Towards semantic-aware and ontology-based e-government service integration – an applicative case study of Saudi Arabia's King Abdullah Scholarship Program

Decision System & e-service Intelligence (DeSl) Lab

Center for Quantum Computation and Intelligent Systems (QCIS)

Faculty of Engineering and Information Technology

University of Technology Sydney, Australia

Abdullah Alqahtani, Haiyan Lu, Jie Lu

Aalqahta@it.uts.edu.au Haiyan.Lu@it.uts.edu.au Jie.Lu@it.uts.edu.au

Abstract By improving the quality of e-government services by enabling access to services across different government agencies through one portal, services integration plays a key role in e-government development. This paper proposes a conceptual framework of ontology based e-government service integration, using Saudi Arabia's King Abdullah Scholarship Program (SAKASP) as a case study. SAKASP is a multi-domain program in which students must collect information from various Ministries to complete applications and the administering authority must verify the information supplied by the Ministries. The current implementation of SAKASP is clumsy because it is a mixture of online submission and manual collection and verification of information; its time-consuming and tedious procedures are inconvenient for the applicants and inefficient for the administrators. The proposed framework provides an integrated service by employing semantic web service (SWS) and ontology, improving the current implementation of SAKASP by automatically collecting and processing the related information for a given application. The article includes a typical scenario that demonstrates the workflow of the framework. This framework is applicable to other multi-domain e-government services.

1. Introduction and Problem

The rapid development and uptake of Information and Communications Technology (ICT) is increasing the efficiency of e-government services. E-government uses the World Wide Web and the Internet to improve access to and delivery of government information and service to citizens, business partners, employees and other agencies (McClure 2000), and open opportunities for service integration which can improve e-government service significantly. Unfortunately, relatively few of the many potential applications of service integration have actually been

implemented in government (Aldrich, Bertot & McClure 2002); most e-government service systems are providing standalone services rather than integrated services.

In order to provide an integrated service, an e-government system should be able to automatically discover, select, compose and execute correct service components across multiple domains for heterogeneous users. Semantic web services (SWS) can be used to achieve this goal. The basic standards are rendered by web services infrastructure, which allows interaction of web services and consists of three phases — publish, find and bind, and three entities — the service provider, the registry and the service requester (Hendler, Berners-Lee & Miller 2002).

Lack of collaboration and integration between agencies and inadequate use of ICT can create great inconvenience for users. For example, in Saudi Arabia's King Abdullah Scholarship Program (SAKASP — a program which provides financial support to eligible Saudi Arabian postgraduate students), online submission is possible but collection and verification of related information is conducted manually. SAKASP involves two players — the student applicant and the Commission of the Ministry of Higher Education (the administering authority). Fig.1 gives an overview of the workflow of SAKASP.

Fig1. Workflow of SAKASP.

The detailed processes/sub-processes for the applicant and the Commission are illustrated in Fig.2 and Fig.3.

Fig2. Workflow seen by the applicants.

Fig3. Work flow seen by the commission.

The SAKASP application procedure is as follows. The student must:

Step1: register with SAKASP online through the Ministry of Higher Education portal;
Step2: obtain written certification from the Ministry of Civil Services to prove that he or she is not employed;
Step3: obtain an up-to-date academic transcript from Ministry of Education, and
Step4: meet a staff member of the Commission of the Ministry of Higher Education to check the paperwork.

The Commission must:

Step5: manually check that the applicant satisfies the SAKASP criteria (in compliance with privacy rules relating to sensitive information, which adds a considerable administrative burden).

The procedure outlined above shows that the applicant must communicate with and/or visit three Ministries to apply for a scholarship and the Commission needs to manually process the application. This set of requirements causes the current implementation of SAKASP to be time-consuming, inconvenient for the applicants, and inefficient.

This paper proposes a conceptual framework of semantic-aware and ontology-based e-government service integration, using SWS and ontology that represents a better solution for SAKASP.

The rest of this paper is organized as follows: Section 2 presents the related work; Section 3 presents the conceptual framework of semantic-aware and ontology-based e-government service integration using SWS and ontologies; Section 4 presents feasibility analysis. Finally, Section 5 concludes the paper and points out some future work.

