Multiview Web Service: The Integration of The Notion of View And Point of View in The Web Services

BOUKOUR Rida, ETTALBI Ahmed and NASSAR Mahmoud

University of Mohammed V-SOUISSI, ENSIAS, Rabat, Morocco

Summary
The adoption of Web services is a major step in the development of interoperable information systems. The composition of services makes it possible to meet the increasingly complex needs of users by combining several Web services in the same business process. The use of Web services in the information systems based on the Web is increasingly frequent. These systems need to be adapted as more and more users with different preferences access the Web by using a wide range of devices (computer, PDA, cell phone, etc. . .). To date, none of the three standards of web services have considered using the concept of view and point of view of web services. The focus of our article is to introduce the concept of view and point of view of Web services that will be called afterwards MultiView web service. This is to achieve benefits such as the interoperability between remote systems according to each user needs and preferences. To reach such benefits, we decomposed web services into sub services and we collected them according to the characteristics and properties of each user called point of view. It is an attempt to reach the ultimate goal to be represented depending on each user view.

Key words:
Web service, View, Point of view, Composition.

1. Introduction

The Web-based Information Systems (WIS) rely more frequently on a service-oriented architecture. This concept of architecture seeks to organize a set of isolated software applications into a set of interconnected services. Each one is accessible through the interfaces and standard communication protocols [7].
The SOA (Software Oriented Architecture) is used due to the numerous advantages. This concept is considered as an interoperability facilitated between the WIS and the pooling of services that can used by several WIS. A SOA can be achieved by the implementation of Web Services in a natural way.
WIS is a program that allows the interoperability between applications users through the Web [12]. This interoperability is possible because the WIS relies on SOA architecture. The different actors (supplier, applicant and register) communicate through standardized languages.
The service provider describes the WIS in WSDL (Web Service Description Language suggested by the W3C), and then saves it in a universal register (UDDI - Universal Description, Discovery and Integration suggested by OASIS) [16]. So a WIS applicant chooses a service to meet his needs through this register. The communication between the applicant and the provider is made through SOAP (Simple Object Access Protocol suggested by the W3C) [8].
With the emergence of new technologies of mobile terminals (PDA, mobile phone, etc.), the user can work and carry out various missions while moving freely from a place to another. This mobility requires new constraints of adaptation process. It must take into consideration not only the user profile (centers of interests, static characteristics…) but also his context (device, environment, localization...).
As a part of this work, we put emphasis on the possibilities of introducing the notion of view and point of view in the web services that will be called MultiView web service. To illustrate this concept, the MultiView web service is a service that adapts to the user's needs while considering his device, environment, location, and preferences. This enables the same web service to be seen differently from one user to another by decomposing web services into sub web services. Thus, each user needs are respected without changing the structure and the role of the web service.
The remainder of this paper is organized as follows. Section Two defines the context and then a brief overview of some works is introduced focusing on possibilities to introduce the adaptation to the context in Web Service .Section three presents our web service multiview approach. In section Four, we deal the multiview customer web service as a Case Study. We conclude this paper by presenting our further works.

Manuscript received February 5, 2014
Manuscript revised February 20, 2014

2. State of the art

2.1 The Web Services

A Web Service is an application that is made available on the Internet by a service provider. This application allows the interoperability between users throughout the Web. This interoperability is ensured by using standards and open protocols [6] [11]. The following ones can be cited:
- WSDL (Web Service Description Language) that describes the interfaces of WS [20].
- UDDI (Universal Description, Discovery and Integration) that includes the descriptions of WS [18].
- SOAP (Simple Object Access Protocol) that uses the descriptions of WS [19].

The life cycle of Web Services is shown in Figure 1: the service provider constructs and publishes the service description in the directory. The client makes a request to the directory seeking a service. The directory finds the service requested and provides it to the user. Once the service is found, the client will have a direct interaction with the service [11].

