# iSemServ: Towards the Engineering of Intelligent Semantic-Based Services

Jabu Mtsweni[1,2], Elmarie Biermann[2], and Laurette Pretorius[2]

[1] SAP Research Center Pretoria/SAP Meraka UTD, Faerie Glen, Pretoria, South Africa
jabu.mtsweni@sap.com
[2] University of South Africa, Pretoria, South Africa
pretol@unisa.ac.za; bierman@xsinet.co.za

**Abstract.** The emergence of Semantic Web Services is stimulating the need for modern enterprises to efficiently and rapidly develop and deliver machine-processable and machine-interpretable value-added services in order to automate a variety of tasks on the Web. However, semantic-based services are scarcely adopted and utilised as there are few real-life examples that demonstrate the possibilities and benefits of such services. Furthermore, there is a lack of service creation frameworks and technical platforms that purport to guide and promote simple, flexible, rapid, and unified engineering of semantic-based services. In addition, current semantic service platforms do not support the construction of semantic services that are intelligent beyond the application of ontologies. In this position paper, preliminary efforts that seek to address the challenges of simplifying and speeding-up the engineering process of intelligent semantic services are presented. The goal of the work presented in this paper is about providing service providers, designers, and consumers with simple, unified, and yet simple tools that can aid in the technical implementation of intelligent semantic-based services. The main contributions envisioned from this research is a conceptual service creation framework called *i*SemServ and a technological service creation platform, which is intended to simplify and support the phases of building intelligent semantic services in an integrated manner. The proposed research adopts a quantitative approach with the main focus on model-building, prototypes, laboratory experiments, and computer-based simulations.

**Keywords:** Intelligent Semantic-based Services, Service Creation Framework and Platform, Ontology, Intelligent Agents.

## 1 Introduction

The need to expeditiously deliver value-added services on the Web is becoming a business imperative. From the current research literature, it is clear that Web Services (WS) have transformed the World Wide Web (WWW) from being a repository of data and information to a platform of distributed services [1] . However, WS only focus on the syntactical description of services; making it a challenge for services to be automatically discovered, selected, composed, invoked, and executed without or with little human intervention. Consequently, Semantic Web Services (SWS) are emerging to address such challenges [2].

F. Daniel and F.M. Facca (Eds.): ICWE 2010 Workshops, LNCS 6385, pp. 550–559, 2010.

© Springer-Verlag Berlin Heidelberg 2010

In general, SWS can be defined as the convergence of Semantic Web (SW) and WS concepts in order to enable machine-processable and machine-interpretable services. SWS strives for WS that cannot only be processed by humans but by machines or software programs as well [3]. This is accomplished by leveraging WS with semantic annotations derived from domain and service ontologies to facilitate and enable automation of various aspects of WS such as discovery, selection, invocation, and execution [4, 5]. Ontologies as core to SWS refers to the conceptual knowledge representation that can be used to annotate services [6].

Although the SWS domain promises services that are machine interpretable and processable, this domain still suffers from a number of challenges. Studer [6] notes that "SWS are still searching for their 'killer' application". According to Agre *et al.* [3], SWS and related applications are scarcely adopted and utilised as there are few pilot applications that fully demonstrate the possibilities and benefits of semantic-based services, and related applications. This might be further attributed to the fact that currently a combination of disconnected software tools are used to design and develop such services. This can however lead to undesirable consequences such as long service development time, high development costs, lack of services reuse, and even lack of semantics reliability and reusability.

It should be noted though that there are various software tools (e.g. WSMO studio [7] and OWL-S IDE [8] ) emerging that seek to promote the development of SWS in general. However, most of these tools are not directly integrated with existing service development platforms. Moreover, these tools and platforms do not support the engineering of semantic-based services that are intelligent beyond the use of ontologies.

Furthermore, the issue of simple, rapid, and integrated engineering of semantic-based services tends to receive little exploration within the service-oriented research community. de Cesare *et al.* [9] affirm that limited work has been conducted with regard to the practical implementation of semantic-based services within the service engineering community.