2. Related Work

Some research effort has been devoted to the exploitation of SWS to improve e-government services. Gugliotta et al. (2005) proposed a semantically-enhanced architecture to address the issues of interoperability and service integration in e-government. This work uses SWS as the basis of the semantically-enhanced middleware of a public portal. Other researchers used SWS to develop techniques for efficient integration of e-government services across multiple domains but heterogeneous problem cannot solve by this work (Liuming et al. 2004; Medjahed et al. 2003). In our paper, we use SWS infrastructure to automatic SAKASP service that allows discovery, composition and invocation of multiple, heterogeneous web services.

On the other hand, a specific ontology can be seen as an explicit specification of an abstract, simplified view of a world we desire to represent (Gruber 1995). According to Zilli et al. (2009), ontologies are schemas of the world in which every item (concept, relationship, attribute) is described using a natural language vocabulary, and their power is demonstrated when applications are integrated in a technical platform and when intensive search for data, information and knowledge on the Web is needed. An ontology has four components: concepts, relations, in-

stances and axioms (Tang et al. 2006). Ontologies accord with the vocabulary usage pertaining to a specific region with a consistent approach. In other words, ontologies are devices for formalising knowledge and encryption of higher-level data models incorporating life events, services and procedures (Asunción 2003). Many previous researchers have proposed the use of ontologies to improve e-government services. For example, Gugliotta et al. (2005) used ontology for describing life events, services and e-government knowledge in middleware of a public portal. Ontology evolution has been used to improve the management of e-government changes (Ljiljana et al. 2004), and Goudos et al. (2007) produced a generic government domain ontology by defining a formal model for a Public Administration service on the basis of web service modeling ontology (WSMO).

4. Architectural Solution

An architecture of an integrated e-government service using SWS and ontology consists of three layers, as shown in Fig.4.

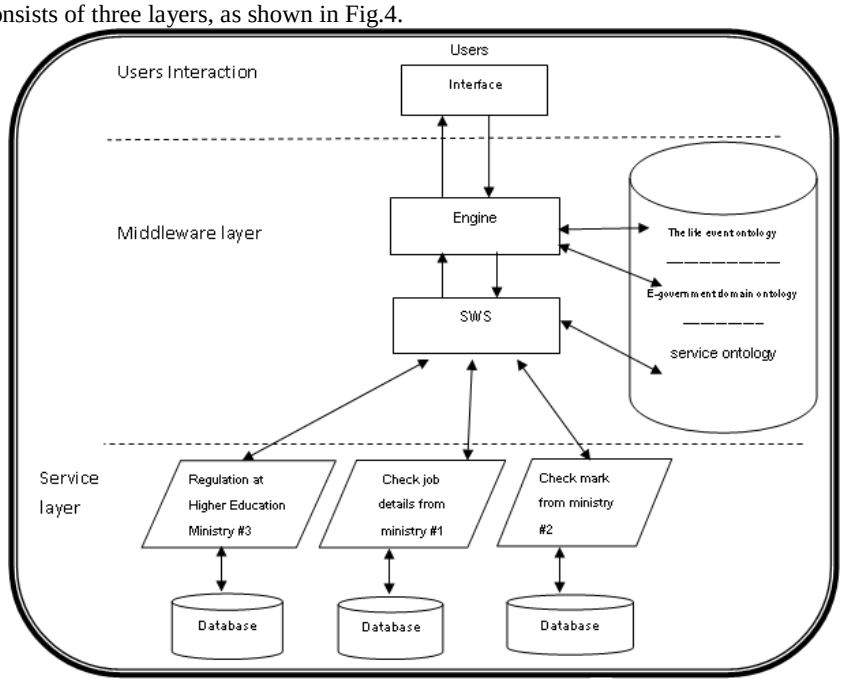

Fig4. An architecture of integrated e-government service (E-SAKASP) using SWS and ontologies

The first layer is the user layer, which includes an interface component. The interface is used to present the services, their descriptions, and the results of applied service and to collect information to run service from applicant such as his/her

Public key Infrastructure**.** The applicant can select any available service through the interface by clicking on it, and enters information through the interface when he or she needs to use the service.