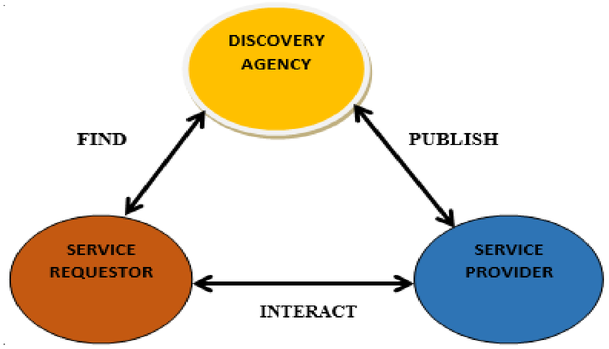


Fig.1 Basic Web services architecture.

2.2 The adaptation to Context in Web Services

The context is widely used and defined as a set of circumstances that follow an event. Based on this idea, we can talk about the context in which the interaction between an individual and a system takes place.

In the adaptation field, the researchers have not yet agreed to one generic definition of the notion of context. However, the definition of Day is among the widely accepted definitions: the context covers all the information that can be used to characterize the situation of an entity. An entity is a person, a place or an object that may be relevant for the interaction between the user and the application. The user and the application themselves are included [5].

Many studies have considered the context in the Web Services:

- The works of [3] [2] have keen interest in the impact of context on the application. This team developed a platform (SECAS) that makes applications adaptable to different contexts of use on their three constituents: data, services and presentation. An adaptation strategy was established in the medical field.
- The works of [14] [2] set out an information system in tourism (CATIS). It allows the adaptation to mobile devices in terms of content and presentation. User preferences, location and type of device are taken into account and the direction and speed of the user's previous location are calculated.

-The work of [17] [2] provides a platform that facilitates the development and the deployment of adapted Web Services to the context of use. They allow adaptation to mobile devices regarding the adaptation to the user location.

Our findings show that the above works present scheduled services in order to meet properly the specific needs of users.

2.3 Views and viewpoints in the object approach

The notion of view/point of view has been integrated in many object oriented languages. The latter use classic relations of object paradigm (heritage, delegation and composition) in order to implant this notion. In the same way, several works attempt to integrate the point of views in prototype languages

In the object approach by viewpoint, a view is defined as being a partial abstraction of the model; it is a sub-model. A viewpoint is a user's view of the model. A viewpoint is, thus, a combination of several views.

The introduction of the notion of viewpoint in object-oriented modeling of complex systems can elaborate a unique model that is shareable and accessible following next several viewpoints [4]. The advantage of this new approach appears at the consistency of data, deletion of some redundancy, enhancement of the multi-model approach and the definition of access rights.

In the object approach, various research works have dealt with the concept of views in order to integrate it. These include the work of [4], the VBOOM method (View Based Object Oriented Method) proposed by [9] and VBOOL language (View Based Object Oriented Language) put forward by [10] and VBOOL compiler [13].

3. Our approach

In this section, our objective is to integrate the concept of view, the point of view of web services called web service multi- views, and its implementation in a concrete case.

3.1 The integration of the notion of view, point of view in Web Services

A point of view means a position from which we observe or look at things: landscape, animals, a tree, a car ... It is clear that we can only see the surface of the object which is facing us. Thus, we can only examine the surface characteristics of the observed object.

The characteristics noticed by a point of view are relevant and valid according to this point of view. They depend on the location, environment, climate, the level of knowledge of the person, and his characteristics and needs …. The point of view of a person may also change when he or she changes their position or environment.

As noted above, the multi-view web service is a web service that can have many views according to the view of each user, in other words, it depends on the characteristics, needs, profile and properties of each user involving use of this web service. It can also be said that the same web service has a different point of view according to the view of the user.

The point of view of a web service will be generally the characteristics and properties of the user so that the web service can be adapted to the user. The view is the final presentation of the web service so that it can be displayed depending on the perspective of each individual.

3.2 The Decomposition of web services

The basic idea of architecture is to decompose web services into sub web services according to the user view then to collect them according to the point of view of each user. The Web services will be named according to the view of each user. The naming of these web services will enable the decomposition of the latter into sub web services in order to be adapted to each view (Figure 2).

Once the decomposition is made, the web services will be collected again in a well arranged and objective way according to the point of view of each user to meet his needs (Figure 2).

