Developing Web of Data Applications from the Browser

Pavel Arapov, Michel Buffa, Amel Ben Othmane

▶ **To cite this version:**

HAL Id: hal-01083320
https://hal.inria.fr/hal-01083320

Submitted on 18 Nov 2014

Developing Web of Data Applications from the Browser

Pavel Arapov
WIMMICS research team
I3S Laboratory
Sophia Antipolis, France
arapov@i3s.unice.fr

Michel Buffa
WIMMICS research team
I3S Laboratory
Sophia Antipolis, France
buffa@i3s.unice.fr

Amel Ben Othmane
WIMMICS research team
INRIA
Sophia Antipolis, France
amel.ben-othmane@inria.fr

ABSTRACT

WikiNEXT is a wiki engine 100% written in JavaScript that relies on recent APIs and frameworks. It has been designed to author web applications directly in a web browser, which can exploit the web of data. It combines the functionalities of a semantic wiki with those of a Web-based IDE (Integrated Development Environment) in order to develop web applications in addition to writing classic documents. It gives developers a rich internal API (Application Programming Interface) and provides several functionalities to exploit the web of data. Our approach uses templates, a special type of wiki pages that represent the semantic data model. Templates generate wiki pages with semantic annotations that are stored as quadruplets in a triple store engine. To query this semantic data, we provide a SPARQL endpoint. Screencasts are available on YouTube (look for WikiNEXT).

Categories and Subject Descriptors

K.4.3 [**Organizational Impacts**]: Computer-supported collaborative work.

Keywords

Semantic Web, Web2.0, Wikis, Semantic Wikis, Knowledge Management, Web Applications

1. INTRODUCTION

The wiki concept was introduced in 1995 by Ward Cunningham. Today many projects such as Wikipedia's MediaWiki[1] engine are based on it. MediaWiki is an example of a wiki engine to do the publication of articles. It is extensible by plugins and extensions which must be installed server side. Later, wikis became very popular as enterprise knowledge management systems; this has led to another, more complex class of wiki engines. In these wiki engines such as Confluence[2], Mindtouch[3], TWiki[6], XWiki[5] or SharePoint[4], users can integrate widgets or macros into the wiki.

At the same time, semantic wikis, such as Semantic MediaWiki[1], MoKi[4], OntoWiki[3] and IkeWiki[2] have enabled wikis to be enhanced with semantic information. Some of these wiki engines are only research experiments. Semantic wikis tend to use one of two approaches, using the wiki either as a data model or to support structured data modeling.

---

[1] http://www.mediawiki.org/

[2] https://confluence.atlassian.com/

[3] http://www.mindtouch.com/

[4] http://office.microsoft.com/en-us/sharepoint/

Recently, it has become possible to use IDEs in a web browser. Some lightweight online IDEs are well known by web developers, including Cloud9[5], Nitrous.IO, jsfiddle.com, and jsbin.com. Cloud9 and Nitrous.IO are examples of Web-based IDEs that support a range of product development features. They support version control for source code, collaborative development for multiple users, as well as debug and run mode for a project. Meanwhile, jsfiddle.com and jsbin.com are examples of code sharing services. In these services, developers can both write HTML/CSS/JavaScript code and share it. They are designed to share a piece of code or a single page web application without a server-side part.

This paper presents WikiNEXT[6], which combines the features of a semantic wiki and a web based IDE, to support developers of web applications that exploit the web of data. In section 2 we will explain how the spread of Linked data has led to increasingly more advanced web applications that use different underlying semantic models. Section 3 introduces WikiNEXT as part of a new trend of using a wiki as a development tool and shows the difference with classic wikis.

2. WEB APPLICATION AND WIKIS

Although Semantic Web technologies have been around for a while, they have had little impact on the development of Web applications. With significant quantities of Linked Data being published on the Web, numerous efforts are underway to build applications that exploit this Web of Data. This has created a challenging environment for exploiting and modeling the knowledge pipeline.

In the Conventional Model, a Web application is composed of three clearly-separated major components, namely the Web browser, the Web server and the backend storage system (e.g., a database). Other models have extended the Conventional Model with extra client-side control of data or computation. For example, Wikipedia articles are collaboratively maintained by users and complex wiki templates are frequently used to enable advanced page layout. A user may also call extensions of a wiki platform (e.g., "parser functions" in MediaWiki) to perform certain computation tasks such as string processing, mathematical computation and visualization. The MediaWiki engine and its successors such as MindTouch largely use this approach. Also note that a wiki page may embed script languages (e.g. Velocity and Groovy in XWiki[7]) for advanced tasks, which allow the API to run in-page applications or macros.

3. WIKI AS A DEVELOPMENT TOOL

Wikis, web-based collaborative authoring tools, have been used in other areas, such as for course management or specific activities in the software development process, such as code reviewing, bug

---

[5] https://c9.io/

[6] http://wikinext.gexsoft.com/

[7] http://platform.xwiki.org/xwiki/bin/view/DevGuide/Scripting

tracking, and functional testing. However, wikis, or wiki-like tools have not been widely used for programming purposes.

