Multilevel Life-Event Abstraction Framework for e-Government Service Integration

Farzad Sanati and Jie Lu
University of Technology Sydney (UTS), Australia
fsanati@it.uts.edu.au
jielu@it.uts.edu.au

Abstract: One of the fundamental attributes of modern government service delivery mechanism is the ability to offer a citizen-centric view of the government model. Life-event model is the most widely adopted paradigm supporting the idea of composing a single complex service that corresponds to an event in a citizen's life. Elementary building blocks of Life-event are based on atomic services offered from multiple government agencies. Composite services are desirable mainly because of their added value to businesses and government agencies. This study found that the methodological mechanics of service integration, and in particular, the requirements engineering for services integration has been overlooked. It introduces a multilevel modelling framework for analysis and design of Life-event within the government service integration context based on the principle of abstraction. It also proposes a top down multilevel abstraction approach to model Life-event candidates and elicit their requirements and specification. This study explains the problem space of e-government service delivery integration, and stresses the ontology analysis and modelling as one of the essential requirements for modelling Life-events.

Keywords: e-Government, integration, life-event modelling, semantic ontology, e-services

1. Introduction

Growing presence of government agencies in the Internet has created a congestion of similar and some times duplicated web services; this problem has prompted governments to plan for e-government service integration. Life-event is the most widely adopted paradigm supporting e-government service integration task model (Vassilakis et al. 2005). It combines basic services offered from multiple public authorities into a single high-level service that corresponds to an event in a citizen's life.

Service Oriented Architecture (SOA), and in a broader sense, Service Oriented Computing (SOC) has influenced Information and Communication Technology (ICT) towards the design of uncoupled yet coherent architecture of services. Current ICT industry trends indicate that organizations and solution vendors moving towards the decomposition of legacy complex processes into atomic and simpler components to handle ever increasing complexity of current information systems (Huhns & Singh 2005). This trend has led to a two-step solution: Step 1 is to transform gigantic architectures into constructs, consisting simpler building blocks, called services; and Step 2 to recompose these services in to composite services in order to achieve added value. This study is mainly concerned with the second step of this theory, and suggests that more research work is needed in order to shade light over the role of Life-event in e-government service integration. This paper makes the following contributions:

- Establishing a multilevel abstraction framework of Life-event as a unit of requirements for e-government service integration;
- Within this framework, proposing a generic requirements engineering model for automated e-government service integration.

This study will apply experimental research methodology (Trochim 2001). Base on the principles of this model, this study plans to achieve its objectives within the following scope and order: Section 2 reviews literature to define the problem space of this study and support its arguments. Section 3 explains a methodological approach to analyse e-government service delivery mechanism and the role of Life-even in e-service composition. Later in that section the theory of ontology analysis and modelling and their place in developing automated e-government integration is discussed. The reason why ontology analysis should be considered an essential part of requirements engineering for such systems is also argued later in Section 3. Section 4 explains the implementation mechanics of the proposed Life-event requirements engineering model that can be put in practice by using previously proposed set of tools (IESD platform) with the help of Meta-modelling technique. At the end of this paper, the main contributions of this study are revisited and explained with a hint on future research plans.

2. Literature review

In addition to the literature reviewed in Section 1, the review of many other recent e-government integration solutions (Madhusudan 2006; Umapathy & Purao 2007) also (Beer, Kunis & Rünger 2006; Dijkman & Dumas 2004; Liu, Husni & Padgham 2007; Lu, Zhu & Chen 2004; Medjahed et al. 2003; Meneklis et al. 2005; Peng, Yanzhang & Xuehua 2006) indicate that most of the efforts have mainly focused on enabling technologies in order to achieve the desired out come, with very little or in some cases no attention to any methodological framework. This fact indicates that most of the research efforts are deeply consumed with the practical implementation of technology aspect of their proposed solution, undermining the importance of a methodology.

Recent research work has recognised the importance of Life-event in e-government service composition, some studies identified the importance of requirements elicitation for Life-event, although the results mainly focus on B2G aspect of e-government. The model presented by Wolf & Krcmar (2008) seems to be very specific application only designed for B2G. They suggest a model of features and phases, which might not sufficiently analysed the provisioning of the further development of such model. In relation to the role of ontology in e-government integration there are other works by Grosof et al. (2004) and Lara et al. (2003) also to some extend in Stojanovic & Apostolou (2006) work where practical implications of ontology in e-government integration interoperability issues are analysed. These later works seem to have only listed a set of functional requirements for ontology building and overlooked the overall qualitative criteria that ontology should address. The analysis of other relevant literature (Trochidis, Tambouris & Tarabanis 2007) indicates that there are two main approaches for modelling life-events. The first approach suggests a model of life-events to be the workflow of related public services and actions (Trochidis, Tambouris & Tarabanis 2006). The second approach is suggesting modelling the life-events using ontology (Peristeras & Tarabanis 2006) thus capitalizes on the idea of semantic representation of knowledge. This approach describes ontology as the network of connections between concepts of a particular domain with the aim to provide a well-structured model. This study is capitalising in the second approach in order to conduct further research on the practical mechanics of such theory.

