Combining and Integrating Advanced IT-Concepts with Semantic Web Technology Mashups Architecture Case Study

Amin Anjomshoaa, A Min Tjoa, and Andreas Hubmer

Institute of Software Technology and Interactive Systems, Vienna University of Technology
{anjomshoaa,amin}@ifs.tuwien.ac.at, andreas.hubmer@tuwien.ac.at

Abstract. Even though Semantic Web technologies have flourished consistently in the past few years, it is unlikely to achieve the Semantic Web goals on the global Web in near future. The initial expectations such as turning the World Wide Web to a machine-comprehendible medium are far away from realization. The best proof of this is a look at the current status of World Wide Web and small percentage of websites and services that are Semantically-enabled. The main reason for this situation is that so far it is not easy to get people to learn and apply Semantic Web concepts for their Web content and use them efficiently in their daily life. In this context advanced IT concepts such as Mashups can support Semantic Web goals and at the same time Semantic Web technologies can also improve computer-to-computer and human-to-computer interactions in enterprise systems. In this paper the two-way support of advanced IT and Semantic Web is explored and a novel approach for advancing the power of web forms is presented.

Keywords: Semantic Web, Mashups, Web 2.0, Software Architecture.

1 Introduction

The initial vision of Semantic Web has been "turning the World Wide Web to an environment in which information is given well-defined meaning, better enabling computers and people to work in cooperation" [1]. Semantic Web technology has been widely accepted and used to capture and document context information in many domains. It plays a significant role in information sharing scenarios and interoperability across applications and organizations [2]. This added-value opens the way to integrate huge amount of data and becomes extremely useful when used by many applications that comprehend this information and bring them into play without human interaction.

Unfortunately the content description methods are not being used by all content owners and the current Web's Achilles' heel in our belief is the lack of semantic information that can be used to link this huge amount of information efficiently. This brings many web specialists to expecting another web called Web 3.0 to complete the deficiencies of the current web.

N.T. Nguyen, M.T. Le, and J. Świątek (Eds.): ACIIDS 2010, Part I, LNAI 5990, pp. 13–22, 2010.

The basic reason for this situation is that so far it is not easy to get people to learn and apply Semantic Web concepts for their Web content and use them efficiently in their daily life. The real breakthrough in Semantic Web implementation occurs by the emergence of semantic-enabled content authoring and management tools that makes the paradigm shift from Traditional Web to Semantic Web feasible. The shift away from the traditional web is forced by the growing need for more efficient information sharing, collaboration and business processes [3].

Meanwhile Web 2.0 has set a new trend in the Rich Internet Application (RIA) world. It makes better use of the client machine's processing power and at the same time pushes the SOA paradigm to its limits. Web 2.0 has also introduced new possibilities for a better human computer interaction via rich applications such as Mashups that provide a *user-driven micro-integration of web-accessible data* [4]. At the moment mashups are mainly used for less important tasks such as customized queries and map-based visualizations; however they have the potential to be used for more fundamental, complex and sophisticated tasks in combination with Semantic Web concepts.

In this paper, emerging IT-concepts such as Web 2.0 and Mashups will be discussed and a novel approach for smart integration of global web information into business processes will be introduced. The proposed solution demonstrates how such IT-concepts can benefit from Semantic Web technologies to facilitate the information sharing and smarter inter-process data exchange. We will also show how Semantic Web can utilize the Web 2.0 and Mashups to accelerate realization of its goals and making the global web machine-interpretable.

2 State of the Art

There are several methods for applying the semantic concepts to current web applications and making its content machine readable. One such solution is to embed semantic information into the web content at creation time so that machines can read and interpret the content without the overhead of natural language processing methods. In this context the RDFa initiative of W3C [5], provides a set of HTML attributes to augment visual data with machine-readable hints. It is highly beneficial to express the structure of web data in context; as users often want to transfer structured data from one application to another, sometimes to or from a non-web-based application, where the user experience can be enhanced. There are many major use cases where embedding structured data in HTML using RDFa provides a significant benefit. For example people's contact information, events and content's licenses such as creative commons can be included in web contents using RDFa syntax and relevant namespaces.

