Adaptation of Software Engineering to Semantic Web based System Development

Niyati Baliyan and Sandeep Kumar
Computer Science and Engineering Department
Indian Institute of Technology Roorkee
Roorkee, India
{ niyati.baliyan, sandeepkumargarg } @gmail.com

***Abstract*— Recent web technology transitions have forced the software research community and industry alike, to ponder upon adaptation of various aspects of software engineering, in order to maintain their relevance in present scenario. It is imperative to adapt the software engineering paradigm to technologies such as semantic web for quality software. In this report, various attempts at developing software engineering process for semantic web services have been evaluated. Few works are hybrid in nature, dealing with semantic web and software engineering benefits to each other and the interplay of semantic web and software as a service on cloud. This report presents an analysis of such approaches and highlights some issues that need to be addressed. It is revealed that semantic web services are not just a promising future of the web, but a realization in current web too; therefore, by developing a software engineering process for them, we can develop and deploy quality software (as service) on the web.**

Keywords—*process models, semantic web, software engineering, web engineering*

I. Introduction

After more than a decade of web developments and some huge failures, the advantages of following a process model in creating a timely and high quality result have become clearer. A process model acts as a well-defined roadmap which provides stability, control and organization to an activity which when left uncontrolled, assumes chaotic behavior [1].

The Semantic Web (SW) concept is the brainchild of Tim Berners Lee, who envisioned a web of data or a web in form of a single repository of information instead of a vast collection of websites and web pages [2]. Semantic Web Service (SWS) further enables automated linkage of data across applications, which otherwise is bound to a particular application. Sometimes known as subscription software, the Software-as-a-Service (SaaS) delivery model essentially separates software ownership from the user.

Both SWS and SaaS on cloud are the future of the web, particularly the social web because they provide platform and language independence and on demand service [3]. It is worthwhile to explore the nascent area of software or web engineering process adaptation to semantic web, since there is paucity of literature in this area and one can benefit from adapting software engineering paradigm to these technology transitions well in time. It is intended to contribute a comprehensive review in this field to aid researchers by providing a body of knowledge as starting point, because from own experience, this search can be tedious and time consuming. Before delving into semantic web from software engineering perspective, the underlying concepts in semantic web have been presented in brief.

Rest of the paper is organized as follows. Apart from Introduction to the paper in section I, section II gives background information about semantic web, section III outlines the review method used in this report and summarizes the major findings from analysis of selected literature. The observations taken from the rigorous literature review have been enumerated in Section IV. This section also explains various identified research gaps. Section V presents the scope of this study. Section VI concludes the paper.

II. Background

Before the web era, various software development methodologies have been proposed for the development of software applications aimed at meeting user's requirements and reducing the maintenance cost of the developed software. Currently available software development methodologies are inappropriate to use for the development of web-based software systems, especially for the third generation web, called SW, because there is a need of machine understandable semantics of web contents in this type of software systems.

However, SW does not look to replace the current web; hence SWS are not a replacement of current software or Web Service (WS). They are rather an extension of current software. It is therefore, required to adapt the current Software Engineering (SE) process so that it can be applied to SW. This will provide standardization and ease to the way information is found, reused and combined. SW is only in its nascent stage and shall mature in future, coupled with the vision of web operating system probably being realized.

Adaptation of SE process to SW calls for changes in each lifecycle phase of traditional SE and well as in

umbrella activities which are critical to quality such as change management, risk management, configuration management etc. [4].

III. Literature Review and Findings

The systematic literature review protocol used in this report is defined in Table I.

TABLE I
SYSTEMATIC REVIEW PROTOCOL

Objectives	To identify the current state of progress made at adaptation of software engineering to semantic web based system development.
Main Question	What are the trends and future directions in different aspects of software engineering for the development of semantic web enabled software, particularly how is the software engineering process affected?
Intervention	Case study, Prototype description, definitions and other research related to semantic web services composition.
Keywords	semantic web; ontology; software engineering; web engineering; software process model
Source selection criteria	Availability of the papers over the web and search engines, using keywords
Methods	Automated search; Search through bibliographic references of selected papers; Search by authors; Search for related conferences and journals
Source Engines	IEEE, Springer, ScienceDirect, ACM, Google Academic Search, Microsoft Academic search
Inclusion criteria	Availability of full text; The study involved definitions, experience reports, handled open research issues in the area of semantic web from software engineering perspective.
Exclusion criteria	Non availability of full text; The study was not directly related to semantic web; The study handled semantic web without considering the software engineering perspective.
Initial studies selection	On the basis of title, abstract, keywords
Studies quality evaluation	Through PROMPT criteria [5]
Results summarization	Filtered studies were analysed and the main idea and findings were summarized in a single textual report

