Journal of Theoretical and Applied Information Technology
31st August 2013. Vol. 54 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

THE USER-CENTRIC SOA AND ITS IMPACTS ON OTHER DISCIPLINES

[1]MERIEM BENHADDI, [2]KARIM BAÏNA, [3]EL HASSAN ABDELWAHED

[1]PhD candidate, Faculty of Sciences Semlalia, Cadi Ayyad University, BP 2390-Marrakesh, Morocco

[2]Habilitated Professor, ENSIAS, Mohammed V Souissi University, BP 713 Agdal-Rabat, Morocco

[3] Professor, Faculty of Sciences Semlalia, Cadi Ayyad University, BP 2390-Marrakesh, Morocco

E-mail: [1]m.benhaddi@uca.ma, [2]baina@ensias.ma, [3]abdelwahed@ucam.ac.ma

ABSTRACT

The end user service development known as the user-centric SOA emerged as a new approach that allows giving the end user the ability to create on the fly his own applications that meet a situational need. In fact, the classical SOA was designed for developers and is characterized by a heavy technical stack which is out of reach of end users. The user-centric SOA approach brings great added value taking advantage from the creativity and the participation of end users; furthermore, it is becoming a key factor of competitiveness within enterprises and is influencing other disciplines. The Cloud Computing and the Enterprise Architecture are two fields that captured growing attention in recent years and that can gain maturity and efficiency from adopting and embracing the user-centric SOA approach. In this paper, we present the user-centric SOA approach and we give an overview of our solution for an efficient user-centric services creation. Then we study the impact of the user-centric SOA on the Cloud Computing and Enterprise Architecture fields.

Keywords: *SOA, End User Development, Integration Patterns, Cloud Computing, Enterprise Architecture.*

1. INTRODUCTION

1.1 The New Era of End User Services Development.

The End User Development (EUD) approach puts the user at the front and gives him the ability to create on the fly its own applications that meet a situational need. [1] affirms that the new trend for the next few years will be to make systems easy to develop and create new environments that allow non technical users to develop applications. We cite the EUD definition given by [1]: ”End-User Development can be defined as a set of methods, techniques, and tools that allow users of software systems, who are acting as non-professional software developers, at some point to create, modify or extend a software artefact”.

[2] predicts that the number of end-user developed applications will far exceed the number of professionally developed applications (integration, composition, automatic generation or infrastructure): 55M performers in end-user programming against 2.75 M professional developers.

The rest of this paper is organized as follow: sections 1.2, 1.3 and 1.4 give a definition of the end user and the user-centric SOA, and show an example scenario of the user-centric SOA. Section 2 presents an overview of our contribution to an efficient user-centric SOA. Sections 3 and 4 discuss the impacts of the user-centric SOA approach on the Cloud Computing and the Enterprise Architecture fields.

1.2 End User Definition

A software end user is a person who interacts with information systems solely as a final information consumer. It’s a user with minimal technical knowledge, and who uses the software in the context of daily life or daily work for personal (business or leisure) purposes, without having any intentions to produce other systems. He is not interested in computers per se, and do not worry about system technologies as long as he can get what he needs quickly [3] [4].

End users have many requirements that should be respected by system designers and developers in order to deliver systems that satisfy end users. Based on the work of [5] and [6], we have grouped

Journal of Theoretical and Applied Information Technology
31st August 2013. Vol. 54 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

into four criteria the end users requirements, which are listed in table 1.

Table 1. End User Requirements

Criteria	***Description***	***Problem of criteria lacking***
Functional richness	Features requested to execute different tasks.	Limited set of offered features.
Usability & intuitiveness	User interfaces, interaction and dialogue mode.	Lack of visibility, feedback, consistency, non-destructive operations, discoverability, scalability, reliability [7].
Efficiency, reliability, maintainability and portability (ERMP)	Difficulties that do not refer directly to system features.	Lack of documentation, performance, security, supportability.
Personalizability , customizability	Capability of end user to tailor themselves their systems.	Useless systems that lack many important features.

1.3 Why a User-Centric SOA?

According to Gartner analysis [8], there are several key factors that changed the picture of services supply to end users:

- Users are more technologically savvy and have very different expectations of technology.
- The Internet and social media have empowered end users by offering them the opportunity to develop their skills.
- The rise of powerful, affordable mobile devices has improved the accessibility, reachability and availability of technologies for end users of all types.
- End users have become innovators and participate in the creation of added value in the internet or within enterprises.