The RDFa is not the only solution for providing more intelligent data on the web. A similar approach for embedding machine-readable data in web content is delivered by microformats [6] which are supposed to be coinciding with the design principles of "reduce, reuse, and recycle". The main difference between these two approaches can be explained by its historical background. The microformats have grown out of the work of the blog developer community as an easy and ad-hoc response to common applications,whereas RDFa is built with a more systematic vision of the W3 Semantic Web group and its associated thinkers.

In parallel to Semantic Web concepts and technologies the World Wide Web evolution has resulted in new topics such as Web 2.0 that has hit the mainstream, and continues its rapid evolution. Successful Web 2.0 applications are characterized by crowd-sourcing and social networking where people create and share the content. The Web 2.0 has set a new trend in Rich Internet Application (RIA) world. It makes a better use of the client machine's processing power and at the same time pushes the SOA paradigm to its limits. In this context, Internet will play the role of a global operating system that hosts the Web Services. In other words, the Web 2.0 fosters the global cloud computing idea where business services are presented on the Internet and developers should select and weave them together to create new compound services.

In this context SOA and Semantic Web Services are supposed to provide and manage the required services in a smart way, but despite all advantages of SOA and Semantic Web Services (SWS), they are complex and cost-intensive technologies. Moreover the target user group of SOA and SWS are IT-professionals who have a thorough understanding of these technologies and their underlying standards. In other words, there are not suitable for end-users who lack these capabilities.

Another shortcoming of SOA-based approaches is their inability to react rapidly to the changes in the business environment, as implementations of such systems are cost and time intensive and any changes in the environment may require some modifications in the system.

An important characteristic of systems for end-users is their capability for personalization and customization. Most of today's business processes and SOA-based solutions are still mainly designed to satisfy the mass of users. Unfortunately customization and mass generic production are at odd with each other. As a matter of fact IT solutions typically focus on 20% of user requirements that have an impact on most users and the long tail of requirement is usually ignored by IT providers. It is too difficult for an ordinary user to take benefit of the available services by composing the appropriate services together. This is the reason why SOA-based approaches are mainly covered in the hands of IT-departments of big firms who have a complex stack of technologies to realize SOA-based scenarios. Per contra end users require simple, cost-effective techniques which enable them to design solutions in an ad hoc and "quick and dirty" manner.

The Web 2.0's approach for creating such "quick and dirty" solutions is given by Mashups that introduces new possibilities for a better human-computer interaction. The term mashup originates from music industry where a song or composition is created by blending two or more songs, usually by overlaying the vocal track of one song seamlessly over the music track of another [7]. Mashups owes its popularity and fast improvements to its two basic blocks namely Web 2.0 and SOA. Mashup envisions building effective and light-weight information processing solutions based on the exposed Web Services of organizations. Such Web Services may range from simple Web Services such as RSS [8] and REST-based [9] services to complex BPEL services for more serious use cases; however the Mashups benefits in the latter use cases is not yet known to IT decision makers.

The power of mashups is also being examined in real world information management scenarios and has attracted many attentions [11]. Mashups can be applied to a broad spectrum of use cases ranging from simple data widgets to more complex use cases such as task automation and system integration. The mashup applications are roughly categorized under the following groups:

- Consumer Mashups (Presentation Mashups): are the simplest group of Mashups used to facilitate the creation of information portals from different resources for presentation purpose. This group of mashups has the lowest degree of customization and is usually implemented as pre-built widgets that can be added to user interfaces.
- Data Mashups: which are used to integrate data and services from different resources such as Web Services, RESTful APIs, Web Extractors, RSS Feeds, etc. These kind of mashups aim to facilitate the data access and cross-referencing between resources.
- Enterprise Mashups (Logic Mashups): always involve programming and therefore are the most complex mashup category. They connect two or more applications, automate certain tasks and include awareness of workflow [12]. Enterprise mashups usually depend on some server-side components and compete with data integration and service orchestration technologies such as BPEL and Enterprise Service Buses (ESBs).

According to market research reports, this situation is going to change quickly in the coming years. Mashups are identified among the top 10 strategic technologies for 2009 [10] and it is expected that many serious business scenarios will be delivered through enterprise mashups. The power of mashups is also being examined in real world information management scenarios and has attracted many recent efforts [13].