The main issues addressed, contributions and open issues in the 25 selected studies are summarized in Table II. On the basis of commonality in objective, the reviewed works can broadly be grouped into four clusters. Studies [6]-[11] deal with reengineering approaches for semantic enabled software while [12]-[16] discuss development from scratch for semantic web enabled software. The studies [17]-[21] highlight semantic web and software engineering benefits for each other, while [22]-[30] are hybrid in nature and deal with semantic web on the cloud such as application of ontology to SaaS on the cloud.

TABLE II
SUMMARY OF LITERATURE REVIEWED

Reengineering approaches for developing Semantic enabled Software			
Available Work	**Objective**	**Contribution**	**Limitation**
[6]	Ontology based reverse engineering of web application for easy evolution	Reverse engineering approach evaluated through case study	Web services have not been reverse engineered
[7]	Reengineering web applications into SWS for reuse	Reduced designer intervention observed on validating reengineering framework	The tool can be refined to fit complex industrial cases.
[8]	Obtaining specification of SWS using Model Driven Development (MDD) based reverse engineering	Semantic web developers freed of writing low level codes for SWS specification	Automated annotation of specification with Web Service Description Language (WSDL) missing
[9]	Reverse engineering web service on Web Service Modeling Ontology (WSMO), for more dynamic web	Proposed approach validated on Amazon web service	WSDL file partially explored; Universal Description Discovery Integration (UDDI) and Simple Object Access Protocol (SOAP) can be explored
[10]	Transitioning legacy web applications to SWS for industry acceptance	Semi-automatic method applied on Z models	Tool support can be developed for verification
[11]	Reverse engineering WSDL for WSMO specification	Development of semi-automated tool WSDL2WSMO	Automated specification of WSMO by including SOAP and UDDI information
Development from scratch of Semantic enabled Software			
Available work	**Objective**	**Contribution**	**Limitation**
[12]	Framework for developing SWS with efficient design	Reduced cost of semantic annotations	Design of WSMO concepts can be improved
[13]	Requirement engineering of semantic web software	A process model for semantic web software	Validation of model needed before using it can be used as a benchmark
[14]	Semantic web engineering lifecycle model	Comparative analysis of proposed model with existing works like WEESA and WSDM	Implementation of model and validation for efficiency can be done
[15]	Semantic web development based on	Up-to-date web engineering methodology	Web modeling language not completely

	software engineering and semantic methods	extended for semantic web	implemented for SWS
[16]	Process framework for semantic aware software	Results in more productive software, which evolves proactively	Industry level case studies needed to further validate the framework
Semantic Web and Software Engineering: Mutual Benefits			
Available work	**Objective**	**Contribution**	**Limitation**
[17]	Introducing ontologies in software at build time for cost reduction	First ever architecture of semantic enabled IDE	Potential of ontology in software engineering can be explored further
[18]	Incorporation of the ontology paradigm into a general-purpose programming language	Simple and agile approach that brings the most important features of the ontology into the software engineering realm	Toolset and languages not mature; validation performed only on small scale
[19]	Model-Assisted Software Development: a variation on the MDD	Defining semantic representation in software development phases can help provide more automation within and across phases	Need for sophisticated tooling that leverages domain and phase-specific ontologies
[20]	Ontological applications throughout software engineering life cycle captured in a framework	Exploiting benefits for both knowledge engineering and software engineering	The extent of domain ontology leverage can be checked on real software
[21]	Ontology development from matured software engineering methodology	Extends UML modeling to ontology development in context of MDD and Web ontology Language (OWL)	Yet to develop proposed domain ontology and evaluate its efficiency
Relationship between Semantic Web and Software as a Service			
Available work	**Objective**	**Contribution**	**Limitation**
[22]	Model for federations of services with specification of semantics	Through this model, users may search , select, compose services optimally	Appropriate validation is missing
[23]	Systematic review of cloud computing ontologies, their applications and focuses.	Presented open issues for using ontologies to improve security and interoperability of cloud computing	A limited number of studies is chosen over limited period.
[24]	An ontology cloud model and its operation is presented	The model is foundation of building information, knowledge and intelligent system.	Practical demonstration of proposed definitions is missing
[25]	Ontology based composition and matching for dynamic services	The matching solution is more flexible and scalable	Nonfunctional aspects such as fault tolerance and trust needs to be investigated
[26]	Semantic Search Engine(SSE), demonstrates how semantic computing can benefit cloud computing	SSE can bring advantages to cloud computing to grow more quickly	The architecture can be refined further and demonstrated through some real world application.
[27]	Semantic based access control in cloud computing scenario	Easily solves the problem of access in cloud computing environment.	Further support possible with a real world application implementing the proposition
[28]	Detailed ontology for the cloud	Establishment of knowledge domain in the area of cloud computing	The ontology needs to be objectively validated for universal acceptance followed by its practical use.
[29]	Multi agent-based Cloud service discovery protocol through ontology based matching	Better utility, success rate and flexibility than that without cloud ontology	Evaluation not comprehensive enough since it is based on price and timing factors
[30]	Semantically enabled approach to the integration of service families in the cloud	Demonstrated how ontologies and semantic web technologies can be employed to automatically identify correspondences between business processes and features.	Tool support is not complete. Real case studies can be applied to validate the approach further.