In this position paper, preliminary efforts that seek to address the challenge of simplifying and speeding-up the process of engineering *intelligent semantic services (IsS)* in a unified manner are presented. In the context of the work presented in this paper, an *Intelligent semantic-based Service* (IsS) extends and leverages SWS with intelligent features adopted from intelligent agents such as autonomy (i.e. ability to act without or with little intervention and control own actions and state[10]) and pro-activity (i.e. ability to show goal-directed behavior and be able to take initiatives where possible [10]). Intelligent agents can play an important role in the adoption and usage of SWS [11], specifically in handling the automation of services' aspects and high level of co-ordination between service functionalities on the Web. According to Blois *et al.*[12], intelligent agents can also be quite useful for the development of SWS and related applications, particularly for purposes of reasoning about ontologies linked to services. Intelligent agents can also be beneficial in providing consumers with context-aware services, eliminating the issue of irrelevant service provisioning [13]. In this paper, an IsS is defined *as a semantically-rich software unit representing some business activity, and is capable of being: (1) autonomous, (2) pro-active and reactive, (3) interoperable, (4) composable, and (5) reusable.* Since IsS augments SWS with intelligent agents' capabilities, it should be noted that an IsS possesses all the concepts of WS, SWS and ontologies.

The remainder of this paper is structured as follows: In Section 2, research motivations and objectives that guide this study are detailed. The applicable use-case scenario is presented in Section 3. Section 4 briefly describes the solution approach adopted for the proposed study. The initial iSemServ architecture framework is briefly presented in Section 5 In Section 6; the main contributions from this study are explained. The state-of-the-art related to the semantic service engineering platform is analyzed and presented in Section 7. In Section 8, a brief description of the work done to date is given including current and future plans for the proposed study. Finally, in Section 9 the position paper is concluded with a summary.

## 2 Motivation and Objectives

The preliminary research work presented in this position paper is motivated by the need to minimize and simplify the technical complexities (e.g. ontology development [6]) experienced by service designers and developers when implementing and deploying semantic-services. Secondly, service designers, and providers value user-friendly, simple and yet unified technical platforms that are capable of assisting to rapidly deliver value-added services for consumption on the Web. Nevertheless, such platforms are hardly available in the SWS domain, and this in turn result in a tedious and error-prone process of engineering intelligent semantic-services, leading to services that are of low quality and performance.

The research proposal presented in this position paper is thus based on the following research question: *How could the process of engineering intelligent semantic-based service (IsS) be simplified and accelerated with the use of an open integrated platform?* In order to satisfactory deal with the main research question, the following research objectives are identified. The first objective is to *identify and characterize a set of fundamental building blocks that make up an IsS.* The notion of IsS is emerging, and with this particular objective we attempt to understand and provide solutions as to what an IsS is, how is an IsS distinct from WS and SWS, and what components (building blocks) make up an IsS from the technical perspective. Furthermore, within this objective the goal would be to understand how the identified building blocks will be interoperated to further improve and promote simplicity when it comes to the actual development of IsS. The second objective is about *devising a service creation framework.* To engineer (i.e. model, build, and advertise) a functional IsS within an integrated platform; a number of essential components need to be identified, and analysed. The objective in this regard is to review and propose components that forms part of the service creation framework that will serve as a blue-print for engineering IsS with an integrated service creation platform. The framework will be designed based on principles of *simplicity, rapidness, complexity hiding, reusability, uniformity, and intelligence*. Thus, guiding and supporting engineering of services through simple ontologies development, intelligence wrapping of services to promote service dynamism and degree of automation, and development of semantic-based services in a cost and time-efficient manner.

The third objective is about *exploring, selecting, and/or developing software tools that can contribute to the simplification and acceleration of the process when engineering IsS* for different purposes. The *development of an integrated service creation* *platform* using deliverables from the third objective forms part of objective number four. In this objective, a technical platform would be delivered for purposes of testing and validating the suggested framework. The other objective of this proposed study is the actual development of prototypes based on identified use-case scenarios.

## 3 Use-Case Scenario

One of the use-case scenarios that have already been identified for this study is an intelligent e-procurement scenario adapted from [6]. In this scenario, we look into demonstrating the implementation of a web-based application composed of intelligent semantic services developed by retail suppliers. This is purported to also illustrate how intelligent semantic services can minimize the hurdles experienced by small scale traders in the rural areas of South Africa who travel long distances on regular basis back-and-forth to urban cities to procure stock for their retail shops. In this scenario, suppliers develop and make available intelligent semantic services using our suggested service creation platform. These services are capable of co-ordinating with intelligent agents in assisting small scale traders in the rural areas to seamlessly "shop around online" using their mobile devices or desktop computers. For example: a small scale trader can use his/her device to simply draw up a digital shopping list. After that, the intelligent agent(s) cooperating with semantic services take over, and with little human intervention shops around from available suppliers and considers users requirements.