3 Semantic Web and Mashups

The bidirectional support of Semantic Web and Mashups boost each other and has the potential to provide a solid basis for many interesting applications. The Mashup support for Semantic Web has come alertness via ad-hoc mashups which on the one hand enables the connection to preconfigured information resources and the processing of the data of these resources and on the other hand maps its context data to the relevant domain ontology. In other words, instead of embedding the semantic meaning to the web content, semantics is attached to relevant content via mashups in a dynamic and loosely coupled manner. Today, there are a handful of projects such as Google Sidewiki [14], Dapper [15], and Lixto [16] that follow this approach to extract and enrich web page information; however in most of the cases the process remains at a simple data extraction and text annotation level and semantic aspects are not completely covered. In the case of Google Sidewiki which is a recently released browser sidebar, users are enabled to contribute and read information alongside any web page. Google Sidewiki does not support semantic concepts at the moment and just aims to use the power of the crowd to enrich metadata of web pages. The interesting part of this approach is the free annotation of page content which may overlap each other. In other words part of the page content can be annotated differently by different end-users based on their target applications and use cases. Hence the quality of user annotations will be evaluated by the crowd and better annotations appear on top of the list.

In the near future, the Mashup support for Semantic Web will play an important role in realizing the Semantic Web goals and transforming the current web to an information space that can be interpreted and used by machines for advanced use cases.

The other facet of the relationship between Mashups and Semantic Web is the support that Semantic Web and ontologies can provide for Mashups in facilitating the creation of Mashups for novice users. This application of Semantic Web has its roots in Semantic Web Services that are aiming to automate service discovery and composition without human intervention. The basic difference between Semantic Web Services and Semantic Mashups approaches is derived from their different target users. The Semantic Web Services are mainly managed and used by IT experts who are aware of underlying data structures and corresponding services, whereas the Semantic Mashups target group are novice users who need to combine the Mashup Widgets for their specific purposes [11]. This issue is especially important for the creation of Enterprise Mashups which always involves programming and includes awareness of workflow. Enterprise mashups usually depend on some server-side components and compete with data integration and service orchestration technologies such as BPEL and Enterprise Service Buses (ESBs). In Semantic Mashup context, these server-side components are described by an appropriate domain ontology that eases the composition of mashup widgets and creation of new mashups.

One of the major drawbacks of Mashup based solutions is the fact that such solutions are fragile and not as stable as formal business process. To provide a solid basis for more serious business application, the gap between mashups and well established BPEL solutions should be closed. In the next section a novel approach for bridging this gap will be introduced which enables the end user to benefit from simple service composition of Mashups and at the same time to manage and control the process by stable business process engines.

4 Mashup to BPEL Conversion

As explained before Mashups are very helpful in creating fast solutions for data integration. To provide a solid basis for more serious business application, the gap between mashups and well established BPEL solutions should be closed. A closer look into the main mashup categories (presentation, data, and enterprise mashups) reveals that the functionality of mashups is to some extent similar to SOA and its service composition; however the overlap with SOA is much more blurred for logic (enterprise) mashups than for the other two categories. Client-side mashups compete with server-side orchestration technologies such as BPEL (Business Process Execution Language), but that has not hindered large enterprise SOA vendors as IBM, Microsoft, BEA, Sun and Oracle from embracing enterprise mashups. The reason here is that, like BPEL, mashups depend on standardized Web services and APIs, all of which SOA is aimed at making them available. This does not mean that mashups depend on SOA. In a field survey, nearly two thirds of enterprises that use browser-based mashups have done so without SOA, and one of the biggest advantages of mashup technology is the low barrier to entry, with many applications using free services on the Internet [17].

As the number of serious enterprise mashups grows and they get used for more serious tasks, there is a growing need to make mashups more stable and robust for complex business scenarios. As a matter of fact, in the context of SOA and Mashup solutions, stability and ease are at odd with each other (see Figure 1). In a mashup-based approach the services can be easily composed to create a situational solution; however the result is not that stable. On the other hand a SOA-based approach provides a stable basis for creating and running the business processes for expert users who are familiar with complex stack of SOA, but a novice user will not be able to create solutions based on available services.