The second layer of the architecture is middleware, which includes the engine, the government domain ontology, the life-event ontology, service ontology, and SWS. The engine manages requests made by the applicant through the interface and discovers all events related to the requested service from life-event ontology. For example, as shown in Fig.5, if the applicant selects E-SAKASP the engine presents all related events (such as 'issue financial guarantee' and 'delay my scholarship') from the life-event ontology. In addition, the engine retrieves the semantic description of goal which is described via service ontology such as the semantic description of mark.

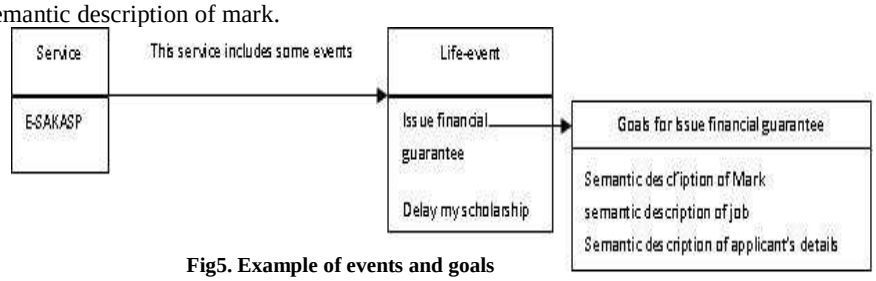

Fig5. Example of events and goals

The e-government domain ontology describes information related to business, end-user, legal and economic concepts (Gugliotta et al. 2005). The case study presented in this article uses e-government domain ontology to describe the main concepts related to an application to E-SAKASP. Fig.6 shows e-government domain ontology but does not cover all aspects connected with e-government.

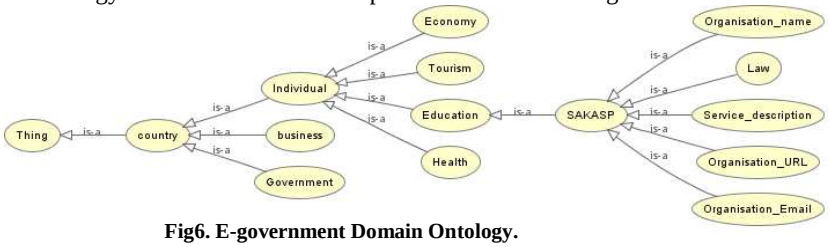

Fig6. E-government Domain Ontology.

Life-event ontology is the basis of services integration (Sanati & Jie 2008). Castellano et al. (2005) described a life event as a citizen's basic service request (such as applying for a scholarship or to transfer between schools). Therefore, as shown in Fig.7, we suggest life event ontology to describe event. Life event ontology includes many classes: applicant, goals, rules (laws), parameter and result. 'Applicant' is used to enter applicant's details such as applicant Public Key Infrastructure. 'Goals' include objectives which are used to achieve events across web services such as 'check mark'. The goal is retrieved from service ontology. 'Rules (laws)' are retrieved from e-government domain ontology such as the mark up %85. 'Parameter' is useful to store data retrieved from web services. 'Result' is used to present the final result to applicant (such as 'issue financial guarantee'). In

conclusion, life-event is used as a model for the user's point of view. This means that a life-event allows users to identify their particular situations and better describe what they want to achieve (Gugliotta et al. 2005).

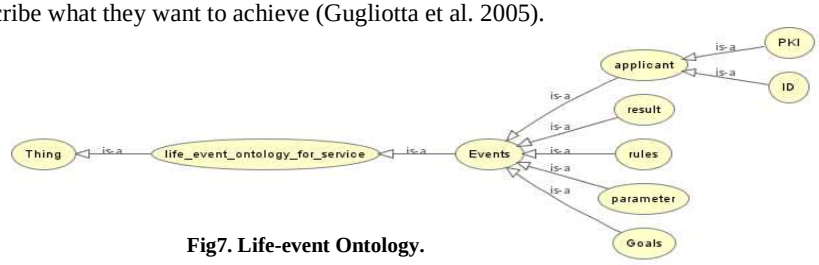

Fig7. Life-event Ontology.

