects and furthermore an outlook about future development work constitute the main part of the poster.

1 Introduction

Semantic Web [BHL01] and Web 2.0 [Or05] are two concepts that expanded and redefined the possibilities of the Internet. The Semantic Web expands the World Wide Web with ability to represent information in a machine readable way by formally defining the Semantics of the content. Web applications that use such technologies are called Semantic Web applications. To further integrate the user into the web application's processes, an extension of functionalities, which come from the ideas of Web 2.0, is necessary. These Web 2.0 functionalities serve to expand and improve the existing knowledge base and allows many different views on a resource.

Many web applications that were created in an evolutionary process can not be adapted with little effort to new techniques, technologies and use cases. The only way in most cases to guarantee an up-to-date appearance is to completely redesign the application. Vakantieland.nl was one of those cases. The amount of added data that needed to be displayed and the huge collection of functionalities that had to be developed, formed the decision to completely redesign the application. While in the second section the original design of the application is presented, section three then describes the new version and the changes made. In section four we describe the included social aspects based on Web 2.0 and finally the fifth section gives an outlook about functionalities that still need to be developed.

2. The original application

The previous version of vakantieland.nl displayed tourism destinations (e.g. destinations and information points in cities), which are called “Point of Interest” (“POI”). Every POI was associated with at least one category and could be viewed either by selection of the category or through other relations, it had, to other POIs. The output of POIs were in the form of lists and points on maps. The maps, where those points were displayed upon, were saved as images in different zoom steps. Then the tourism destinations where positioned on the maps with the help of their coordinates. For every POI information like name, contact addresses, opening hours and pictures was provided and could be viewed in detail next to the map as can be seen in Figure 1. Additionally a map of the area showed a list of close-by POIs. These associations served as a further mean of navigation between POIs.



Figure 1: GUI of the previous version of the vakantieland.nl portal

The data of this application was kept in a relational data base and were retrieved with ASP¹ to display them. Furthermore filtered information was extracted with VisualBasic and saved in XML². This extracted data was used in external pages.

3. The new version of the vakantieland.nl portal

The company of the portal required the usage of forward-looking technologies to provide a better overview, improved usability and more diverse information on the one hand and to allow the employees an easier handling and maintenance of the site on the other hand. The user also was supposed to be integrated and should have the ability to interactively

¹ASP : Active Server Pages, <http://asp.net/>

²XML: eXtensible Markup Language

create and expand the knowledge base. All data should be made shareable in a machine-readable way, so it could be processed by other applications and sites. To fulfill all the above mentioned requirements the use of Semantic Web technologies with exchangeable data formats and standardized namespaces and APIs, which allow the processing thereof, was necessary.

The scheme and the data of the model of vakantieland.nl was created and is kept in RDF [Be04]. RDFS³ or OWL⁴ enables us to define every conceptual part of the data model (like classes and properties) and the related instances as resources with the corresponding Resource Identifier (URI). Displayed categories are modelled in a class tree. This serves to allow sub classification of categories. For example the category *Hotel* contains the subcategories *ApartmentHotel*, *EconomyHotel*, *CastleHotel*, etc. In the model, these subcategories are subclasses of the class *Hotel*, as can be seen in Figure 2.



Figure 2: Excerpt of the model

To assign POIs to different categories polymorphism is necessary using this way of modelling. To give an example, there could be a special hotel which also has a public restaurant, which belongs to another category. To make the model accessible by the application, it was saved in a database-backed TripletStore [MD00]. For this the Powl-

³RDFS : <http://www.w3.org/RDF/>
⁴OWL : <http://www.w3.org/2004/OWL/>

API⁵ [Au05] was used, which also handled the necessary transformation. To verify if the resulting model and the used SPARQL⁶-queries [Be06] were reusable, we tested it with Ontowiki [ADR06].

The structure of the application, which is implemented in the script language PHP⁷ follows a well known architectural pattern called MVC⁸ [Bu98]. The model component abstracts the accesses by the controller on the Powl-API, which in this case also manages the extraction of data from the TripletStore to make the data suitable for the requirements. The RAP⁹-API [OB04] which is used by the Powl-API allows to query the model with SPARQL [Be06]. SPARQL was used to identify and find resources with specified characteristics in an efficient manner. After the identification of the resource URI, an object with all the existing information is created with the Powl-API. Such generated objects will be requested by the appropriate controller and assigned to the view component. The view component produce the output with the help of templates.