Our application, WikiNEXT[8], a wiki/web-based IDE, is used for developing web applications that exploit the Linked Data (LD). It provides two editors: a HTML editor and a JavaScript editor that can be used in development mode for advanced users. Development mode offers the opportunity to create powerful and dynamic applications through the web browser. Wiki pages, in this environment, are considered as web applications. The life cycle of wiki pages that embed application code is considerably more complicated than that of classic wiki pages. The lifecycle of a classic wiki page is shown in Figure 1.
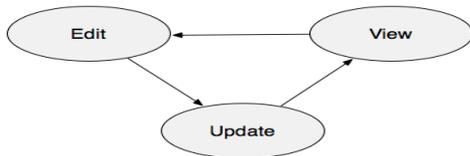

Figure 1: The lifecycle of a classic wiki page

This is different from the lifecycle of program development in a web-based IDE (see Figure 2). Since a wiki page *is an application* in WikiNEXT, developing an application and editing a wiki page are the same process.
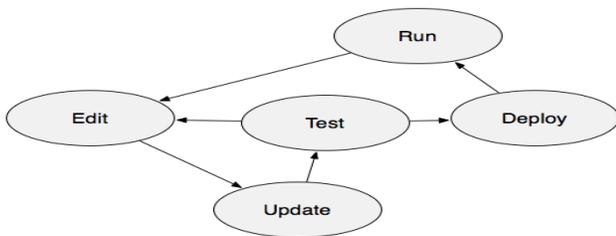

Figure 2: Life cycle of a program development

Therefore, the "deploy" and "run" stages are implicit steps that occur when a page is saved and rendered. We think that this environment is well suited for retrieving, combining and reusing multiple sources available in the web of data.

The main problem for development in wikis is how to store the data. Existing wiki engines require access to the wiki database or an external data store. We propose using semantic annotations integrated into wiki pages for storing application data. In the next section we introduce the WikiNEXT Data Model and describe how it can be used to model structured data and generate semantic metadata.

4. WikiNEXT Data Model

Semantic wikis enrich conventional wikis by allowing users to edit and query structured semantic annotations, not just plain wiki text. This enables a novel, transparent, and light-weight social Web application model, which lets developers collectively build Web applications. Thus, semantic wikis can be used for data modeling, data management, data processing and data presentation. We were inspired from this trend to develop our model.

Unlike other semantic wikis (e.g. Semantic MediaWiki) that use extensions to support some data processing abilities, WikiNext uses JavaScript as a scripting language to create and query about structured annotations. A special type of wiki page called a

---

[8] https://github.com/pavel-arapov/wikinext

"template" page also allows variables to be used as input parameters. Template wiki pages can be reused later in the same wiki page for displaying or processing data. Templates can also be used for modeling structured data and generating semantic data. By aggregating the data management and data processing features with JavaScript, we can design interactive, visualized interfaces to manipulate semantically enriched data. In a WikiNEXT application, many elements can be constructed using the HTML editor. Utilizing templates permits the data to be presented with various visual elements (e.g., tables, graphs and trees).
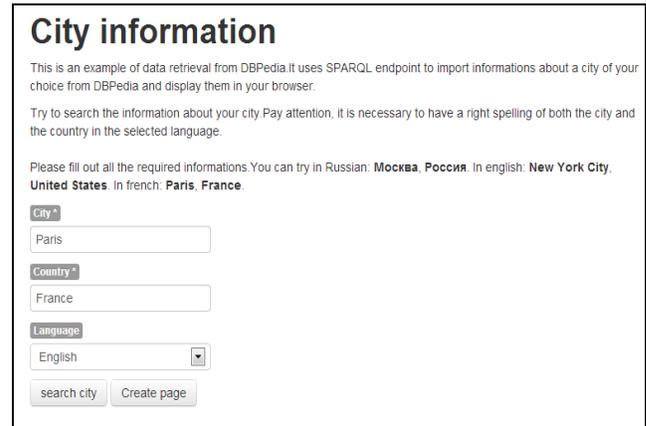

Figure 3: City application UI

The metadata generated by applications embedded in WikiNEXT pages is also represented in the page as RDFa/RDFa lite[9] annotations. For example, a list of countries resulting from a request or a web services call to DBPedia.org, can be reused in the page. WikiNEXT parses these annotations and saves them in a modified version of RDFStore-js, which supports SPARQL 1.1. The traditional page content is stored as a JSON objects in a MongoDB[10] database.

To understand how templates work for both data visualization and semantic data generation, we now describe a WikiNEXT-based Web application that exemplifies the general methodology described above.