These new facts led to think of a new generation of services development: the end user services creation. Hence, we define the user-centric SOA as the expectation of end users, their future hope, and the promise for better information systems (figure 1). A user-centric SOA offers:

- Empowerment of the end user: Easy and flexible composition on the fly of services by all end users.
- Openness of the Information System to the public: the democratization of SOA and the installation of the global SOA or the Internet of Services [9].
- More independence of SOA: the adoption of a variety of interoperable technologies in order to meet the great variety of the web.
- Lightweight SOA technologies: the support of SOA technologies by all mobile devices. According to [10], traditional SOA is lacking mobility; in fact, SOA implementation and integration technologies are very heavy for devices with limited capabilities. WSDL and SOAP are instances of complicated XML documents, which makes the WS* services very demanding in terms of computing power, bandwidth and storage.

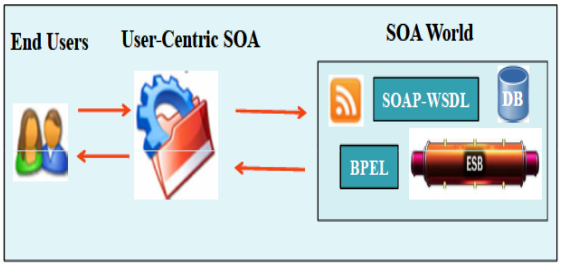

Figure 1: User-Centric SOA Concept

Our Objective consists of formalizing and enhancing the user-centric services composition by:

- Introducing a new rich integration language,
- Proposing new intuitive & self-explanatory semantic methodology for end user services composition.

1.4 User-Centric SOA Scenario: Public Health Use Case

As an example of an end user service creation, let us describe a public health field use case that can be used by any end user. Our end user, Mark, got diabetes with kidney complications. Mark lives in a small city, so he wants to plan a medical consultation with a lower cost by comparing costs in three different neighboring cities. Then Mark would like to search for kidney doctor addresses in the city with the lower cost, and display the addresses on a map. In order to watch his diet,

Journal of Theoretical and Applied Information Technology
31st August 2013. Vol. 54 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Mark would like also to have a list of diabetes products that are sold in the supermarkets of the city with the lower medical consultation cost, so he could both visit a doctor and buy the diet products (figure 2).

While the web does not offer a solution to Mark, but several separate services that Mark has to compose, there is a need for an environment that offer intuitive user interfaces, intuitive and rich constructs, intuitive & semantic data mapping and helpful & interactive environment.

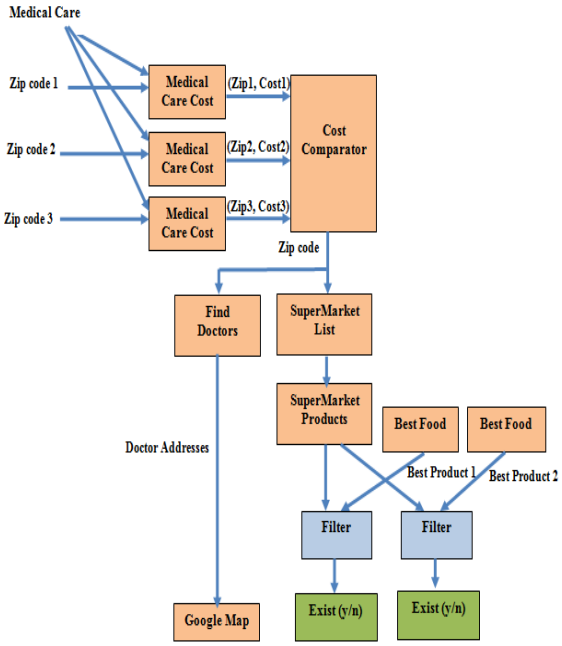

Figure 2: User-Centric SOA Scenario

2. OUR NEW USER-CENTRIC APPROACH

In this section, we present in a brief our approach for a user-centric services composition. More details on our approach, the implementation results and the related work can be found in [11] [12].

2.1 Richness via the Enterprise Integration Patterns

Enterprise Integration Patterns (EIPs) [13] consists of sixty five (six groups) patterns that propose common solution to integration problems. EIPs are used by sophisticated mediation buses such as Camel, Mule and Apache in order to achieve very complex integration scenario.

In order to meet the first end user satisfaction criteria (functional richness), we have developed a user-centric composition language based on these Enterprise Integration Patterns. The different basic elements that form our new EIPs-based language are listed in table 2.