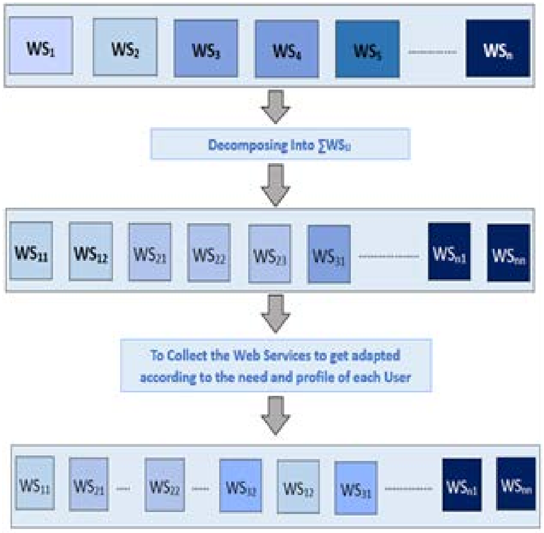


Fig.2 Decomposition of web services into sub web services.

The decomposition of web services into sub web services is especially based on needs and access right of users. Table1 presents an example of decomposition.

Table 1: Decomposing web services

Web Services	*Sub Web Services*
WS1	WS11, WS12, …, WS1a
WS2	WS21, WS22, …, WS2b
…	…
WSm	WSm1, WSm2, …, WSmc

The collection of web services into sub web services is especially based on a matrix called Users/WS Matrix (Table2).

Table 2: Collecting sub web services

USERS ***WS***	***U1***	***U2***	**…**	***Un***
WS1	WS11,…	WS12,…	…	WS1n,…
WS2	WS21,…	WS22,…	…	WS2n,…
…	…	…	…	…
WSm	WSm1,…	WSm2,…	…	WSmn,…

4. MULTIVIEW WEB SERVICE: Case Study

In our example we modeled system house car sales as a web service. It includes many basic sub web services that belong to different organizations: Car Purchase, Car Sale, the Credit Institution, Claim Service.... etc. The basic idea is to decompose these web services into sub web services to meet the needs of each user and also to respond to the mutliview notion.

Here, the house car manager has the right to add or remove a car brand or change the price. He has also the right to check the number of cars he sells, the number of cars remain in stock, cars that are being delivered and cars in the process of being ordered. He has also the right to change orders (add or cancel an order). The owner has the rights to deal with users complaints.

On the other hand, the customer has the right to see the car brands, their characteristics as well as the prices. He has also the right to check the availability of cars in stock and order a car or cancel an order that has already been made within a specific period. For some specific needs, a customer can file a claim. Additionally, the user has other services that are not compulsory to the manager like the credit service for the funding of cars purchased by costumers.

Differentiation between the user and the manager will be made by a simple authentication. The latter will be illustrated by a web service. (Figure 3) shows a set of intervening parties in this example.

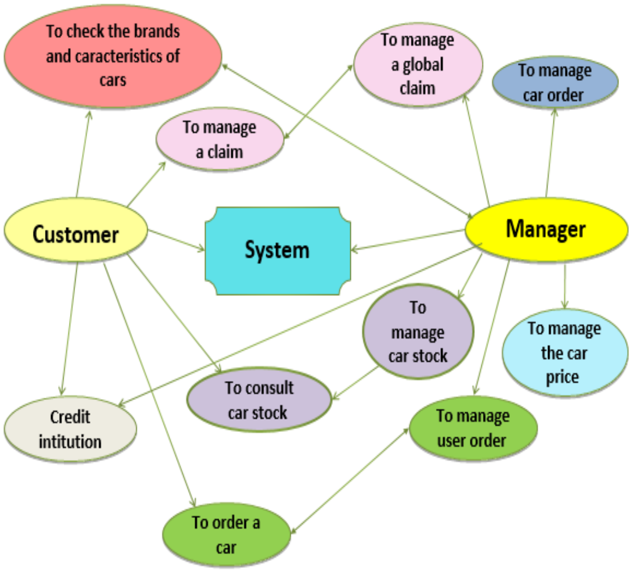


Fig.3 Interactions between customer and manager.