3. Life-event modelling and analysis process

Life-event is also described as guiding metaphor for customer-centric public service provision, from e-government integration point of view, a Life-event is a collection of actions including at least one public service, which when executed in its appropriate workflow to fulfil the needs of a citizen arising from a new real-life situation (Trochidis, Tambouris & Tarabanis 2007). This section explains the principles used for analysis and modelling of Life-event.

3.1 Multilevel life-event abstraction

Requirements engineering is mostly considered to be a set of activities concerned with identifying and communicating the purpose of a software-intensive system, and the contexts in which it will be used. However the evidence indicate that classical software engineering processes such as Object Oriented Analysis and Design (OOAD), Business Process Modelling (BPM), and Enterprise Architecture (EA) frameworks are not fine tuned to handle the analysis and design of SOA (Sanati & Lu 2007).

The concept of abstraction in object-oriented paradigm plays an important key roll in representation of complex data structures. Abstract objects or data structures can form hierarchical representations to provide easy to understand solutions for complex models. Abstraction is the means by which only certain level of details of information is exposed by the entity, depending on the levels of representation intended for that model. Deferent levels of data abstractions are also known as level of granularity of the model. This study invokes the principle of data abstraction in context of Life-event to represent composite service in deferent levels of granularity depending on the detailed information about its underlying service structure and business rules.

3.2 Semantic ontology qualitative analysis

Ontology in general, plays an important role in semantic web mainly because it can provide a more flexible way of introducing semantics into web-based information systems than other proposed standards. This paper stresses that performing semantic analysis for available services is an important activity in *Requirement Specification* phase of a software project, it enables the target system to define its own vocabulary based on existing domain concepts (Pan 2007). Therefore, it is also necessary for service integration projects to prepare a catalogue of service, regulatory, and domain ontology to enhance semantic interoperability. One of the most important areas of ontological analysis discussed in this study is government regulatory rules and processes. These regulations organized in an ontological tree that binds the semantic correlation of all requisite regulatory rules of government services together in order to achieve a correct order of execution and acceptable *legal outcome* from the Life-event workflow execution.

This paper argues that the requirements engineering definition for automated semantic software applications must be deferent than its currently commonly accepted definition in most traditional development processes. Traditional software development processes more or less agree on the statement by Nuseibeh & Easterbrook (2004) that the context in which requirements engineering takes place is usually a human activity system, and the problem owners are people. However as it explained later in this paper, some of the main stakeholders within service integration projects are other software components or remote systems rather than people, even though the end user of the Life-event intended to be citizens. This argument highlights the need to incorporate the semantic and ontology analysis in to requirements engineering for automated service composition projects. To achieve this goal we need to understand the reasoning criteria that ontology should address. Other related research works (Sabou et al. 2005) indicate that various categorizations of government related ontology seem to be falling in to two main types. First is Generic Ontology that captures the domain independent aspect of Life-event such as workflow execution rules. They need to be rich axioms to facilitate creating formal descriptions for reasoning purposes. The second is Domain Ontology that contains domain specific knowledge that is used to complete the generic descriptions. The importance of the later type of ontology is more evident when dealing with developments of automated composite services in a specific domain such as e-government.

3.2.1 Ontology requirements of e-government integration

The proposed modelling framework "*E-SIM*" suggests that the use of three types of ontology analysis is needed in order to achieve more comprehensive requirements elicitation for targeted life events:

- *E-government Ontology:* This ontology is of Domain Ontology type, it is cataloguing a semantic schema of government specific terms (i.e. technical, organisational and workflow process). This type of ontology contributes domain knowledge to the integration process. The use of Ontology Web Language (OWL) profile, enables semantic searching by travelling throughout concept branches of particular domain ontological models (Bell et al. 2007). Semantic search over several models grounded in real world "things" provides a greater scope for matching to a requestor's concept.
- *Regulatory Ontology:* As it was strongly acknowledged by other research literature (Lytras 2006), the diversity of structures, regulations and procedures affecting networks of heterogeneous administrative units represents a challenge for semantic integration. This type of ontology is specifically designed for e-government service integration since every service participant in any Life-event may imply or to be effected by one or more regulation. These regulations are the governing rules of composite services, specifically because regulations are one of integral parts of interagency processes (i.e. where Life-event process flow crosses multiple agencies). Furthermore, regulatory knowledge required for designing an inter-agency workflow that crosses the boundaries of local, state, and federal agencies.
- *Service Ontology:* while the regulatory ontology plays an important role in ensuring a legal outcome on execution of composite service workflows, service ontology is required to automate the acquisition of atomic service in the Life-event workflow. It provides service specific information such as availability, service type, service profile, required regulations and required communications parameters to the run-time workflow construction process. Service ontology descriptors could also connect to other ontology descriptors to obtain semantic information required by the workflow. Figure 1 illustrates the interconnected and shared ontologies.

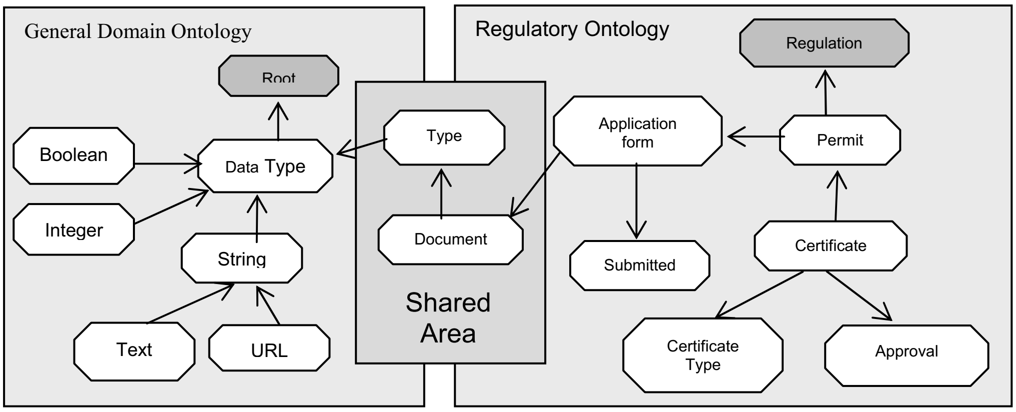

Figure 1: Partial domain ontology construct

4. Design and implementation of life-event

An automated dynamic process is required to enable the gradual integration of government services in an intelligent way to reduce the cost of data and process integration. Automated nature of E-SIM reduces the cost of data and process integration by allowing gradual and incremental integration of web services, where government agencies can decide when, how and which services to integrate.

4.1 Life-event construction process

Figure 2 is a view of IESD platform from users prospective, atomic services are registered in IESD platform in order to become available to participate in a Life-event. During the registration service owners provide additional semantic information about the nominated service to help construct service ontology. Later a business analyst is responsible for creating a Life-event Candidate (LeC) in E-SIM process, his/her view of the LeC is mostly business oriented therefore s/he does not need to see or know about most of the underlying technical details of the LeC. In this stage Life-event mostly reviles business related details and very little about its underlying technical structure. This is the first level of Life-event abstraction that provides very little details of its underlying implementation. In later stages of Life-event life cycle, as far as an integration engineer is concerned Life-event must expose a greater deal of technical details in order to construct its workflow schema or what is called here Life-event Meta-model (LeM). In Stage 2 IESD software component is acting as another stakeholder in E-SIM process. It must see every business rules and technical details of the life-event to be able to perform run-time reasoning, instantiation and execute of the Life-event Instance (LeI). Figure 2 illustrates different views of different stakeholders in Life-event life cycle.