Table 2. Constructs Of Our New Eips-Based Language

Construct	Description
Task	is the goal of the end user performing the integration. Each task can have a frequency of execution.
Tag	key words used to describe a task
Mashup	A Mashup application represents the realization of a task and includes a set of integration taking place between several resources.
Process	Is the composition process of the Mashp application resources and consists of parallel or sequential integration flows.
Step	Is a step in the integration process and consists of a link between two or several components.
Component	Is the integration process node: resource, input of the end user, router or translator.
EndUser	Represents the interaction with end users during the integration process.
Resource	Represents the applications to integrate by the Mashup. A resource is described by its type, address and exchange format.
Expose Resource	Represents an exposed resource with input and output variables. The same resource can be exposed many times within the integration process.
Channel	Allows communication between two components and supports the single atomic integration step.
Message	is the entity transferring in a channel between two components.
Router	Is a node forwarding messages between resources, end user fields or translators.
Translator	Is the messages translation node.
System Manager	Each Mashup application can have one or several managers to improve reliability and maintainability.
Transaction	End users may want to synchronize actions of components to realize a transaction.

2.2 Intuitiveness via Semi-Automatic Services Composition

The user-centric development has revolutionized the development world by introducing new concepts. In fact, there is no need to master a programming language to be able to create a new application. A new era of development environments emerged to enable everyone to create new applications. To create intuitive, easy to use and self-explanatory development environment, we have put forward and implemented new concepts:

Journal of Theoretical and Applied Information Technology
31st August 2013. Vol. 54 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

1) Goal composition: The end user point of view

End users compose services based on the end-user knowledge: objective of the goal, final result, frequency, degree of importance, duration.

When creating new applications, end users try to achieve a new goal by composing existing sub-goals (figure 3). Each sub-goal is represented by a service. In this way, when composing services, end users try to resolve a problem whose solution does not exist yet on the web. In fact, the answer exists in the form of many subparts – services – dispersed on the web.

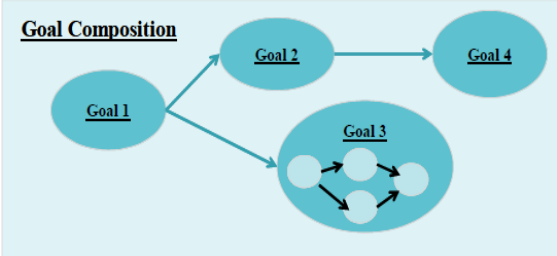

Figure 3: Goal Composition vs Service Composition

2) Goal Patterns-based suggestions

When compositing goals/services in response to a new goal, the inexperienced end user faces many challenges (ex. determine the types of resources, find resources that meet the end user criteria (quality, price, etc.), determine necessary actions for the use of interfaces (selection problems), determine how to arrange and coordinate resources (integration), etc).

The system has the role of helping end user to answer these different questions, by suggesting resources, providing guidelines for the coordination of resources and providing feedback and documentation for each selected action.

Goals patterns represent common and repetitive use cases, and can also be called end users experience patterns since they are driven from the end users experience. They provide answers to questions like "How to automate the execution of two consecutive tasks - eg. Turn on the light on the entrance of the house and turn on the heating - in response to a triggered event? - ex. presence of a person detected by the sensor.

Figure 4 shows two examples of goals patterns: 'Health Care' and 'Smart Home'.

While software design patterns are derived from the experience of the software developers, goals patterns are created, improved and enriched by end users themselves.

The system will utilize the goals patterns database to suggest to the end user links and components in order to build new applications. In fact, end users will be guided in the process of services composition through the database of goals patterns that contains the possible links between the various goals/services.

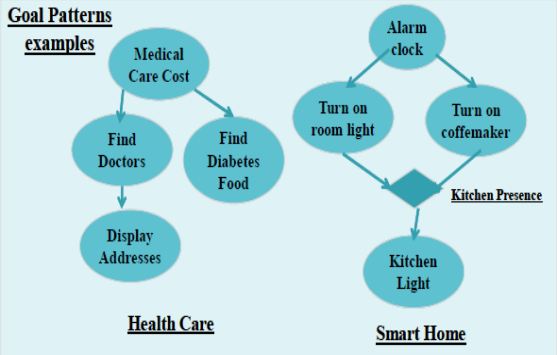

Figure 4: Goal Patterns Examples

Our suggestion model is similar to e-mail interfaces - ex. Gmail. When writing a new message, and when the first recipient address is entered by the user, other addresses are proposed and suggested at the basis of the previous messages sent by this user.