Fig.8 shows the service ontology, which is WSMO. According to Roman et al. (2005), WSMO is ontology for describing elements of service that allow the automation of web services discovery, composition, mediation and invocation. WSMO includes mediators, goals, web service and ontologies. Mapping among components is defined by the mediators. The objectives that are to be attained by users through the web services are represented by goals. The descriptions of WS incorporate explanation of communication of web services (choreography) and composition of web services (orchestration). The formally specified terminology of information is offered by ontologies and used by other elements.

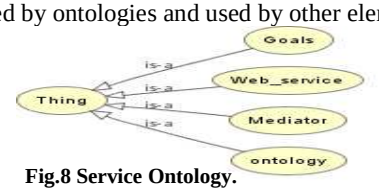

Fig.8 Service Ontology.

SWS is the last component of middleware. In general, OWL-S and WSMO are the approaches most commonly used to define SWS. According to the results of a comparison between OWL-S and WSMO made by Lara et al. (2004), the WSMO is more useful because WSMO provides a conceptual model of core elements of SWS which allow the discovery, composition and execution of multiple heterogeneous web services. Therefore, in this article WSMO is used to describe SWS.

The final layer of the architecture is the service layer, which includes web services. Three web services are offered by three Saudi agencies (Ministries) relevant to SAKASP and provide all the information necessary to achieve E-SAKASP.

5. Feasibility Analysis:

To demonstrate the feasibility of the proposed E-SAKASP system, assume a student wants to apply for SAKASP. He needs to use an interface which offers all related services (such as 'transfer between schools' and 'apply to E-SAKASP'). As shown in Fig.9, he selects the E-SAKSP service to lodge his application (1). The engine (E-SAKASP) manages his request and discovers all events which are related to his application such as 'issue financial guarantee' and 'delay my schol-

arship' from the life event ontology (2). Then, the system presents those events to the student through the interface (3). In the next phase, he selects one event which he wants to achieve — for example, 'issue financial guarantee' (4). After he selects 'issue financial guarantee' to execute, the workflow of the event runs. As shown in Fig.10, the engine retrieves all goals which are related to 'issue financial guarantee' (5); those goals are description semantic of check mark, description semantic of check job and description semantic of check details, and include objectives which are used to achieve 'issue financial guarantee' across web services. Then, the system asks the applicant to enter his PKI (6). The engine connects with SWS to discover the student's most recent mark via the Ministry (WS) which is responsible for student assessment (7) & (8) and stores his mark in parameter class (9). After that, the system repeats until all goals (check his job and check his details) are achieved (10). The system checks parameter against rules which are retrieved from e-government domain ontology such as mark up %85 (11). Finally, the system presents the result (Yes/No) via the interface from result class (12) & (13).

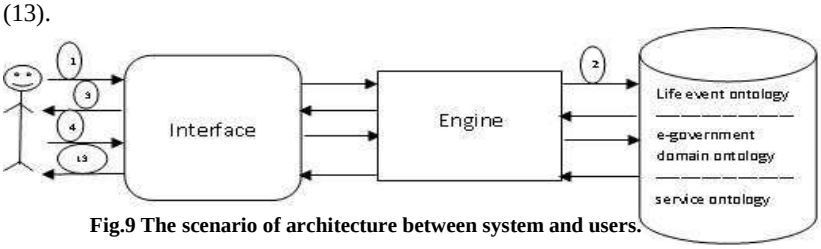

Fig.9 The scenario of architecture between system and users.

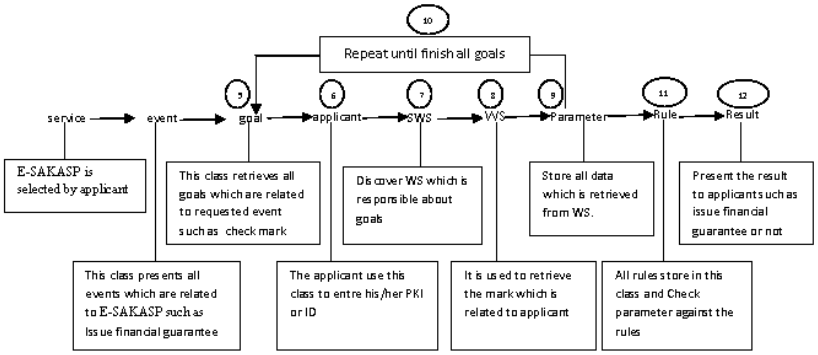

Fig.10 The workflow of the system.