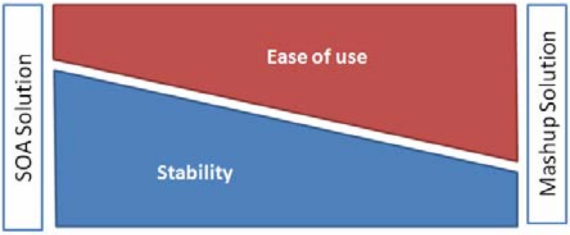

Fig. 1. SOA Solution vs. Mashup Solution

To change this situation, a novel approach is proposed that benefits from the advantages of both SOA and Mashup solutions to facilitate the creation of situational solutions that are also stable and as a result, business processes can rely on them.

The core idea of the proposed approach is to break the design-time and runtime processes between Mashup and SOA environments respectively. In other words the solutions will be first created in a mashup environment and benefits from all advantages of mashup solutions. As the next step the Mashup will be translated to a formal business process using a Mashup-BPEL convertor. Finally this means that any stable BPEL engine can run the process and take care of process management issues. As a result the end user will benefit from simple service composition of mashups and at the same time the process will be managed and controlled by well established business process engines. *In this way, the process of running a mashup is safely transferred from the browser on the client computer to a BPEL engine on server-side.* This approach might not be that helpful for presentation and data mashups, but will be of great advantage for enterprise mashups. It is also important to note that, the proposed approach is not trying to reinvent the wheel and duplicate the functionality of BPEL editors. Unlike a typical BPEL editor, the proposed approach hides the complexities of a business process such as defining the partner links, services, etc and saves the end-users from technical details and let them solely concentrate on the their use cases. To demonstrate the feasibility of this approach a small use case will be presented that shows how a simple BPEL process can be generated from its corresponding mashup. This use case is based on the basic math services that can be combined to do more complex calculations. Figure 2 depicts a simple Mashups that uses math services and delivers the result to display.

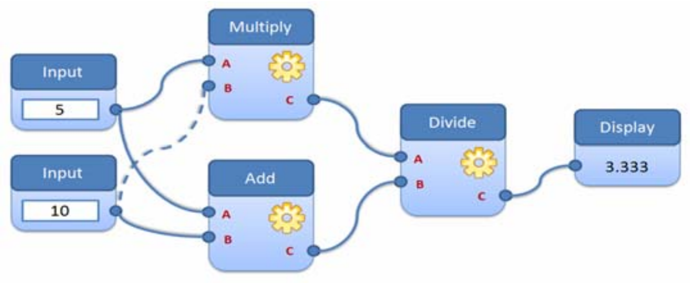

Fig. 2. Simple calculation Mashup

The corresponding BPEL process for this mashup is shown in Figure 3. The Mashup-BPEL convertor uses the available widgets and connections of mashups and formulates the widget services as BPEL invoke actions. The resulted business process can be then deployed on any BPEL engine for execution. At runtime, user will submit the required input values to this process and the rest of steps will be handled by BPEL engine.

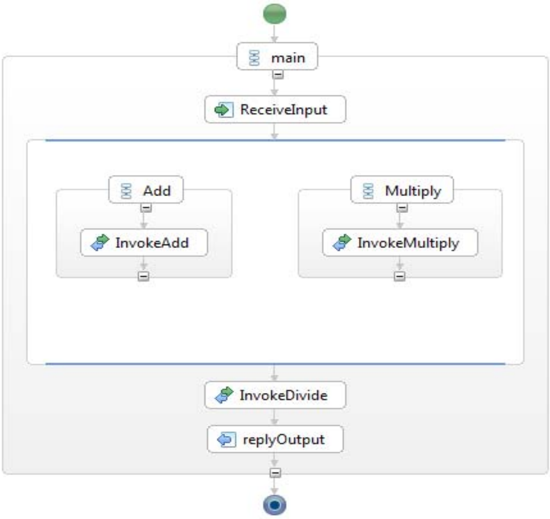

Fig. 3. The generated BPEL process for Simple calculation Mashup

The proposed approach for converting mashups to BPEL processes can get more complicated when an end-user's contribution is required. Unlike mashups where user inputs are simply added on screen, BPEL does not include the human interactions in its domain. Despite wide acceptance of Web services in distributed business applications, the absence of human interactions is a significant deficiency for many real-world business processes. For this reason BPEL4People extends BPEL from a sole orchestration of Web services to an orchestration of role-based human activities as well [18].