3) Semantic & Form-based data mapping

In order to compose goals/services, end users use their knowledge consisting on the objective of the goal, the final result, the frequency, the degree of importance, the duration, etc. This end-user knowledge represents the semantic which, alone, should be involved in the interaction between the end user and the user-centric SOA platform. Indeed, the service-to-service interaction, which is based on the syntax, is not valid at the interface level. The interface provides graphical display of services (called gadget) that represent sub-goals, which is an abstraction of services; therefore, the interaction and communication way at the interface level should also be an abstraction of the communication way between services.

In addition, using forms is the easiest way for end users to interact with interfaces. In fact, [14] [15] define five primary style of user interaction design, among which the forms that provide simple data entry and are easy to learn.

4) No prior knowledge of the entire composition design: Dynamic results visualization using a tree.

The static services composition used by existing end user services composition frameworks does not

Journal of Theoretical and Applied Information Technology
31st August 2013. Vol. 54 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

meet end users needs. In fact, in some situations, end users do not have a clear understanding of how to design the entire composite application. A tree-like and step by step displaying of results helps end users to determine the future actions.

Figure 5 shows the interface of our new framework, which implements the four user centric concepts described above.

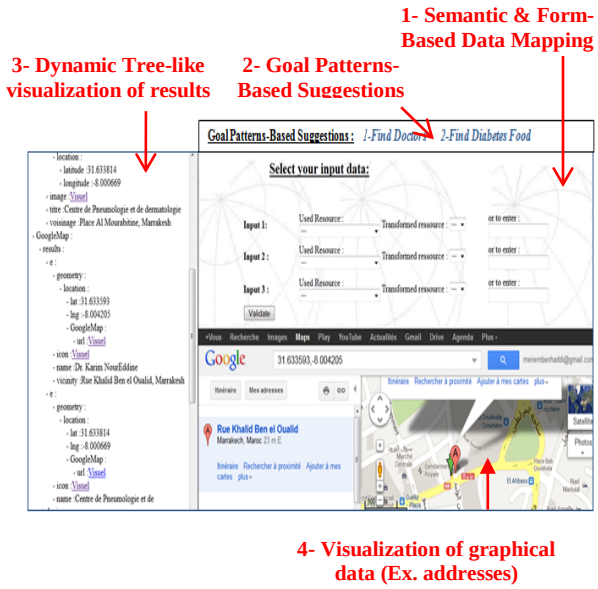

Figure 5: Intelligent interface with suggestions

3. THE IMPACTS OF THE USER-CENTRIC SOA ON THE CLOUD COMPUTING

According to [16], the Cloud Computing is an emerging paradigm that is based on compute and storage virtualization to deliver reliable services to customers. Customers can access data and applications anywhere in the world on demand.

In recent years, the Cloud Computing emerged as one of the most important area and gained interest in both academic and industrial world. In fact, the Cloud Computing promises reliability, efficiency and availability of services.

3.1 The Cloud Computing and the SOA

The Service Oriented Architecture is among the most important key Cloud Computing concepts. In fact, Web Services technology and SOA allow managing Cloud services and distributed storage for backup and world-wide data access [17]. Three service models exist: 1-Software-as-a-Service (SaaS) that represents web-based applications such as web-based email, 2- Platform-as-a-Service (PaaS) that allows customers to build and deploy their applications and services using the provider tools, and 3- Infrastructure-as-a-Service (IaaS) that provides processing power and storage. A good Cloud Computing services taxonomy can be found in [18].

3.2 Towards a User-Centric Cloud Computing: Clouds on the Cloud

The Cloud Computing is a rich research filed and is subject to constant flow of innovative ideas as it affects several area (hardware, software development, software security, etc). The user-centric development is one of these area that can bring a big added value to the Cloud Computing. In fact, considering the Cloud in its current conception (non user-centric), the end user has a large but limited choice of services. The end user can benefit from all the Cloud resources, but he does not have the power to change it in order to meet his new needs. [19] highlights this problem and says that the end user have to use isolated services that he must separately configure. Example is complementary services that cannot share contact lists or other user personal data.

According to [20], a private Cloud is an infrastructure operating solely for an organization and managed by the organization itself or by a third party. This means that private Clouds reflect the need of organizations, not individuals. This leads us to say that a private Cloud is different from the new user-centric Cloud concept.

On the other hand, personal Clouds (public/personal or private/personal Clouds) consist of a move of users' data and services from the personal devices to Clouds. Hence, personal Clouds are about the physical location of data and services, while user-centric Clouds are about empowering end users to create new services.