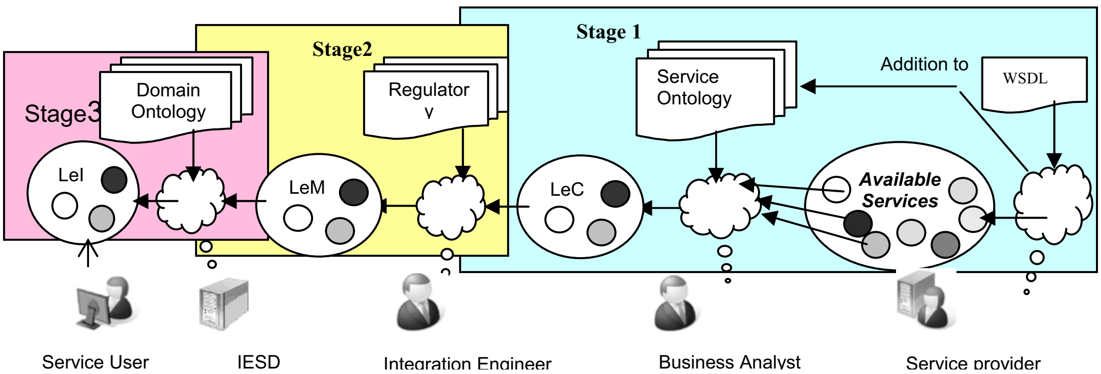

Figure 2: Life-event life cycle from users perspective

E-SIM process is of Control-oriented type using deadlock resolution, exclusion, concurrency, and process activation & deactivation. Therefore flowcharting that is one of the primary process-oriented modelling techniques is the best suitable modelling techniques that can be used to model the requirements of this type (Thayer & Dorfman 1977).

Figure 3 is the illustration of Life-event life cycle from a technical prospective beginning with the initiation candidate to the proposed Meta-model and finally the execution of a Life-event instance all in deferent levels of abstraction. This model is a component modelling view of E-SIM process with different types of stakeholders and their requirements, also demonstrates that a software component or a remote software system for example “Dynamic Reasoner” invokes atomic services to execute a Life-event is one of the main stakeholders of the system, as it is required to make execution decisions in run-time.