Once the "shopping" is completed, a preliminary basket with a variety of deals or quotations is sent back to the user for confirmation. The users chooses the preferred option, and the agent(s) in co-ordinating with semantic-services residing at relevant suppliers places an order, and forwards the final invoice to the shop owner. Depending on the business model, once the payment has been received, the supplier interacts with the shop owner to finalize delivery of ordered goods. The main aspects that are to be addressed in our work, and are implicitly illustrated in this scenario are that of simplifying the process of modelling semantic service functionality using techniques such as the Eclipse Modelling Framework (EMF) [14]; seamless transformation of service model into partial semantic service functionality using converters such as UML2Java; manual and guided extension of service functionalities from partial translations, automatic generation of syntactic service descriptions, seamless translation of service descriptions into partial ontological specifications (e.g. WSDL2WSMO [15]), visual and/or textual building of domain ontologies for suppliers, and service ontologies for different services (e.g. product catalogue services) using for example Web Service Modelling Toolkit (WSMT [16]); visual and automatic annotation of services; and lastly intelligence wrapping of semantic services.

## 4 Solution Approach

The proposed research adopts a quantitative research approach with the main focus on model-building, prototypes, laboratory experiments, and computer-based simulations. The primary research methodology adopted for this study is that of modelling, where a service creation framework/model is devised and tested based on functional and non-functional characteristics as deduced from literature and use-case scenario as briefly described in Section 3. The service creation framework (see Section 5), is developed as a base for the technical platform that could make it possible to simplify the process of engineering semantic services.

Using at least two use-case scenarios; domain specific services and applications demonstrating the practicality and utility of the service creation framework and the integrated service creation platform will be produced via the prototyping research approach. Laboratory experiments will be conducted with an objective to gain deep insight into the service creation framework and to note the effects of the framework and the platform when developing IsS as compared to currently existing models and platforms. Through the experiments; simplicity (ease-of-use), usefulness, and uniformity of the service creation framework and the unified engineering platform will be tested and measured through simulations and usability test with sampled service developers. In the proposed research, computer-based simulations are applied mainly for demonstrating philosophical properties of an IsS such as *intelligence* and *context-awareness*.

## 5 Preliminary Solutions

In this section, a preliminary *i*SemServ architecture framework is presented as one possible approach for accelerating and simplifying the process of engineering intelligent semantic-based services that can autonomously accomplish Web-based tasks on behalf of its users. The architecture-framework is structured into the *service layer*, *semantic layer*, and *intelligence layer*. The service layer deals with the activities related to the actual design and development of syntactic services.

In the semantic layer, ontologies are generated using WSMO [7] and semantic annotations are performed using various techniques adopted from WSDL-S [17] and ServFace[1]. As the main contribution from the main research of this position paper, the intelligence layer enables semantically-annotated services to be intelligent, by enabling semantic services to be autonomous, proactive, context-aware, and highly automatic.

The architecture framework follows a SOA-based and an Agent-based approach. Furthermore, the *i*SemServ architecture framework is based on these functional requirements as per research literature and challenges addressed in the overall study (see Section 1). In the following subsections, the main constituent components of the *i*SemServ architecture framework are briefly explained.

### 5.1 Visual Modeller

This component deals with service modelling. In order to satisfy the requirements of *simplification* and *rapidness,* VUML [18], a view-based design method using UML is adopted for the *i*SemServ. This is to enable users to model the service using mainly notations and diagrams. VUML is essential for *i*SemServ as it can ease the service engineering process by automatically generating relevant service source code once a visual service model has been created.

[1] http://www.servface.eu

## 5.2 Service Composer/Importer

It is possible for the required IsS to be constructed from existing WS or SWS as provided by external providers. Hence, the *service composer/importer* component deals with the activities related to the composition of external services and/or importing of single services for use during the development process. External services can be imported or composed using service descriptions (e.g. WSDL) from external registries. The *flexibility and rapidness* requirements of the framework are partially satisfied by this component.