By introducing the new concept of end user development into the Cloud Computing, we bring a new breath that could lead to a powerful new Cloud Computing generation. In fact, empowering end users to create easily new services can lead to a profusion of services. Every end user will have the possibility to create his own new Cloud that he decides to make private or to share to become part of the public Cloud. This way, several new end user Clouds will be born to enrich the public Cloud and lead to what we call "*Clouds on the Cloud*" (figure 6).

[19] speaks in this context about personal Clouds in the context of an "activity sphere". [21] speaks about the personalization in the Cloud Computing and evoke usability and user issues.

Journal of Theoretical and Applied Information Technology
31st August 2013. Vol. 54 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Moreover, the user-centric development environments can themselves rely on the Cloud Computing services to provide basic infrastructure services and to give rich constructs to the end users. For example, Enterprise Service Buses could be used for their routing and translation capabilities, BPEL engines could be used for their orchestration capability and the CRUD services offer different services such as identity management, persistent storage, resources access, routing and translation (figure 7). This way, customers do not have to use the environment integration services if they find better services in the cloud.

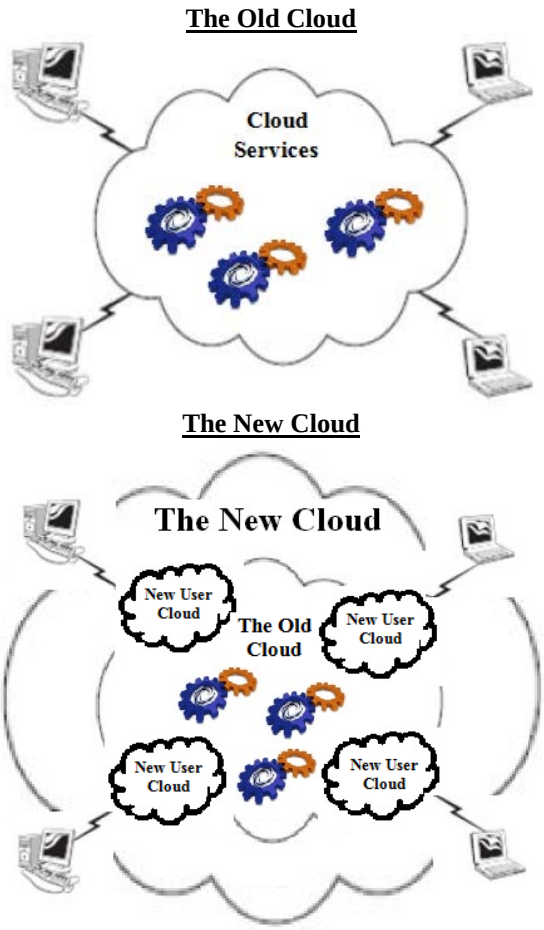

Figure 6:User-Centric Cloud Computing

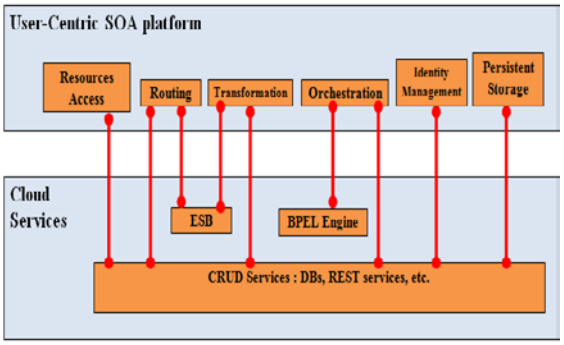

Figure 7:Cloud-Based User-Centric SOA Platform

4. THE IMPACTS OF THE USER-CENTRIC SOA ON THE ENTERPRISE ARCHITECTURE

According to [22], Enterprise Architecture (EA) is "the set of descriptive representations (i.e. models) that are relevant for describing an Enterprise such that it can be produced to management's requirement (quality) and maintained over the period of its useful life (change)". In other words, EA allows the alignment between business and IT.

In recent years, enterprise architecture has captured growing attention as a means to align business and IT artefacts in order to provide holistic decision support [23].

4.1 The EA and the SOA

[24] highlights the strong relationship between the Service Oriented Architecture and the Enterprise Architecture and states that the SOA supports EA in many facets. In fact, EA provides answers to allow linking processes to performance measures, while SOA offers IT solutions and robust platform upon which the business can apply their applications [24] [25]. [26] asserts that SOA is part of EA and that there should be a synergy between the EA and the SOA world to bring the needed enterprise capabilities.