In context of the proposed approach, the required user interactions are implemented using XForms [19] to send the missing data to the business processes and provide a simplified BPEL4People integration. To clarify this, consider the previous example depicted in Figure 2 and suppose that the dashed connection between the input and multiply widgets is removed. As a result the corresponding BPEL can be run in one service call and will break on “multiply web service” invocation. At this point the end-user needs to provide the missing parameter for continuation of business process. In order to include the user contribution in the target business process, the Mashup-BPEL convertor will detect the location of missing parameter and inserts a people notification web service call before this location.

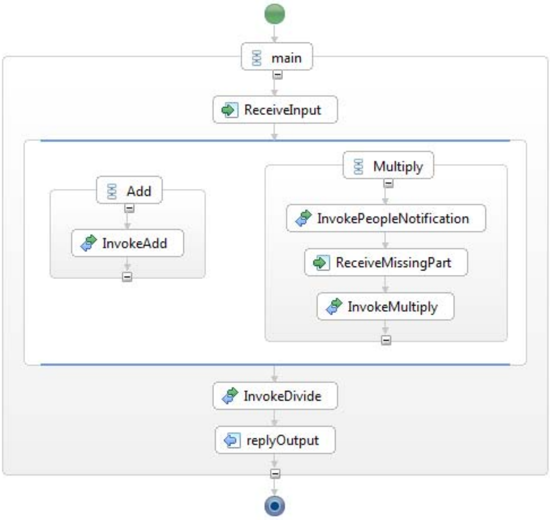

Fig. 4. The generated BPEL process for People Interaction

One great advantage of XForms is the fact that an XForms client can send its data as XML. This capability can be elegantly used to send a SOAP message which is also encoded in XML to end point of web service. In the context of the proposed solution, this feature of XForms has been used to receive the missed parameters from user.

5 Web Form Services

Nearly all human-computer interactions are performed through Input forms which are responsible for receiving the user input and sending it to appropriate component for further processes. Particularly, a large portion of Internet advances owes the human-computer interaction and data exchange via web forms and we are using them extensively in our daily activities. Current complex Internet applications demand a significant amount of time for development and maintenance of web forms which are solely designed for human users and cannot be automated and reused in business processes. In this section a novel approach for overcoming the web form integration obstacles, namely Web Form Services will be presented. Web Forms Services are a translation of ordinary web forms that provide a web service interface for corresponding forms and make the reuse and integration of the web form logic more convenient. For instance the converted web form functionality can be reused in Enterprise Mashups to automate data interaction between different web applications. This idea is especially helpful in automating the tasks between multiple web applications in a formal way. It is now very common in big enterprises to use web applications for different systems and usually there is no unified system that can take over all the tasks. So enterprises are now encountering a bunch of different web applications such as accounting, personnel management, etc., and each application is responsible for part of the business activities. In use cases requiring data records that are managed by different applications, end users should either wait for some complex backend integration from software architects, or manually transfer the data between applications to get the required results. In this context Web Form Services provide a unified way of describing the services and reusing them for different business activities. Figure 5 shows the WFS server and its components that are used to transform web forms to Web Services.

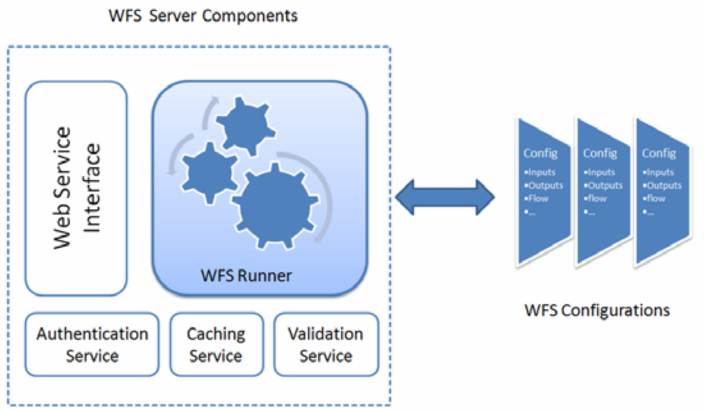

Fig. 5. Web Form Service architecture overview

The core element of WFS architecture is its configuration files that define the necessary interactions between system and a specific web form to achieve the required results. The configuration file captures such interactions in a formal way and at runtime the system will use these documented interactions to simulate the actions of a human user. The configuration files are also used to generate the definition of corresponding web services and presenting them as WSDL files that include input and output parameters.