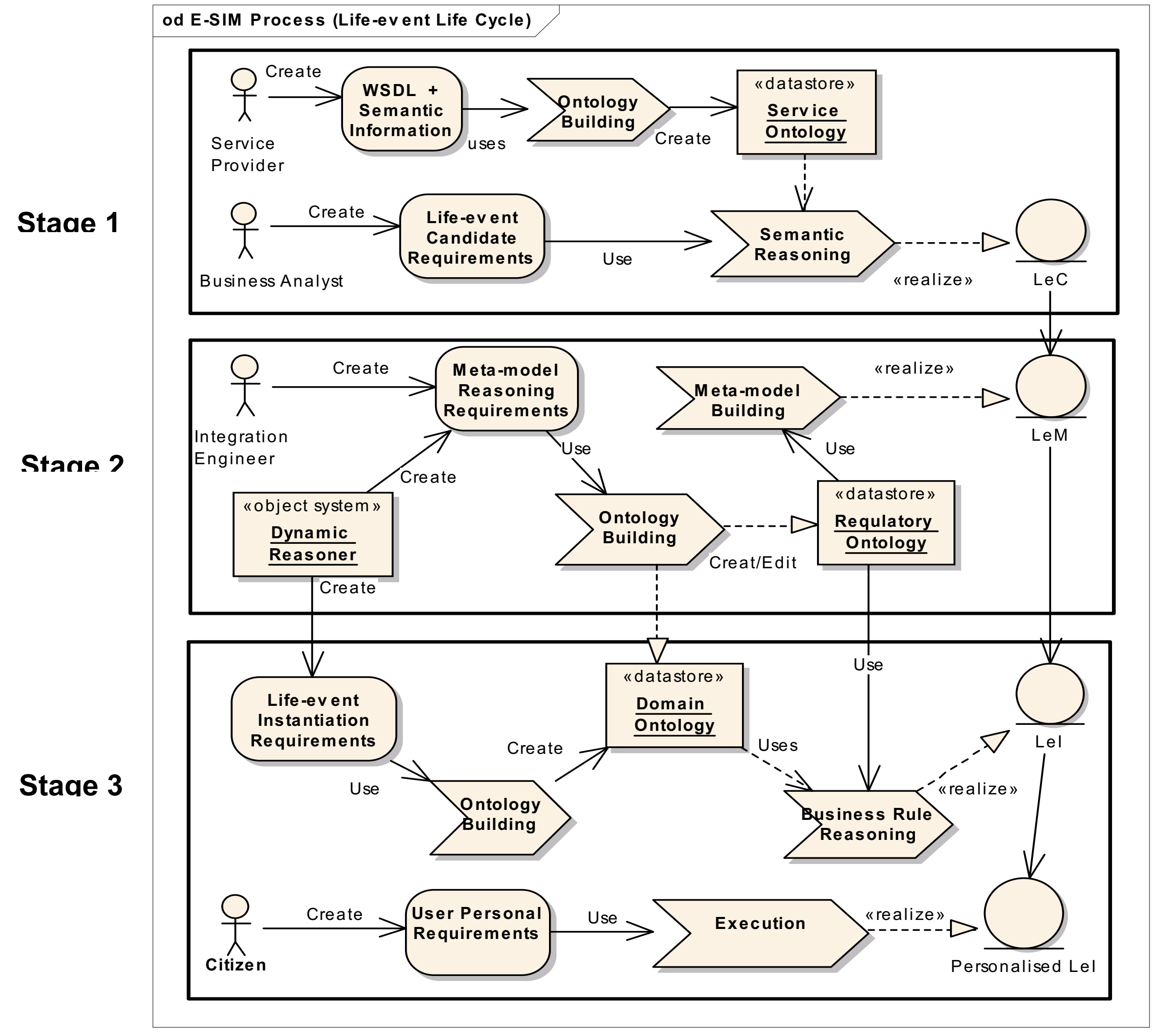

Figure 3: Multilevel abstraction of life-event in the E-SIM process

Following is the summarised explanation of technical prospective of E-SIM process and the contribution of ontology catalogues within its life cycle illustrated in Figure 3:

- *Stage 1:* LeC is proposed and service ontology must be created within the scope of the candidate. Requirements of this stage consist of participating service specifications and their Quality of Service (QoS) requirements (WSDL) and complementary semantic information about the service, which is needed to form an ontology schema. The main stakeholders of this stage are: (1) Web Service providers (remote atomic Web Service owners), (2) Business analyst that creates the first level abstraction of a Life-event. Output of this stage is a LeC and its service ontology schema.
- *Stage 2:* LeM is to be created based on LeC specification from Stage 1. Requirement of this stage is regulatory specification, which provides the governing rules for the workflow of the LeM and the flowchart of the LeC runtime workflow. One of the main stakeholders of this stage is an integration engineer that performs analysis and modelling to create the second level abstraction of a Life-event, which is a LeM. The outputs of this stage are regulatory ontology schema, and the Meta-model specification or what is called LeM.
- *Stage 3:* This is the Life-even instance execution stage; in this stage the third level abstraction of Life-event is created, this is an executable Life-event Instance (LeI) that is created based on its

LeM upon a service consumer request. One of the important stakeholders of this stage is the *Dynamic Reasoner* component of the IESD platform that will have great interest on reasoning upon regulatory ontology to deduce the alternative logical pathways of LeM. The requirements of this stage consist of but not limited to user preferences data, domain information, and QoS. The input of this stage is Meta-model specification from Stage 2 and a domain ontology schema that is required for run time reasoning, the output of this stage is an executable instance of the LeM.

4.2 Dynamic service integration

OWL has proven to be a very powerful tool to enable the use of semantic information in web applications. In terms of implementation there is a narrow but fundamental deference between the implementation of traditional software and the implementation of Life-event. A Life-event is not just another software written in a certain language then tested and installed for the end-user to use. As we discussed earlier it is a description of the mechanism on how to assemble a composite service from already existing web services, although it would require testing and delivery, but assembly and delivery of Life-event is mostly an automated process.

Modelling and design methods in e-service integration must help the visualization of Life-event technical design, this design document must clearly describe Life-event candidate for the delivery platform to create a Meta-model and consequently an execution workflow (LeI). The process must provide traceable technical information regarding the technical specification of the Meta-model. The workflow modelling and design could further be divided in to generation and specification stages (Husni & Padgham 2007). However workflow specification and generation model descried by Liu and Padgham (2007) are only suitable for static workflows. Static workflows are only generated at compile time (not run-time), where E-SIM process (Sanati & Lu 2007) allows for designing Meta-models, which then instantiated by an intelligent reasoning engine, resulting in one or more alternative instances of the workflow.

There are three types of workflow models (manual, semi-automatic or automatic) in respect to their design and execution process, they defer by the level of semantic and dynamic knowledge representation of the model. Different technologies are used to specify and design composite service workflows, depending on the level of semantic intelligence representation they can be implemented using different standards. Industry standards such as BPEL4WS are more suitable for static workflows with no semantic intelligence and are entirely configured at design time, where as OWL (McGuinness 2004) contain specific semantic information that could be used to design a dynamically generated workflows.