Enterprise Integration (or system integration) is the capability to integrate a variety of different system functionalities (business processes and data)[27]. Enterprise Integration is an important component of enterprise architecture. In fact, in the '*New Enterprise*' [28][29], the faculty to connect applications quickly is becoming a necessity and a competitive advantage. Moreover, with the explosion of endpoints (Application Programming Interfaces or APIs) and data, the development model shifts from writing lots of code to composing and integrating APIs together [30].

[31] states that sustainable Enterprise Architecture can be achieved through Enterprise

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Application Integration that provides integration not only on a technical level but on a business process level too.

The Service Oriented Architecture provides application integration through sophisticated Enterprise Service Bus, which uses the famous Enterprise Integration Patterns that we used in our contribution (section 2.1 of this paper).

4.2 Toward a User-Centric Enterprise Architecture

With the new era of internet (open, mobile and big data, explosion of applications, data sources and open APIs, social internet, etc), system integration is becoming equal to a competitive advantage [30]. In fact, in order to bring more added value, the enterprise information system has to be open to the public (customers and partners). This brings new constraints and challenges to enterprises; the next generation of integration has to be:

- Fast: enterprises need instant response to any new need for integration in order to create more added value from existing hundreds of applications.
- Lightweight: the number of customer mobile devices is increasing; thus enterprises need to use adequate and lightweight – integration - methods and technologies in order to incorporate these devices.
- Powerful: to be able to provide adequate response to any new need, enterprises should utilize powerful and strong integration methods – patterns - and technologies that provide a solution to any integration problem.
- User-centric: the end user dimension has to be taken into account to involve customers and partners in the production of added value.

[30] talks in this context about a new wave of application architecture based on the introduction of API (Application Programming Interface) and "Integration" components.

According to the definition that we gave in section 1.3, the user-centric SOA offers openness of the information system, lightweight technologies and empowerment of the end user. The use of a user-centric SOA into enterprises, known as enterprises 2.0 [32][33], will generate a great added value as it fosters and eases the creation on the fly of new services, and alleviates the IT department burden. In addition, using a user-centric SOA by enterprises allows customers and partners to create services that meet their needs, which is known as "co-creation" [34], thus establishing close and solid relationships between each part, and improving agility and innovation of the enterprise.

The mobility is an important dimension in the enterprise world. With the new lightweight user-centric SOA consisting on lightweight implementation and integration technologies, mobile devices will be easily integrated within the overall scope of the enterprise.

4.3 Challenges in User-Centric Enterprise Architecture

There are some risks in adopting Enterprise 2.0 and especially a user centric SOA in enterprises. In fact, enterprises should tackle many challenges such as security, governance, administration, repository management, user support and quality of service. [35] published also an Enterprise 2.0 Adoption Survey and listed a numbers of barriers to adoption of enterprises 2.0 technologies. Among the faced challenges, the culture or resistance to change, difficulty in measuring ROI (Return On Investment), integration with existing technologies, security concerns, and budget. [36] makes it clear that the user-centric environment must be supported by a solid and robust infrastructure. [35] says that a major obstacle is that Enterprise 2.0 requires management to give up control. In fact, the proposed approach of a user-centric SOA will foster the creation of new services, thus giving birth to a large number of new functionalities that could manipulate critical data in enterprises. Privacy of data is also an important issue that the user-centric SOA environment should tackle, and that raises when end users integrate data in real life and more importantly life hostile problems [37][38].

To overcome the security and privacy problems, it is very important to focus on the enterprises governance; enterprises managers must have a clear policy towards the use of new technologies and create a strategy allowing the secure and successful adoption of a user-centric SOA. The strategies adopted should take place within the "Enterprise Infrastructure"; [38] propose a declarative approach to automatically combine data taking into consideration the data privacy constraints deduced from privacy policies, which determine the services that could be created by each role.

5. CONCLUSION

In this paper, we presented the user-centric SOA approach by defining it and by showing its usefulness in the daily life of end users. In fact, the user-centric SOA approach describes the

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

expectations and the future hope of end users of all types. We presented an overview of our solution for an efficient end user services creation, where we highlighted new concepts that help achieving end user satisfaction. Then, we discussed the great impacts that the user-centric SOA approach has on other fields such as the Cloud Computing and the Enterprise Architecture. Our discussion ended by pointing out some challenges in embracing the user-centric SOA approach in the Enterprise Architecture field.

REFRENCES:

[1] Lieberman. H., Paternò. F., Klann. M., and Wulf. V. (2006). 'End-User Development: An Emerging Paradigm', in End User Development. Kluwer Academic Publisher,2006, pp. 1-8.