6 Conclusions

The shift away from traditional Web 1.0, is forced by the growing need for more efficient information sharing, collaboration and business processes. Mashup Architecture is one of the outcomes of Web 2.0 paradigm that has been widely accepted and used for user-centric information processing. At the moment mashups are mainly used for less fundamental tasks such as customized queries and map-based visualizations; however it has the potential to be used for more fundamental and sophisticated tasks too.

As more serious applications make use of mashup architecture, there is a growing need to study the business chances and feasible scenarios of mashup architecture and foster its applications for organizational use cases. In our belief, mashups have the potential to facilitate the transition from traditional web to Semantic Web era and support this paradigm shift with “zero footprints” on the web pages. More specifically, the proposed approaches in this paper shows how elaborate different information integration solutions in combination with mashups can be used to facilitate the information integration in business processes and support creation of situational solutions that can be shared and reused by other users. Furthermore the translation of enterprise mashups to stable business processes can accelerate the application of mashups for more serious use cases such as organizational data sharing and data integration scenarios.

Acknowledgments. This research is partly supported by FIT-IT project, Secure 2.0 (project number: 820852).

References

1. Berners-Lee, T.: The Semantic Web. Scientific American (May 2001)
2. Anjomshoaa, A., Karim, S., Shayeganfar, F., Tjoa, A.M.: Exploitation of Semantic Web technology in ERP systems. In: Procs. of Confenis 2006 (2006)
3. Anjomshoaa, A., Bader, G., Tjoa, A.: Exploiting Mashup Architecture in Business Use Cases. In: NBIS 2009, Indianapolis, USA (2009)
4. A Business Guide to Enterprise Mashups, JackBe Corporation (April 2008)
5. RDFa in XHTML Syntax and Processing, http://www.w3.org/TR/2008/CR-rdfa-syntax-20080620 (last visited, November 2009)
6. Microformats, http://www.microformats.org (last visited, November 2009)
7. Music Mashups, http://en.wikipedia.org/wiki/Mashup_music (last visited, May 2009)
8. Really Simple Syndication, http://en.wikipedia.org/wiki/RSS(last visited, May 2009)
9. Representational State Transfer, Roy Thomas Fielding, Architectural Styles and the Design of Network-based Software Architectures, Ph.D. Thesis, University of California (2000)
10. Top 10 Strategic Technologies for 2009, Gartner Symposium/ITxpo, http://www.gartner.com/it/page.jsp?id=777212 (October 2008)
11. Hoyer, V., Stanoevska-Slabeva, K.: The Changing Role of IT Departments in Enterprise Mashup Environments. In: ICSOC 2008. LNCS, vol. 5472, pp. 148–154. Springer, Heidelberg (2009)
12. Mashup Basics: Three for the Money, Andy Dornan, http://www.networkcomputing.com/showitem.jhtml?articleID=201804223
13. Hoyer, V., Fischer, M.: Market Overview of Enterprise Mashup Tools. In: Bouguettaya, A., Krueger, I., Margaria, T. (eds.) ICSOC 2008. LNCS, vol. 5364, pp. 708–721. Springer, Heidelberg (2008)
14. Google Sidewiki, http://www.google.com/sidewiki/intl/en/index.html15. Dapper homepage, http://www.dapper.net/ (last visited, November 2009)
15. Lixto Solutions, http://www.lixto.com (last visited, November 2009)
16. Dornan, A.: Mashup Basics: Three for the Money (2007), http://www.networkcomputing.com/data-networking-management/mashup-basics-three-for-the-money.php (last visited, November 2009)
17. BPEL4People, BPEL4People, http://en.wikipedia.org/wiki/BPEL4People
18. XForms 1.1, W3C Recommendation (October 2009), http://www.w3.org/TR/xforms11/ (last visited, November 2009)