4.3 Meta-modelling technique

Design specification of dynamic workflow models also differ from those of static models in a way that for dynamic workflow models the designer only needs to produce a model that basically only specifies the type of the services, regulatory rules, and the order of execution. This model, which is called Meta-model contains semantic regulatory information to dictate the terms and conditions of the execution to determine, whether it is the right time to execute a particular service, and how the results of this execution would effect the overall state of the workflow. Later Meta-model is instantiated to generate instances of executable workflows (Life-event instances or LeI) suitable for different service user scenarios. Diagram in Figure 4 is the illustration of a Meta-model and its run-time instantiation by IESD platform. The Meta-model is not a concrete entity therefore its attributes are tend to be mostly generic since much about the details of the executable instance is not known until the run-time. Attributes of the Meta-model could include things like:

- The types of participating atomic services
- Regulatory rules (ontology) that could indicate the conditions and the order of the execution of atomic services
- Service ontology that provides semantic information about remote services
- Intermediate data required to enable asynchronous and interrupted workflow executions, and

A LeM is in fact the generic specification of workflows in design time, and LeI is an instance of LeM that implements its runtime specification. Our design allows for the use of Meta-model, to instantiate and executes specific Life-event in run-time based on availability of atomic services and other quality attributes of the participating services. E-government service integration must pay specific attention to the Ontology of Regulatory Rules, since the regulations are the main contributors to the execution order and runtime specification of workflows. Figure 4 illustrates the concept of Meta-model in the context of life-event.

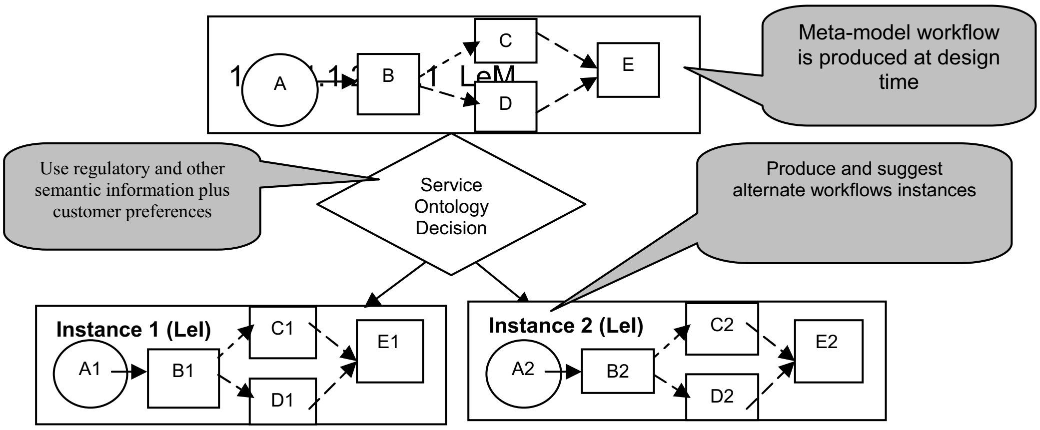

Figure 4: Composite e-service execution meta-model

Our design strategy is to facilitate seamless evaluation of services that are nominated for composition. Improved compose-ability for run-time workflow construction requires mechanisms that address the ontology requirements, profiles, and their underlying formalism. Composite service Meta-model for a Life-even is designed to fulfil those requirements, this generic model is specified as follows:

- A LeM workflow only indicates the type of an atomic service nominated for composition, as well as the order of execution. The specification of the instance of individual services (LeI) is configured dynamically at run-time. For example a workflow is designed to use a service of type “driving school”, there may be X number of different school services available at any given time.
- A reasoning Engine can use available ontology information to decide which specific service can be used at run-time, given the customer requirement parameters and current state of execution.

4.4 Life-event life cycle activities

In this section we explain the main three stages in Figure 3 in more details to show the actual interaction of IESD platform with the system users within the E-SIM framework. It is important to note that much of the effort of developing ontology could be devoted to hooking together classes and properties in ways that maximize their implications. The diagram in Figure 5 illustrates how IESD platform is performing a fusion of static data provided by WSDL and semantic information obtained from service provider to construct the relevant ontology in OWL format. In this activity IESD platform interacts with system users (perhaps a business analyst that is acting as the LeC designer) to construct a LeC by combining the syntactic information from WSDL with semantic information provided by the LeC designer.