## 5.3 Service Editor

This component deals with the actual coding and extensions of the functionalities related to the required service using either the VUML generated code or service descriptions of imported or composed services. Eclipse plug-ins such as WSDL2Java could be used for reverse-engineering purposes, with regards to the imported services. In order to satisfy the *complexity hiding* requirement, other aspects of service development such as service description are hidden from the developer as service descriptions are generated automatically.

## 5.4 Service Descriptor

This component is mainly about formally describing constructed syntactic services. Although this process is envisioned to be automatic and semi-transparent to the user, it is worth explaining. This process is mainly about syntactically describing the functional and non-functional properties of the constructed service. These descriptions will be automatically generated from the service source code as generated and made available during the service construction phase. Java2WSDL, an Eclipse plug-in can be used for automatically generation service description file.

## 5.5 Ontology Editor

This component deals mainly with the construction of service, domain, and context ontologies for the service produced in the service layer.

## 5.6 Agent Platform

This component deals specifically with the development and/or reuse of agents that can envelop semantic-based services to produce IsS. The agent platform also facilitates the mapping of agents with semantic-service capabilities, service and domain ontologies, and context knowledge. This is done to simplify and automate the process of reasoning and processing ontologies services and agents interaction.

## 5.7 Intelligence Wrapper

The *intelligence wrapper* component is responsible for wrapping a developed semantic-based service with *intelligence* to materialize a concrete IsS. This activity is semi-automatic and is achieved through the use of the agent platform component called JADE [19].

### 5.8 IsS Validation, Deployment, and Publication

Depending on the complexity of the constructed IsS, the validation process could be undertaken to ensure that the service behaves and executes as desired. The final stages of the proposed *i*SemServ architecture framework deal with the internal deployment and publication of a functional IsS into an environment where users can access and execute available intelligent semantic-based services.

## 6 Main Contributions

The main scientific contributions from the proposed research to the field of Service and Web engineering respectively is a multi-layered service creation framework and an open integrated service creation platform that purport to simplify and accelerate the process of engineering intelligent semantic-based services by introducing visual service design and development tools, user-friendly and integrated ontology development tools, seamless intelligence wrapping of the service, and finally improve level of automation in services through the use of software agents.

On the other hand, the social contributions are on aiding both traditional and modern enterprises in designing and developing intelligent semantic-based services and applications that could promote their competitiveness and growth on the Web, in a cost-effective manner. Moreover, from the proposed study, the following research contributions are also envisaged: (1) provide a suitable test-environment for semantic-based services and related applications, (2) promote the uptake of semantically-enriched services and applications by supporting semantics reusability and interoperation in heterogeneous semantic-based applications, and (3) minimize the time and costs required for engineering semantic-based services and related applications. The limitations that are envisaged in this study are that of limited evaluation and validation. Intelligent semantic-services would not be deployed in a real-life environment, but mostly tested in a lab environment.

## 7 Related Work

ODE-SWS, a SWS development environment by Corcho *et al.* [20] focuses on developing SWS in a language independent approach. Various semantic description languages can be used within this platform. The framework is integrated within WebODE, an ontology engineering workbench, responsible for exporting provided ontology into other ontology languages [21]. WebODE is considered in the semantic layer of the proposed *i*SemServ framework for flexibility and interoperability purposes. On the down side, ODE-SWS environment does not consider intelligence aspects for services as included in our proposed research work

One of the frameworks that claim to be the first toward SWS engineering is called INFRAWEBS [22]. It focuses on constructing semantics data for existing and new services. INFRAWEBS is made up of different units (i.e. SWS creation, monitoring, selection, discovery, composition, and conversion), which are essential to the actual development and implementation of semantic-based services. INFRAWEBS is limited in a sense that it is tightly bound to a specific ontology language (i.e. WSMO); hence it is not flexible and does not use software agents for achieving high degree of automation.

Internet Reasoning Service (IRS-III) is a comprehensive framework and a platform for creating WSMO-based SWS [23]. IRS-III is promoted as a development framework for SWS. According to Domingue *et al.* [24], the main goal of IRS-III is to support capability-based discovery and invocation of semantic services. Its other main role is to mediate between service providers and service consumers using ontologies to further enhance interoperability and collaboration. However, IRS-III does not necessarily provide an environment where new semantic-based services can be visually engineered based on users' requirements.