Figure 3: GUI of the new vakantieland.nl portal

The design of the templates was chosen to appear similar to the previous version, but the information and the namespaces are anchored in the HTML¹⁰-code to make it machine-readable and allow processing by other applications. Thus, the title and the description of a POI for example are marked Dublin Core¹¹ properties. The POIs are positioned and

⁵Powl-API: <http://powl.sourceforge.net/>

⁶SPARQL: recursive Acronym for „SPARQL Protocol and RDF Query Language“

⁷PHP : recursive backronym for „PHP: Hypertext Preprocessor“ <http://www.php.net>

⁸MVC: Model View Control

⁹RAP: RDF API for PHP

¹⁰HTML: Hyper Text Markup Language

¹¹Dublin Core : <http://dublincore.org/>

displayed on the map by using the GoogleMaps-API¹² as you can see in Figure 3. The API is used to retrieve the coordinates of the POIs and also to display the map material in conjunction with the related map-markers.

4. Social Semantic Web in vakantieland.nl

Another requirement for the new version vakantieland.nl is to achieve a higher degree of user interactivity. In the previous version of vakantieland.nl users only had the possibility to get informations about POIs. One way to improve the user interactivity is the addition of more functionalities to collect user generated contents. Thus users get the possibilities to share there experiences now.

Therefore new functions were added like the evaluation of POIs and feedback abilities to grade the up-to-dateness of the data and the user now has the possibility to write textual comments like in a guest book. Through this, other interested users now have the advantage to profit from those shared experiences and personal opinions. Furthermore users can now make statements about the actuality and validity of data, which supports the administrators of the application and minimizes the effort to keep the data up-to-date. A well defined, closed user group (registered users with the role 'author') now has the ability to change data like the description of a POI or the opening hours directly.

In this manner interested people that want to explore the netherlands can do it now on an easy semantic path.

5. Outlook

Finally, we give a short description about planned, but not yet implemented functionalities to further improve the diversity of available information and the ability to interact with the application. Users might wish more information than is provided by the knowledge base of the vakantieland.nl project. It is possible, that they, e.g. in search of a museum in Amsterdam, might not be only interested in opening hours, but also want to know more about the history of the museum or even the history of the city, the museum is situated in. To provide for this demand queries are send to a SPARQL-Endpoint of the DBpedia¹³ portal to get more background information about the city, the POI itself or other relevant issues. The results can be integrated in already displayed information. Finally, web services are planned, which supply data in different formats like JSON¹⁴ and RDF to requesting clients.

¹²GoogleMaps-API: <http://www.google.com/apis/maps/>

¹³Dbpedia : <http://dbpedia.org>

¹⁴JSON: JavaScript Object Notation

References

- [Au05] Sören Auer : Powl – A Web Based Platform for Collaborative Semantic Web Development, Scripting for Semantic Web, Vol-135, 2005
- [ADR06] Sören Auer, Sebastian Dietzold, and Thomas Riechert. Ontowiki - A tool for social, semantic collaboration. In Isabel F. Cruz, Stefan Decker, Dean Allemang, Chris Preist, Daniel Schwabe, Peter Mika, Michael Uschold, and Lora Aroyo, editors, The Semantic Web - ISWC 2006, 5th International Semantic Web Conference, ISWC 2006, Athens, GA, USA, November 5-9, 2006, Proceedings, volume 4273, pages 736-749. Springer, 2006.
- [Be04] Dave Beckett : RDF/XML Syntax Specification (Revised). W3C. 10February 2004.
- [Be06] Dave Beckett : SPARQL RDF Query Language Reference V1.8 2006 <http://www.dajobe.org/2005/04-sparql/SPARQLreference-1.8.pdf>
- [BHL01] Tim Berners-Lee, James Hendler, Ora Lassila: The Semantic Web. Scientific American 284 (2001) 35–43
- [Bu98] F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad, M. Stal. Pattern orientierte Softwarearchitektur, Addison Wesley 1998
- [MD00] Sergey Melnik, Stefan Decker: A Layered Approach to Information Modeling and Interoperability on the Web 2000 <http://infolab.stanford.edu/~melnik/pub/sw00/>
- [OB04] Radoslaw Oldakowski, Christian Bizer - RAP: RDF API for PHP, 1st International Workshop on Interpreted Languages (colocated with the NetObjectDays 2004 Conference), Erfurt, September 2004
- [Or05] Tim O'Reilly – What is Web 2.0, 2005, <http://www.oreilly.com/>