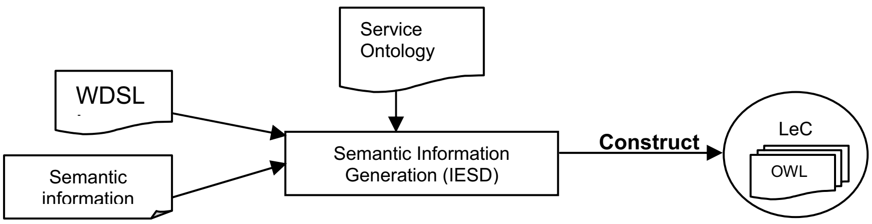

Figure 5: Service ontology and LeC construction

Next logical step (Stage 2) in E-SIM framework is demonstrated in Figure 6. Functionality of IESD platform allows for the interaction with an integration engineer to create a LeM. In this activity, the integration engineer can use functionality of the IESD platform to construct a LeM. This process uses the LeC specification data produced in ontology analysis activity of Stage 1 namely OWL documents. In this activity a software engineer is to create or edit the Meta-model using automation functionalities available in the IESD platform.

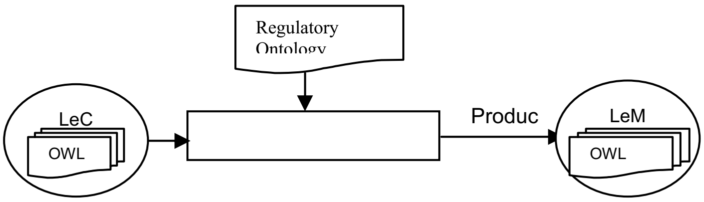

Figure 6: LEM creation and editing

The final stage of Life-event life cycle in E-SIM process is where e-government service users request the execution of a Life-event. In this stage the IESD platform analyses the user's life-event request specifications in conjunction with LeM workflow requirements and domain ontology specifications to deduce execution decisions. In this stage: 1) appropriate available services are selected, 2) regulatory rules are applied and, 3) service user profile is constructed in order to instantiation and execute a personalised composite service workflow or what we call here "Personalised LeI". The event of LeI instantiation is illustrated in Figure 7.

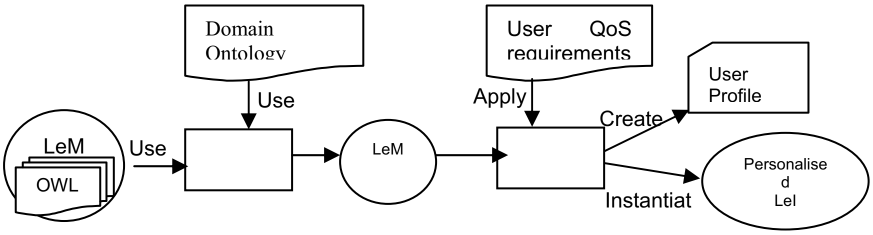

Figure 7: Runtime workflow construction and LeI execution

5. Conclusion and future research

This study proposes a repeatable process to e-government service integration process, which is mostly overlooked in relevant literature. The lack of a unified common practice for e-service composition projects is clearly visible in e-service development domain, and consequently e-government service integration is particularly suffering from this problem. It not only proposes enabling tools and technologies but also introduces an innovative approach towards the whole process of e-government integration. Particularly, this paper proposes an evolutionary concept of using Life-event as an abstract unit of requirement for composing e-government services, and introduces a model that illustrates the roll of Life-event within the process of e-government service composition.

More research is required to fine grain and specify the types of documentation required for our proposed model. In addition to modelling and implementation of ontology applications, future research will need to focus on how semantic attributes of service components can be technically modelled and expressed in service descriptors to enable automatic discovery, integration, reasoning and verification using domain and service ontology techniques.

References

Beer, D., Kunis, R. & Rünger, G. 2006, 'A Component Based Software Architecture for E-Government Applications', paper presented to the *First International Conference on Availability, Reliability and Security*, Hawaii, September 2006, in CD.

Bell, D., Cesare, S.d., Iacovelli, N., Lycett, M. & Merico, A. 2007, 'A framework for deriving semantic web services', *Information Systems Frontiers*, vol. 1, no. 9, pp. 69-84.

Dijkman, R. & Dumas, M. 2004, 'Service-Oriented Design: A Multi-viewpoint Approach', *International Journal of Cooperative Information Systems*, vol. 13, no. 4, pp. 337-368.

Grosof, B., Gruninger, M., Kifer, M., Martin, D., McGuinness, D., Parsia, B., Payne, T., Tate, A. & van der, W. 2004, *Semantic Web Services Language Requirements*, http://www.daml.org/services/swsl/requirements/swsl-requirements.shtml, 2007.

Huhns, M.N. & Singh, M.P. 2005, 'Service-Oriented Computing: Key Concepts and Principles', *IEEE Internet Computing*, vol. 1, no. 1.