Finally, one of the recent research endeavours that is closely related to the work of this paper is that of Srinivasan, Paolucci, and Sycara [25]. The authors proposed and developed an integrated development environment (IDE) called CMU's OWL-S Development Environment (CODE) for developing, deploying, and consuming semantic-based services. CODE adopts and extends existing WS and ontology engineering tools (i.e. OWL-S editor, WSDL2OWL-S converter, and so on) in order to support developers with the process of developing, deploying, and consuming semantic-based services [26]. It is embedded within the Eclipse environment and is purely based on OWL-S and Java; and follows a multi-methodological approach by applying both code-driven and model-driven methodologies. The CODE platform supports various SWS activities such as discovery, invocation, and execution. However, on the down side, the platform does not cater for flexibility, interoperability, and intelligence, which is one of the main contributions from this paper. It should be noted that some of the components within the proposed *i*SemServ framework are adopted and extended from the CODE platform.

## 8 Current Status and Future Directions

The theoretical investigation on technologies such as WS, SWS, ontologies (e.g. WSMO and OWL-S), and service engineering has been completed. The deliverables from the investigation are three draft literature review chapters. The task of identifying and characterizing the basic building blocks that constitute an IsS has been completed. Currently, preliminary experiments are being conducted to operationalize and analyse the identified basic building blocks. Furthermore, the components that are envisaged to form part of the service creation framework are being identified, analysed, and selected.

A preliminary *i*SemServ architecture framework is presented in this paper (see Section 5). The framework has been theoretically evaluated through comparison analysis, against other service creation frameworks such as INFRAWEBS [3], CODE [25], and IRS-III [24]. Above that, the service creation platform will then be developed based on the *i*SemServ framework, and use case scenarios will be refined for prototypical demonstrations and experimentation purposes.

## 9 Conclusion

The emergence of machine-interpretable and machine-processable Web services promises much improved service provisioning and consumption. However, without robust, simple, and mature platforms; the wide spread adoption and application of semantic-based services could always be limited. Furthermore, enterprises facing challenges of software costs, development costs, and technical skills will also lag behind. Hence, it is essential for the research community to timely provide guidelines, techniques, user-friendly tools, and open platforms that can enable businesses and Web users to produce and consume semantic-based services in a real-world setting. In this position paper, a proposal was presented on how these challenges can be addressed. The main contributions envisaged from the proposed study are a service creation framework and a technological service creation platform intended to simplify and support the phases of building a functional intelligent semantic services and related applications in an integrated manner.

## Acknowledgment

The support of SAP Research Centre Pretoria and SAP Meraka UTD (CSIR) towards this research is hereby acknowledged. Opinions expressed and conclusions arrived at are those of the authors and not necessarily to be attributed to the companies mentioned in this acknowledgement.

## References

1. Alonso, A., Casati, F., Kuno, H., Machiraju, V.: Web Services: concepts, architectures, applications. Springer, Heidelberg (2004)
2. Lu, J., Zhang, G., Ruan, D. (eds.): E-Service Intelligence: Methodologies, Technologies and Applications, vol. 37. Springer, Heidelberg (2007)
3. Agre, G., Marinova, Z., Pariente, T., Micsik, A.: Towards Semantic Web service engineering Workshop on service matchmaking and resource retrieval in the semantic Web (SMRR 2007), CEUR (2007)
4. Corcho, O., Silvestre, L., Benjamins, R., Bas, J.L., Bellido, S.: Personal ebanking solutions based on semantic web services. Studies in Computational Intelligence (SCI) 37, 287–305 (2007)
5. Bensaber, D.A., Malki, M.: Development of semantic web services: model driven approach. In: 8th international conference on new technologies in distributed systems. ACM, Lyon (2008)
6. Studer, R., Grimm, S., Abecker, A. (eds.): Semantic web services: concepts, technologies, and applications. Springer, Berlin (2007)
7. Dimitrov, M., Simov, A., Momtchev, V., Konstantinov, M.: WSMO Studio - a semantic web services modelling environment for WSMO. In: 4th European Conference on the Semantic Web: research and applications. Springer, Innsbruck (2007)
8. Elenius, D., Denker, G., Martin, D., Gilham, F., Khouri, J., Sadaati, S., Senanayake, R.: The OWL-S editor – a development tool for semantic web Services, The Semantic Web: Research and Applications, pp. 78–92. Springer, Heidelberg (2005)
9. de Cesare, S., Holland, G., Holtmann, C., Lycett, M.: Semantic-based systems development. In: 22nd ACM SIGPLAN conference on Object-oriented programming systems and applications companion OOPSLA 2007. ACM, Montréal (2007)
10. Jennings, K., Wooldridge, M.: Software Agents. IEEE Review, 17-20 (1996)