[2] Boehm. B., Bradford. C., Horowitz. E., Westland. C., Madachy. R., and Selby. R. (1995). 'Cost models for future software life cycle processes: COCOMO 2.0'. Journal of Annals of Software Engineering, Special Volume, Vol. 1, No. 1. (16 December 1995), pp. 57-94.

[3] Cypher, A.(1993). 'Watch What I Do: Programming by Demonstration'. The MIT Press, Cambridge.

[4] Allison, H and R. Kelly, R. (1992) 'The Influence of Individual Differences on Skill in End-User Computing'. Journal of Management Information Systems I Summer 1992, Vol. 9, No. 1, pp. 93-111. (1992).

[5] McCall, J.A., Richards, P.K., and Walters, G.F. (1977) Factors in Software Quality, RADC TR-77-369, 1977, Vols I, II, III, US Rome Air Development Center Reports. Italie. (1977).

[6] ISO/IEC 9126-1. (2001) Software engineering – Product quality - Part 1: Quality model. ISO.

[7] Norman, D. and Nielsen, J.. (2010) 'Gestural Interfaces: A Step Backward In Usability'. Interactions' magazine, Vol. 17 Issue 5, September + October 2010 ACM New York, NY, USA.

[8] Gartner Newsroom: Gartner Says the Personal Cloud Will Replace the Personal Computer as the Center of Users' Digital Lives by 2014. March, 2012.

[9] Schroth, C. and Janner, T. (2007) 'Web 2.0 and SOA: converging concepts enabling the internet of services'. Journal of IT Professional, Vol. 9, No. 3, pp.36–41. (2007).

[10] Guinard, D. and Trifa, V. (2009) 'Towards the Web of Things: Web Mashups for Embedded Devices'. Paper presented at The 18th Int World Wide Web Conference, April, 2009, Madrid, Spain.

[11] Benhaddi, M., Baïna, K. and Abdelwahed, E. (2013). 'SOA for the Masses : End Users as Services Composers'. Paper presented at The 10th ACS/IEEE International Conference on Computer Systems and Applications, Fès-Ifrane, Morocco, Mai 27-30 2013.

[12] Benhaddi, M., Baïna, K. and Abdelwahed, E. (2013). 'Formalization of The User-Centric SOA Approach'. Paper presented at The 15th International Conference on Enterprise Information Systems, Angers, France, July 4-7 2013.

[13] Hohpe, G. and Woolf, B. (2003) Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions (Book). Addison-Wesley Professional.

[14] Sommerville. I. (2009). Software Engineering (Book). Publication Date: March 13, 2010 | ISBN-10: 0137035152 | ISBN-13: 978-0137035151 | Edition: 9.

[15] Shneiderman, B. (1998). Designing the User Interface (Book). 3rd edition. Reading, Mass.: Addison-Wesley.

[16] Buyya, R., Yeo, C. and Venugopal, S. (2008) 'Market-Oriented Cloud Computing: Vision, Hype, and Reality for Delivering IT Services as Computing Utilities'. Paper presented at The 10th IEEE International Conference on High Performance Computing and Communications (HPCC-08), pages 25{27, Los Alamitos, CA, USA, 2008. IEEE

[17] Wang. L., Laszewski. G., Younge. A., He. X., Kunze. M., Tao. J. and Fu. C. (2010). 'Cloud computing: a perspective study'. New Gener Comput 28(2):137. doi:10.1007/s00354-008-0081-5

[18] Höfer, C.N. and Karagiannis, G. (2011) 'Cloud computing services: taxonomy and comparison. Journal of Internet Services and Applications', 2 (2). pp. 81-94. ISSN 1867-4828

[19] Ardissono. L., Goy. A., Petrone. G. and Segnan. M.. (2009). 'From Service Clouds to User-centric Personal Clouds'. Proceedings of IEEE 2009 Int. Conf. on Cloud Computing (CLOUD-II 2009), pp 1–8. E-ISBN : 978-0-7695-3840-2.

[20] Mell. P. and Grance. T. (2011). 'The NIST definition of Cloud Computing'. Computer Security Division. Information Technology Laboratory. National Institute of Standards and Technology. Gaithersburg, MD 20899-8930. September 2011.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[21] Joshi. K.P., Yesha. Y., Ozok. A.A., Yesha. Y., Lahane. A., Kalva. H., Agarwal. A. and Furht. B. 'User Centric Smart Services on the Cloud'. Published in: 'The smart internet' (Book). Pages 234-249. Springer-Verlag Berlin, Heidelberg. October 31, 2010.