Most studies proposed were semi or fully automated processes i.e. utilized tools for their framework implementation or for validation of their framework such as WSMO validator plug in [17]. Overall, the area of interest is of recent significance as majority of the studies were spread over five years. The primary need for adaptation have been identified as machine understandable semantic web content. None of the proposed semantic web

engineering model was fully implemented and most of the aforementioned works propose a reengineering approach for transforming web applications into semantic web applications and only a few handled the case of development from scratch. It has been observed that some researchers have adapted MDD since ontology is a natural choice for modeling.

After going through proposed approaches for SE model for SW, it was observed that since there is little dependency amongst the phases of SW development, they can be sequenced as required. This flexibility makes possible to shape the development process to the need of the type of service. However, the validation rules shall guarantee that whatever lifecycle cycle is implemented can be checked for completeness and consistency of design.

IV. Observations and Research Gaps

There is a need for implementation and development tool support especially for early web engineering activities. SE framework needs to be developed for SW with empirical validation using case study or open source industry projects.

SE for SW is not fully automated. There are no standards defined for reengineering of WS to SWS. SW technology can be incorporated in SE through Model Driven Architecture (MDA); ontologies being vocabulary of concepts are natural choice for modeling. The focus of MDA is on forward engineering, i.e. producing code from abstract modeling diagrams (e.g. class diagrams).

There is a need to identify SE process phases where understanding of semantic content by machine can be incorporated i.e. can it run in parallel with usual software process or is there a necessary sequence. Similarly, semantic annotations need to be added at some stage in life cycle.

Majority of literature is focused on reengineering as the SW trend is currently catching up; therefore existing apps are transformed to exploit their benefits. However, the future web shall see dominance of SW and SaaS and hence they must be incorporated in the beginning phases of engineering life cycle. This strengthens our case in favor of essential and urgent custom SE lifecycle models for SW.

Although there have been preliminary attempts to address the specific requirement of SW software and service i.e. 'machine understandable semantics of web content', none of the approaches is mature enough or holistic enough to be called software engineering framework or model. That is, it describes one or more aspect of development but not the process in entirety. This calls for an approach based on rudiments of SE yet flexible enough to accommodate changes coupled with SWS and cloud services.

Today, there are many frameworks built around SWS description approaches, yet a common SE framework or SE process for SWS is missing. It is an open issue to supply a standard SE process for SW enabled software. This shall further an academic understanding of the benefits and tradeoffs of standard process versus adhoc development.

SW provides formal models to SE that enables reasoning about resources through the flagship language OWL, SW can support SE lifecycle phases like analysis (ontology based documentation), maintenance(using knowledge based system) etc. in numerous ways [31]. Similarly SE contributes architecture, tool support and visual modeling techniques to SW through MDA with UML being the flagship language for the purpose. Thus, the relationship between SE and SW is symbiotic where both domains affect each other in an advantageous manner [32]. Besides analyzing such effects, it is intended to direct the research towards how the SE discipline in terms of SE process (model/framework) adapts to SW and SWS.

V. Scope

The need to adapt current software engineering to recent technology transitions which form a part of web engineering has been highlighted [33]. The purpose of our study is twofold. At one hand, it is aimed to explore how the business model called cloud computing needs more standardized process for delivery of SaaS. On the other hand, it may also be explored how the SW can benefit from a SE process particularly adapted to suit SWS. The effect of SW on the overall web engineering process in general and also on individual phases of the process, in particular will be studied.