Several WikiNEXT-based Web applications have advanced User Interfaces; for example, the City application[11] for city searching and annotation is shown in Figure 3. In this application, in order to create structured data about cities we retrieve data from DBPedia.fr and we create wiki pages with Web of data - compliant data. Below is a simplified example of a template used in the City application.

```
<span about={{_id}} typeof="City">
<h1 property="name">{{city}}</h1>
</span>
```

In the code above we define an object with an identifier "{{_id}}" which has RDF type "City", from the vocabulary http://schema.org/City. This vocabulary states that a City has a

---

[9] http://www.w3.org/TR/rdfa-lite/

[10] http://mongodb.org

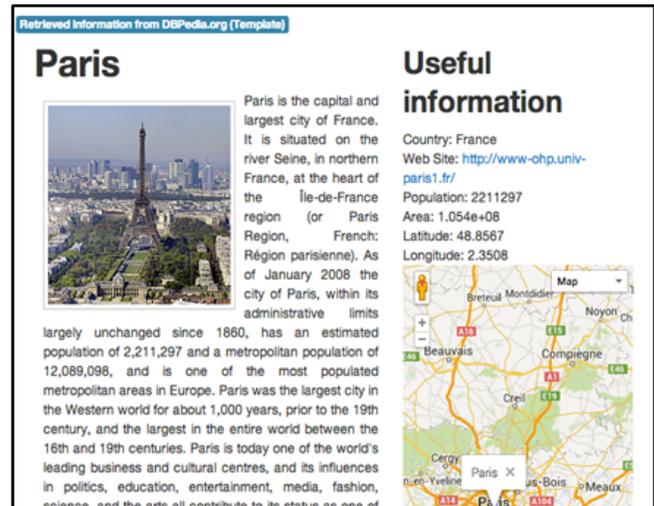[11] http://wikinext.gexsoft.com/wiki/519e04c580194c4178000001

Figure 4: Difference between default rendering (on the left) and template-based rendering (on the right)

property "name", and we will give it the value "{{city}}". WikiNEXT uses the Mustache[12] template engine on the client side, and this syntax comes from the template engine.

The JavaScript code below fills the template with real values:

```
var pagedata = {};
pagedata.city = city; // dbpedia result
article = wikinextHelper.city(pagedata);
```

Figure 4 illustrates the difference between the default rendering and template-based rendering of a query result. The template-based result page on Figure 4 on the right is augmented with semantic data and can be queried by other web applications. The semantic metadata of this page is shown in Figure 5. This data is accessible through the SPARQL endpoint for the wiki applications or external use.
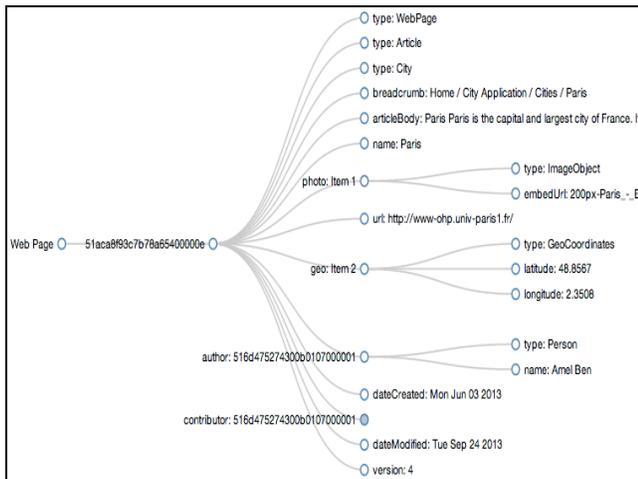

Figure 5: Semantic graph of a created city page

5. CONCLUSION

WikiNEXT allows users to create web applications that use linked data via simple access through a web browser. The demo version is available online. We also conducted several usability tests. The protocol we used, as well as detailed results of this evaluation are available in WikiNEXT pages online, and show that WikiNEXT saves time, compared to traditional programming using a Dekstop IDE, for writing applications that query the web of data.

6. REFERENCES

[1] KRÖTZSCH, Markus, VRANDEČIĆ, Denny, and VÖLKEL, Max. Semantic MediaWiki. In: The Semantic Web-ISWC 2006. Springer Berlin Heidelberg, 2006. p. 935-942.

[2] Schaffert, S. (2006, June). IkeWiki: A semantic wiki for collaborative knowledge management. In Enabling Technologies: Infrastructure for Collaborative Enterprises, 2006. WETICE'06. 15th IEEE International Workshops on (pp. 388-396). IEEE.

[3] Auer, S., Dietzold, S., & Riechert, T. (2006). OntoWiki–A tool for social, semantic collaboration. In The Semantic Web-ISWC 2006 (pp. 736-749). Springer Berlin Heidelberg.

[4] Marco Rospocher, Chiara Ghidini, Viktoria Pammer, Luciano Serafini, Stefanie N. Lindstaedt: MoKi: the Modelling wiKi. SemWiki 2009

[5] Canals, G., Molli, P., Maire, J., Laurière, S., Pacitti, E., & Tlili, M. (2008). XWiki Concerto: A P2P wiki system supporting disconnected work. In Cooperative Design, Visualization, and Engineering (pp. 98-106). Springer Berlin Heidelberg.

[6] CHU, Samuel Kai-Wah. TWiki for knowledge building and management. Online Information Review, 2008, vol. 32, no 6, p. 745-758.

---

[12] http://mustache.github.io/