11. Garcia-Sanchez, F., Valencia-Garcia, R., Martinez-Bejar, R., Fernandez-Breis, J.T.: An ontology, intelligent agent-based framework for the provision of semantic web services. Expert Systems with Applications 36, 3167–3187 (2009)
12. Blois, M., Escobar, M., Choren, R.: Using agents and ontologies for application development on the semantic web. Journal of Brazilian Computer Society 13, 35–44 (2007)
13. Aziz, Z., Anumba, C., Ruikar, D., Carrillo, P., Bouchlaghem, D.: Semantic web based services for intelligent mobile construction collaboration. ITcon 9, 367–369 (2004)
14. Budinsky, F., Steinberg, D., Merks, E., Ellersick, R., Grose, T.: Eclipse Modeling Framework. Addison Wesley Professional, Reading (2003)
15. El Bouhissi, H., Malki, M., Bouchiha, D.: A reverse engineering approach for the Web Service Modeling Ontology specifications. In: Second International Conference on Sensor Technologies and Applications (SENSORCOMM 2008), Cap Esterel, pp. 819–823 (2008)
16. Kerrigan, M., Mocan, A., Tanler, M., Fensel, D.: The Web Service Modeling Toolkit (WSMT)- an integrated development environment for Semantic Web Services. In: 4th European Conference on the Semantic Web: Research and Applications. Springer, Innsbruck (2007)
17. Akkiraju, R., Farrell, J., Miller, J., Nagarajan, M., Schmidt, M.-T., Verma, A.S.K.: Web Service Semantics - WSDL-S. W3C (2005)
18. Nassar, M., Anwar, A., Ebersold, S., Elasri, B., Coulette, B., Kriouile, A.: Code generation in VUML profile: A model driven approach. In: IEEE/ACS International Conference on Computer Systems and Applications, pp. 412–419. IEEE, Rabat (2009)
19. Bouhissi, H.E., Malki, M., Bouchiha, D.: Towards WSMO ontology specification from existing Web Services (2006)
20. Corcho, O., Gomez-Perez, A., Fernandez-Lopez, M., Lama, M.: ODE-SWS: A semantic web service development environment. In: 1st International Workshop on Semantic Web and Databases (SWDB 2003), Berlin, Germany (2003)
21. WebODE: WebODE ontology engineering platform. Vol. 2009 (2003)
22. Nern, J., Agre, G., Atanasova, T., Marinova, Z., Micsik, A., Kovács, L., Saarela, J., Westkaemper, T.: INFRAWEBS semantic Web service development on the base of knowledge management layer. International Journal on Information Theories and Applications (IJITA) 13, 161–168 (2006)
23. Cabral, L., Domingue, J., Galizia, S., Gugliotta, A., Norton, B., Tanasescu, V., Pedrinaci, C.: IRS-III: a broker for Semantic Web Services based applications. In: Cruz, I., Decker, S., Allemang, D., Preist, C., Schwabe, D., Mika, P., Uschold, M., Aroyo, L.M. (eds.) ISWC 2006. LNCS, vol. 4273, pp. 201–214. Springer, Heidelberg (2006)
24. Domingue, J., Cabral, L., Galizia, S., Tanasescu, V., Gugliotta, A., Norton, B., Pedrinaci, C.: IRS-III: A broker-based approach to semantic Web services. Web Semantics: Science, Services and Agents on the World Wide Web 6, 109–132 (2008)
25. Srinivasan, N., Paolucci, M., Sycara, K.: CODE: a development environment for OWL-S Web services. Robotics Institute, Carnegie Mellon University (2005)
26. Srinivasan, N., Paolucci, M., Sycara, K.: Semantic Web Service Discovery in the OWL-S IDE. In: 39th Annual Hawaii International Conference on System Sciences, vol. 6, pp. 109–118. IEEE Computer Society, Kauai (2006)