6. Conclusions and Future Work

In our study we showed how SWS and ontology could be used in architecture of an integrated e-government service to improve the existing SAKASP application system, saving time and money for applicants and administrators. An additional benefit of this solution is increased data security, as the service is executed according to the applicant's identification key (ID) or PKI.

Our future work will concentrate on further evolution and development of e-government service integration. Our focus will be on optimising protocol communication between classes (concepts) and using ontology to radically improve the security of e-government service integration.

References

Aldrich D, Bertot JC, & McClure CR, (2002) E-Government: initiatives, developments, and issues. *Government Information Quarterly* 19(4):349-355.

Asunción G, Mariano F, and Oscar C (2003) *Ontological engineering : with examples from the areas of knowledge management, e-commerce and the semantic Web*. Springer-Verlag, London.

Castellano M, Pastore N, Arcieri F, Summo V, de Grecis GB (2005) An E-Government Cooperative Framework for Government Agencies. *2005. HICSS '05. Proceedings of the 38th Annual Hawaii International Conference on System Sciences*: 121c-121c.

Goudos SK, Loutas N, Peristeras V, Tarabanis K (2007) Public Administration Domain Ontology for a Semantic Web Services EGovernment Framework., *2007. SCC 2007. IEEE International Conference on Services Computing*: 270-277.

Gruber TR, (1993) A translation approach to portable ontology specifications. *Knowl. Acquis.* 5(2):199-220.

Gugliotta A, Cabral L, Domingue J, Roberto V (2005) *A Conceptual Model for Semantically-based E-Government Portals*. International Conference on e-Government, Canada: 101-112.

Gugliotta A, Cabral L, Domingue J, Roberto V, Rowlatt M, Davies R (2005) Semantic Web Service-based Architecture for the Interoperability of E-government Services. *In Proceeding of Web Information Systems Modeling Workshop (WISM 2005) in conjunction with The 5th International Conference on Web Engineering (ICWE 2005)*, Sydney, Australia.

Hendler J, Berners-Lee T, & Miller E. (2002) Integrating Applications on the Semantic Web. *Journal of the Institute of Electrical Engineers of Japan* 122(10):676-680.

Lara R, Roman D, Polleres A, Fensel D (2004) A Conceptual Comparison of WSMO and OWL-S. *Proceedings of The European Conference on Web Services, Erfurt, Germany, Sept 27-30*: 254–269.

Liuming L, Guojin Z, Jiaxun C (2004) An infrastructure for e-government based on semantic Web services. *IEEE International Conference on Services Computing*, IEEE:483-486.

Ljiljana S, Andreas A, Nenad S, Rudi S (2004) On Managing Changes in the Ontology-Based E-government. In: *On the Move to Meaningful Internet Systems 2004: CoopIS, DOA, and ODBASE*, vol. 3291/2004, Springer Berlin / Heidelberg: 1080-1097.

McClure DL (2000) Electronic government: Federal initiatives are evolving rapidly but they face significant challenges, Washington

Medjahed B, Bouguettaya A, Ouzzani M (2003) Semantic web enabled e-government services. paper presented to the *Proceedings of the 2003 annual national conference on Digital government research*, Boston, Digital Government Society of North America.

Medjahed B, Rezgui A, Bouguettaya A et al. (2003) Infrastructure for e-government Web services. IEEE Internet Computing.7(1):58-65

Roman D, Lausen H, Keller U et al. (2005) D2v1.2. *Web Service Modeling Ontology (WSMO)*. http://www.wsmo.org/TR/d2/v1.1. Accessed 1 January 2010.

Sanati F, Jie L (2008) Semantic Web for e-government service delivery integration. *5th International Conference on Information Technology: New Generation*, IEEE: 459-464.

Tang J, Li J, Liang B, Huang X, Li Y & Wang K (2006) Using Bayesian decision for ontology mapping. Web Semant 4(4):243-262.