Lara, R., Lausen, H., Arroyo, S., de Bruijn, J. & Fensel, D. 2003, 'Semantic Web Services: description requirements and current technologies', *Proceedings of the International Workshop on Electronic*

Commerce, Agents, and Semantic Web Services held in conjunction with the Fifth International Conference on Electronic Commerce (ICEC 2003), Pittsburgh.
Liu, W., Husni, H. & Padgham, L. 2007, 'E-Service Composition Tools from a Lifecycle Perspective', in J. Lu, D. Ruan & G. Zhang (eds), *E-Service Intelligent*, Springer.
Lu, L., Zhu, G. & Chen, J. 2004, 'An Infrastructure for E-government Based on Semantic Web Services', paper presented to the *IEEE International Conference on Services Computing*.
Lytras, M.D. 2006, 'The Semantic Electronic Government: Knowledge Management for citizen relationship and new assessment scenarios', *Electronic Government*, vol. 3, no. 1, p. 13.
Madhusudan, T. 2006, 'A web services framework for distributed model management ', *Information Systems Frontiers*, no. 9, pp. 9–27.
McGuinness, D.L. 2004, *OWL Web Ontology Language Overview* <http://www.w3.org/TR/2004/REC-owl-features-20040210/ >.
Medjahed, B., Rezgui, A., Bouguettaya, A., Ouzzani, a.M. & Tech, V. 2003, 'InfrastructureFor E-governmant webservices', *IEEE Internet Computing*.
Meneklis, B., Kaliontzoglou, A., Douligeris, C. & Polemi, D. 2005, 'Engineering and Technology Aspects of an e-Government Architecture Based on Web Srvices', *Proceedings of Third European Conference on Web Services*, 2005.
Nuseibeh, B. & Easterbrook, S. 2004, 'Requirements Engineering: A Roadmap', *University of Toronto*.
Pan, J.Z. 2007, 'A Flexible Ontology Reasoning Architecture for the Semantic Web', *IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING*, vol. 19, no. 2.
Peng, Z., Yanzhang, W. & Xuehua, W. 2006, 'Research on the Integration in E-government Based on Multi-Agent', *Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, IEEE.
Peristeras, V. & Tarabanis, K. 2006, '“Reengineering the public administration modus operandi through the use of reference domain models and Semantic Web Service technologies', paper presented to the *AAAI Spring Symposium, The Semantic Web meets eGovernment (SWEG)*, Stanford University, California, USA.
Sabou, M., Wroe, C., Goble, C. & Stuckenschmidt, H. 2005, 'Learning domain ontologies for semantic Web service descriptions', *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 3, no. 4, pp. 340-365.
Sanati, F. & Lu, J. 2007, 'A Methodological Framework for E-government Service Delivery Integration', paper presented to the *eGovernment Interoperability Campus*, Paris.
Sanati, F. & Lu, J. 2008, 'Semantic Web for E-Government Service Delivery Integration', paper presented to the *Information Technology Next Generation (ITNG)* Las Vegas USA, April 7-10.
Stojanovic, L., Stojanovic, N. & Apostolou, D. 2006, 'Change management in e-government: OntoGov case study', *Electronic Government*, vol. 3, no. 1, p. 19.
Thayer, R.H. & Dorfman, M. 1977, *Requirements Engineering*, Second Edition edn, IEEE Computer Society Press, Los Alamitos.
Trochidis, I., Tambouris, E. & Tarabanis, K. 2006, 'Identifying Common Workflow Patterns in Life-Events and Business Episodes', *Proceedings of*
The second International Conference on e-Government.
Trochidis, I., Tambouris, E. & Tarabanis, K. 2007, 'An Ontology for Modeling Life-Events', *Services Computing, 2007. SCC 2007. IEEE International Conference on*, pp. 719-720.
Trochim, W.M.K. 2001, *The research methods knowledge base*, 2nd ed edn, Automic Dog Publishing.
Umapathy, K. & Purao, S. 2007, 'A theoretical investigation of the emerging standards for web services', *Information Systems Frontiers*, no. 9, pp. 119–134.
Vassilakis, C., Sotiropoulou, A., Theotokis, D. & Gouscos, D. 2005, 'A Blackboard-Oriented Architecture for e-Government Service Composition', paper presented to the *IRMA International Conference*, New Orleans, Louisiana, USA.
Wolf, P. & Krcmar, H. 2008, 'Needs Driven Design for eGovernment Value Webs', *Hawaii International Conference on System Sciences, Proceedings of the 41st Annual*, pp. 220-220.