[22] Zachman, JA. 'Enterprise architecture: The issue of the century'. Database Programming and Design. 1997.

[23] Alwadain. A., Rosemann. M., Fielt. E. and Korthaus. A. (2011). 'Enterprise architecture and the integration of service-oriented architecture'. In Proceedings of 15th Pacific Asia Conference on Information Systems (PACIS 2011), AIS Electronic Library, Queensland University of Technology, Brisbane, QLD.

[24] Kistasamy. C., Van Der Merwe. A. and De La Harpe. A. (2010). 'The Relationship between Service Oriented Architecture and Enterprise Architecture'. Published in: · Proceeding EDOCW '10 Proceedings of the 2010 14th IEEE International Enterprise Distributed Object Computing Conference Workshops Pages 129-137.

[25] Rohan. D. (2008). 'How BPM, SOA and EA Can Meet True Customer Needs'. Published in 'BP Trends'. September 2008.

[26] Paras. G., DeBoever. L., and Westbrock. T. (2007). 'Ea Practitioners "Cheatsheet": 12 Ideas You Can Use Today to Improve Your Ea Team‟s Effectiveness'. Retrieved 13 Juin 2013, from http://www.eadirections.com/uploads/file/EAdirections%2012%20Ideas%20Webinar%20Nov-13-07%20Final.pdf

[27] Jinyoul. L., Keng. S. and Soongoo. H. (2003). 'Enterprise Integration with ERP and EAI'. Published in Magazine Communications of the ACM CACM Homepage archive Volume 46 Issue 2, February 2003. Pages 54-60.

[28] Gunasekarana. A., Williams. H.J. and McGaughey. R. 'Performance measurement and costing system in new enterprise'. (2005).. Technovation Journal, May 2005, Volume: 25 Issue: 5 pp.523-533.

[29] Marin. D. and Verdier. T. (2003). 'Globalization and the "New Enterprise"'. Journal of the European Economic Association. Vol. 1, No. 2/3, Apr. - May, 2003.

[30] Mason. R. and Landgraf. A. (2012). 'Application Architecture: The Next Wave'. Retrieved 13 Juin 2013, from http://www.mulesoft.com/sites/default/files/application-architecture-the-next-wave-1.2.pdf

[31] Aier, S. and Schönherr, M. (2005). 'Sustainable Enterprise architecure with EAI–An Empirical Study'. Proceedings of the International Conference on Advances in Internet, Processing, Systems, and Interdisciplinary Research, MIT, Cambridge, Boston MA, USA, IPSI, 2005.

[32] Bughin. J. (2008) 'The rise of Enterprise 2.0'. Journal of Direct, Data and Digital Marketing Practice. Vol.9, No. 3, pp.251-259.

[33] Soriano. J., Lizcano. D., Cañas. M.A., Reyes. M., and Hierro. J.J. (2007).'Fostering Innovation in a Mashup-oriented Enterprise 2.0 Collaboration Environment'. ICABS 2007, Proceeding of the SIWN International Conference on Adaptive Business Systems. Chengdu, China. Vol.1, No.1 pp. 62-69.

[34] Ind. N., Iglesias. O.. and Schultz. M. 'Building Brands Together: Emergence and Outcomes of Co-creation'. California Management Review Journal, Volume55, Pages5-26. 2013.

[35] Enterprise 2.0 Conference (2009). 'Enterprise 2.0: What, Why and How. May 2009'. White paper published by the Enterprise 2.0 Conference. San Francisco, USA. Retrieved 13 Juin 2013 from http://www.e2conf.com/downloads/Enterprise_2_0_what_why_how.pdf

[36] López. J., Pan. A., Bellas. F. and Montoto. P. (2008). 'Towards a Reference Architecture for Enterprise Mashups'. Jornadas de Ingeniería del Software y Bases de Datos. Vol.2, No.2, pp.67-76.

[37] Bhowmick. S.S, Gruenwald. L., Iwaihara. M. and Chatvichienchai. S. (2006). 'Private-iye: A framework for privacy preserving data integration', in ICDE 2006, Proceeding of the 22nd International Conference on Data Engineering Workshops,April 3-8, Atlanta, USA, pp. 91.

[38] Barhamgi. M., Benslimane. D., Ghedira. C., Tbahriti. S. and Mrissa. M. (2011). 'A Framework for Building Privacy-Conscious DaaS Service Mashups'. ICWS 2011. Proceeding of the 9th International Conference on Web Services, DC, USA. 2011, p.323-330.