Table 3: Web services of the house car sales system

Web service name	*Web service description*
WS1 : Car purchase service	Allows actors to buy a car. In our example the user can be either a client or a manager and they will be differentiated by the authentication and the profile of each one of them. The customer can buy a car from a manager and the same thing for the Manager, he can buy cars from individual or factories
WS2 : Car sale service	Is a service dedicated to the manager. It allows the sale of cars to customers in response to a request by the purchase service.
WS3 : Claim service	Allows the actors to manage the claims according to the norms imposed by the service. The claims will be used by the two profiles, the customer and the manager, each one of them see and use only the options that suit his needs
WS4 : The credit service	Is only seen by the customers that intend to buy cars. It allows the users to request the credits from the bodies accredited by the manager.
WS5 : Car price service	Allows the users to consult prices of cars, the customer can only consult the prices and compare them, the manager can consult and modify the car price.
WS6 : Cars stock service	Allows actors to access to car stocks. It is accessible after authentication to be adapted to profiles of each actor. A simple user can only consult the available stock. A manager in addition to consultation can also add cars.
WS7 : User profile service	Allows adapting the conversation depending on the user profile, in our case the profile is whether the customer or the manager.

While considering the claim service example, it can be decomposed into sub web services, the customer can file a new claim or modify it as well as consulting on a claim that has already been filed while the manager is entitled to consult the claims filed by the customers and to process the claims.

The following table (table 4) contains the sub web services obtained after decomposing sub web services according to the actors of our system.

Table 4: Sub Web services of the system house car sales

Web Services	*Sub Web Services*
WS1	WS11: Consult a car to buy WS12: Buy a car
WS2	WS21: Sell a car WS22: Cancel a sale
WS3	WS31: Consult a claim WS32: File a claim WS33: Process a claim WS32: Delete a claim
WS4	WS41: Request a loan WS42: Consult a loan
WS5	WS51: Consult car price WS52: Add car price WS53: Modify car price WS52: Delete car price
WS6	WS61: Consult car stock WS62: Add car in the stock WS63: Modify car stock WS62: Delete car stock
WS7	WS71: Authentication for Manager, with administrator privileges WS72: Authentication for customer

	without administrator privilege

In table 5, we present the result of collecting Sub Web Services according to needs and access rights of the system users.

Table 5: Sub web services used by each user

USERS ***Web Services***	***MANAGER***	***CUSTOMER***
WS1	WS11	WS12
WS2	WS21, WS22	
WS3	WS31,WS33, WS34	WS32, WS31
WS4		WS41, WS42
WS5	WS51, WS52, WS53, WS54	WS51
WS6	WS61, WS62, WS63, WS64	WS61
WS7	WS71	WS72

5. Conclusion and perspectives

Web Services enjoy simplicity regarding its implementation: in fact, we can have easy access to existing application functionalities from the internet without having to modify fundamentally the information system of the enterprise. Hence, the web services use efficiently the Internet exchange standards.

Web services with its protocols and standards make progress toward greater standardization. The protocol exchange of messages SOAP and WSDL for defining the interface standardize the transport layer. The Web Services is based on strong foundations (SOAP and WSDL) that have proven their effectiveness and maturity although complete standardization does not yet exist.

In this paper, a thoughtful description of Web Services has been suggested (WIS) in order to adapt web services to various users. Hence, we introduced a notion of view and point of view to formalize the general need of the user and his profile on the basis of these two principles. The first parts for our work seeks to decompose web services into sub web services and collect them according to the characteristics and properties of each user known as point of view. This is to be finally displayed as a web service adapted to a given user called view. In perspective, we will implement the decomposition and collection processes to give to user a tool to facilitate the decomposing and collecting sub web services.

References

[1] Booch G., Rumbaugh J., Jacobson J., Le guide de l'utilisateur UML, http://www.omg.org/spec/UML/2.4.1/2000.

[2] Bouchra S, Dana k, Florence S, Adapted to web services, IRIT, University Paul Sabatier, 2007.

[3] Chaari T., Ejigu D., Laforest F. and Scuturici V. M., Modeling and Using Context in Adapting Applications to Pervasive Envir onments, IEEE International Conference on Pervasive Services, June 2006, Lyon, France.