By adapting SE to SW for data intensive applications as in Health care etc., the process of linkage of data for automated processing is standardized. By using SE process, semantic web applications can be enriched with SE architecture and tool support through MDA with UML Similarly, cloud software engineering is fit for applications with reusable components (provided as services) and developing a benchmark SE process for the same would enhance its benefits further. We argue that the disparities between SE methodologies reflect the fact that different practices are required to tackle the different challenges in a large array of software projects. The failure of a software project is not caused by a software methodology but by the development team that selected an inappropriate development methodology.

Basili emphasized that we need models that help us reason about the suitability of SE practices and methodologies for a specific software project [34]. Therefore, one must develop through machine learning a model that helps one decide about the suitability of a model to a project.

VI. Conclusion

Traditional SE is no more the same because the very nature of software is evolving; SE needs adaptation to SWS and SaaS on the cloud. The prime concern of this study is to analyze the SE discipline in terms of its coincidence with SW, such that maximum benefits can be exploited from both these fields and quality software can result. The future shall see mainstreaming of SW technology and, hence it is imperative to develop framework for the process

behind development of software on these platforms. It is not recommended that software development methodologies be simply integrated with service deployment, nor is it suggested to draw a straight forward linkage between software development and service deployment models. The two domains must be integrated in a standardized yet flexible manner.

In no way, do we believe the results of our findings are the only and definitive list. Feedback and discussion from the community is welcome to make the list more comprehensive.

REFERENCES

[1] Pressman, R. S., & Ince, D. *Software engineering: a practitioner's approach*, Ed. 5. New York: McGraw-hill. 1992.

[2] Floridi, L."Web 2.0 vs. the Semantic Web: a philosophical assessment." *Episteme*. 2009.

[3] Turner, M., Budgen, D., & Brereton, P. "Turning software into a service". *Computer.*, 36(10), 38-44. 2003.

[4] Yajing Zhao; Jing Dong; Tu Peng;, "Ontology Classification for Semantic-Web-Based Software Engineering," *IEEE Transactions on Services Computing*, vol.2, no.4,303-317. 2009.

[5] http://www.open.ac.uk/infoskills-researchers/resources/S5_PROMPT_Checklist.pdf

[6] Djelloul Bouchiha , Mimoun Malki , Sidi Mohamed Benslimane "Ontology Based Web Application Reverse-Engineering Approach" in *INFOCOMP (Journal of Computer Science)* 6(1):37-46.2007.

[7] Bouchiha, D.; Malki, M.; "Towards re-engineering Web applications into Semantic Web Services", International *Conference on Machine and Web Intelligence (ICMWI),* vol.3, no.5, pp.350-353.2010.

[8] D. Amar Bensaber, D. Benslimane, M. Malki. "A Reverse Engineering Approach for Specifying Semantic Web Service with Respect to MDA." *International Review on computers and software*, Vol 2 No.6 pp: 581-593, Praise Worthy Prize, ISSN 1828-6003. 2007.

[9] El Bouhissi, H., & Malki, M. : "Reverse Engineering Existing Web Service Applications.", *16th Working Conference on Reverse Engineering, 2009. WCRE'09* (pp. 279-283). IEEE. 2009.

[10] Wang, H. H., Gibbins, N., Payne, T., Saleh, A. and Li, Y. "Transitioning Applications to Semantic Web Services: An Automated Formal Approach". *International Journal of Interoperability in Business Information Systems (IBIS).* ISSN 1862-6378.2008.

[11] Houda El Bouhissi et al."A reverse engineering approach for the web service Modeling Ontology specifications" *The Second International Conference on Sensor Technologies and Applications IEEE.* pp.819-82,2008.

[12] M. Brambilla, I. Celino, S.Ceri, D. Cerizza, E.Della Valle, F. Facca. "A Software Engineering Approach to Design and Development of Semantic Web Service Applications", *International Semantic Web Conference (ISWC2006)*,Athens, USA, Springer LNCS 4273, pp.172-186. 2006.

[13] Farooq, A., & Arshad, M. J. A Process Model for Developing Semantic Web Systems, *3*(9), 34–39. 2010.

[14] Kumar, S., Sangwan, S., Kumar, K., & Jain, A. "A Semantic Web Engineering Model." in *International Conference on Intelligent Computational Systems* (ICICS) 2011

[15] Brambilla, M., Ceri, S., Valle, E. Della, Facca, F. M., & Tziviskou, C. Book chapter "A Software Engineering Approach based on WebML and BPMN to the Mediation Scenario of the SWS Challenge".2009.

[16] Olawande J. Daramola "A Process Framework for Semantics-Aware Tourism Information Systems" *ICWE Springer* pp. 521–53.2010.

[17] Bernhard Bauer, Stephan Roser, "Semantic-enabled Software Engineering and Development", *INFORMATIK 2006 - Lecture Notes in Informatics (LNI)*, P-94:293–296.2006.

[18] Djuric, Dragan, and Vladan Devedzic. "Incorporating the ontology paradigm into software engineering: Enhancing domain-driven programming in clojure/java." *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on* 13-14. 2012.

[19] Verma, Kunal, and Alex Kass. "Model-Assisted Software Development: Using a'semantic bus' to automate steps in the software development process." *Semantic Web* 1, no. 1:17-24. 2010.

[20] A. Anandaraj, G. Deepak and K. Raja . "Study of Ontology in Software Modelling Process and Life Cycle" in *International Journal of Research and Reviews in Software Engineering.* Vol 1 ,No. 1 ,pp.13-17.2011.

[21] John, S. "Leveraging Traditional Software Engineering Tools to Ontology Engineering under a New Methodology." In *Future Information Technology (FutureTech), 2010 5th International Conference on*, pp. 1-5. IEEE, 2010.

[22] Ma, Hui, K-D. Schewe, and Qing Wang. "An abstract model for service provision, search and composition." In Services Computing Conference, 2009. APSCC 2009. IEEE Asia-Pacific, pp. 95-102. IEEE, 2009.

[23] Androcec, Darko, Neven Vrcek, and Jurica Seva. "Cloud Computing Ontologies: A Systematic Review." In *MOPAS 2012, The Third International Conference on Models and Ontology-based Design of Protocols, Architectures and Services*, pp. 9-14. 2012.

[24] Zhu, Jiang, and Wenhua Wang. "New-knowledge-view based ontology cloud model." In *Computer Science and Software Engineering, 2008 International Conference on*, vol. 5, pp. 1140-1143. IEEE, 2008.

[25] Pahl, Claus, Veronica Gacitua–Decar, MingXue Wang, and Kosala Yapa Bandara. "Ontology–based composition and matching for dynamic cloud service coordination." *International Journal of Metadata, Semantics and Ontologies* 6, no. 3: 195-206.2011.

[26] Sheu, PC-Y., Shu Wang, Qi Wang, Ke Hao, and Ray Paul. "Semantic computing, Cloud computing, and semantic search engine." In *Semantic Computing, 2009. ICSC'09. IEEE International Conference on*, pp. 654-657. IEEE, 2009.

[27] Hu, Luokai, Shi Ying, Xiangyang Jia, and Kai Zhao. "Towards an approach of semantic access control for cloud computing." In *Cloud Computing*, pp. 145-156. Springer Berlin Heidelberg, 2009.

[28] Youseff, Lamia, Maria Butrico, and Dilma Da Silva. "Toward a unified ontology of cloud computing." In *Grid Computing Environments Workshop, 2008. GCE'08*, pp. 1-10. IEEE, 2008.

[29] Kang, Jaeyong, and Kwang Mong Sim. "Towards agents and ontology for cloud service discovery." In *Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), 2011 International Conference on*, pp. 483-490. IEEE, 2011.

[30] Boškovic, Marko, Ebrahim Bagheri, Georg Grossmann, Dragan Gaševic, and Markus Stumptner. "Towards Integration of Semantically Enabled Service Families in the Cloud." *WS2*: 58.2011.

[31] HWood, D., Carrington, D., & Kaplan, S. "Toward a software maintenance methodology using semantic web techniques." in *Second International IEEE Workshop on Software Evolvability, SE'06.* (pp. 23-30). 2006.

[32] Happel, H. J., & Seedorf, S.). "Applications of ontologies in software engineering" In *Proc. of Workshop on Sematic Web Enabled Software Engineering (SWESE)* (pp. 5-9). 2006.

[33] Kumar, S., S Sangwan, S. Adapting the Software Engineering Process to Web Engineering Process. *International Journal of Computing and Business Research*, *2*(1). 2011.

[34] V.R. Basili, "The role of experimentation in software engineering: past, current, and future", 18th International Conference on SE, Berlin, Germany, pp: 442 – 449, 1996.