[4] Coulette, B. Kriouile, A. Marcaillou, S., L'approche par points de vue dans le développement orienté objet des systèmes complexes, Revue l'Objet, vol. 2, n°4, p. 13-20, 1996.

[5] Dey A. K. et Abowd G. D., Towards a Better Understanding of Context and Context -Awareness, CHI 2000 Workshop on the What, Who, Where, When, and How of Context - Awareness, April 2000, The Hague, The Netherlands.

[6] Fauvet M. C., Models and Systems for Emerging Applications (continued) Chapter 6, Information Systems accessed via Web Services, November 14, 2005.

[7] Haas, H., and Brown, A., Web Services Glossary, W3C Working Group Note, février 2004.

[8] http://www.w3.org/TR/2000/NOTE-SOAP-20000508/.

[9] Kriouile, A., VBOOM, une méthode orientée objet d'analyse et de conception par points de vue, Thèse de doctorat d'Etat, Université Mohammed V de Rabat, 1995.

[10] Marcaillou, S, Intégration de la notion de points de vue dans la modélisation par objets – Le Langage VBOOL, Thèse de l'Université Paul Sabatier de Toulouse, 1995.

[11] Melliti T., Interoperability of complex Web Services. Application to multi-agent systems, Ph.D. Thesis, university Paris IX Dauphine, 8 décembre 2004.

[12] Michael Champion, Software AG, Chris Ferris, Eric Newcomer, David Orchard, Web Services Architecture, W3C, 14 November 2002.

[13] Nassar, M. Kriouile, A. Coulette, B., Programmation par objets et points de vue – le compilateur VBOOL, 6^{e} Conférence Maghrébine des Sciences Informatiques MCSEAI'2000, Fès, Maroc, novembre 2000.

[14] Pashtan A., Heusser A. and Scheuermann P., Personal Service Areas for Mobile web Application, IEEE Internet Computing, November -December 2004.

[15] Rundensteiner, A., A Classification Algorithm for Supporting Object-Oriented Views, Proceedings of the 3rd International Conference on Information and Knowledge Management (CIKM' 94), pages 18-25, Gaithersburg, Maryland, USA, ACM Press, 1994.

[16] The Universal Description, Discovery and Integration (UDDI) protocol. Version 3, 2003. At http://www.uddi.org/.

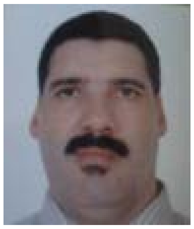
[17] W3C, Composite Capability/Preference Profiles (CC/PP): Structure and Vocabularies 1.0. W3C Recommendation 15 January, 2004. The latest version: http://www.w3.org/TR/CCPP-struct-vocab/, January 2007.

[18] W3SCHOOLS, http://www.w3schools.com/wsdl/wsdl_uddi.asp, December 2006. ARGYRIS C., Knowledge for action: overcoming barriers to organizational learning, InterEditions, 1995.

[19] W3C, recommendation W3C 24 Juin 2003. The latest version: http://www.w3.org/TR/soap12 -part0/, December, 2006.

[20] W3C, Web Services Description Language (WSDL) 1.1, W3C Note 15 M arch 2001. The latest version: http://www.w3.org/TR/wsdl. December, 2006.

BOUKOUR RIDA PhD student systems engineering and models AT the Higher National School for Computer Science and System Analysis (ENSIAS), Rabat. Administrator senior system.



ETTALBI Ahmed Professor at Software Engineering Department of the Higher National School of Computer Science and Systems Analysis (ENSIAS) Rabat. His main research interests Object Modeling with Viewpoints, Software Architecture and Business Process Modeling architecture.

Mahmoud NASSAR Professor Head of Software Engineering Department. Ph.D. in Computer Science from the National Polytechnic Institute of Toulouse, 2005. Research Theme: Context-Aware Service-Oriented Computing, Engineering Component-based, Model-Driven Engineering Subjects taught: Service Oriented Architecture (SOA), bases distributed Administration Database Management Systems database (ORACLE), distributed object systems, Compilation, Algorithms and Programming data.