E-GOVERNMENT SERVICE INTEGRATION AND PROVISION USING SEMANTIC TECHNOLOGIES

Karol Furdík[1], Ralf Klischewski[2], Marek Paralič[3], Tomáš Sabol[3], Marek Skokan[3]

Abstract. The paper presents an approach to the integration of governmental services of various types, i.e. on-line (electronic and web services), as well as off-line (i.e. traditional, face-to-face) services by means of enhancing the service description with semantic information. A software platform, developed within the EU funded Access-eGov project, is described as a framework for creation and maintenance of semantically enriched services and for presenting the services to the citizens via customisable web interface. This paper presents results of testing and evaluation of the implemented Access-eGov platform in real settings within public administration in three EU countries (Slovakia, Poland, Germany) and subsequent technology enhancements.

1. Introduction. Motivation and related research

The importance of the semantically enabled interoperability of the government services was emphasised many times as a key challenge and crucial progress factor in the field of eGovernment [6]. Several initiatives and frameworks exist for central government systems (e.g. EIF IDABC [5], e-GIF [3], etc.), but a solution suitable for local eGovernment applications on the level of regions or municipalities is still challenging [7], [8]. The problems on the semantic data level may be caused by an absence of unified data models directly applicable to local administration entities [7], despite the availability of general semantic metadata standards. On the information level, the modelling of workflow sequences of provided services and maintenance of the corresponding presentation of the services towards the citizens is often a difficult task for public administration employees. Finally, the employment of advanced solutions supporting the semantic interoperability usually requires changing completely (or at least significantly) an existing infrastructure on the public administration, which is often expensive and time-consuming process. Existing solutions are mostly oriented on so-called *back-office integration* [10], which is rather difficult to set up, requires special knowledge on semantics or process modelling (as BPMN in [8] or maintenance of a complex semantic system in [1]).

The FP6 IST project Access-eGov (www.accessegov.org) aims to overcome the difficulties mentioned above by providing a lightweight system for *front-office integration* of the government services on a semantic basis. The integration on the user level is not invasive and does not require any costly changes to the existing infrastructure of a public administration [10]. Services of all the types, i.e. on-line electronic or web services, as well as off-line "face-to-face" services, remain in the existing state; they are only enhanced by a semantic metadata within the Access-eGov platform.

[1] InterSoft, a.s., Florianska 19, Kosice, Slovakia, karol.furdik@intersoft.sk
[2] German University in Cairo, Egypt, ralf.klischewski@guc.edu.eg
[3] Technical University of Kosice, Kosice, Slovakia, {tomas.sabol, marek.skokan, marek.paralic}@tuke.sk

The *Life event approach* [9] was adopted in the Access-eGov project as a method for modelling the government processes from the perspective of users, information consumers - citizens and businesses. The presentation layer was designed as a user-centred web-based application that allows browsing the life events and sub-goals relevant to a concrete business episode of life situation of users. The *requirement-driven approach* [7], a systematic method for building semantic structures based on the needs required by the information consumers, was used to design the underlying semantic structures for generating the presentation layer. Combination of these approaches in the Access-eGov platform enables to separate the inner service layer from the presentation layer. It even makes the maintenance of the platform easier and transparent. The methodology of the requirement-driven approach can be used for further updates of the presentation layer, while for the modifications on the level of services the platform provides a specialised editor, i.e. the Annotation Tool (AT).

After testing an initial version of the system developed within the first trials from October 2007 to January 2008 [2], [4], several requirements towards the improvement of user interface and system behaviour have arisen from the users. These requirements, together with the functionality planned for the second trials, invoked quite significant modifications and resulted in the second version of the system, which is presented in this paper. The following sections contain a description of the architecture, functionality, and technology used for the enhanced Access-eGov system, following by the presentation of the pilot applications accomplished in the second trials together with an evaluation of the achieved results.

2. Architecture and functionality

The architecture of the Access-eGov system is built on the WSMO framework (www.wsmo.org). The original WSMO conceptual model was, however, enriched by the workflow extensions that are capable to represent a process model of the interactions with human actors in the eGovernment domain [11], as it is required by the paradigm of the life-event approach. The updated conceptual model contains the *life event* and *goal* concepts as semantic representations of the needs and requirements of the users that are displayed as a customisable interface on the presentation layer.

The *service* concept, as a generalisation of the original WSMO *WebService* concept, represents a reference to an external service or data resource that is exposed to the users. Semantic representation of the referenced services is accompanied with the *service profile* and *service template* concepts, which specify the *non-functional properties* and pre-defined *interfaces* - required inputs, provided outputs, capabilities, and effects. The service instances have no direct references to the respective goals; instead, for a given goal, a semantic matching of service's capability interfaces against the interface of the goal dynamically retrieves a set of suitable services. Ontology concepts, instantiated as variables of a goal's interface, are customised and resolved by previous actions in the workflow. Complex goals are decomposed into atomic sub-goals with specified ontology instances representing variables in their interfaces. These instances are semantically mediated against the service ontology and used for the service discovery. More details about the technical implementation of process model and semantic matching procedure can be found in [11] and [12]. This dynamic semantic matching enables to separate the life events and goals, as objects that are visualised for citizens, from the inner services. Such a separation is of advantage for further data updates and for the overall system maintenance.

The Access-eGov system architecture, schematically depicted in Figure 1, consists of four main functional modules:

- *AeG resource ontology*, a persistent data repository and a knowledge base that contains WSML representations of the life events and goals. In addition, it contains generic service concepts and service templates that enable the service annotation, as well as the instances of already annotated services.
- *AeG core components* module, which includes inner business logic of the system. The components are responsible for decomposition of a given life event or goal into sub-goals, for orchestration, composition, and mediation of the sub-goals within a workflow thread, for semantic matching and discovery of the services for a given goal, as well as for execution of the retrieved and resolved services.
- *Annotation tool* (AT) for the semantic description (i.e. annotation) of the services that are to be integrated by the Access-eGov system. The web-based interface allows information providers to specify the non-functional properties for various service types, including traditional face-to-face services (in this case, the service is described by an explanatory HTML text that is presented to citizens), electronic, and web services. Capability interfaces, required inputs and provided outputs, and related workflow sequences are determined by a service template used during the annotation. Resulting WSML representations of the annotated service instances are stored in the resource ontology.
- *Personal Assistant client* (PAC), a tool that enables the citizens to browse and navigate through the life event and corresponding sub-goals. This web-based tool is implemented as a kind of wizard that enables to personalise and customise the thread of sub-goals by answering a set of customisation questions, which can be defined in a process model of the semantic representation of corresponding life events and sub-goals.

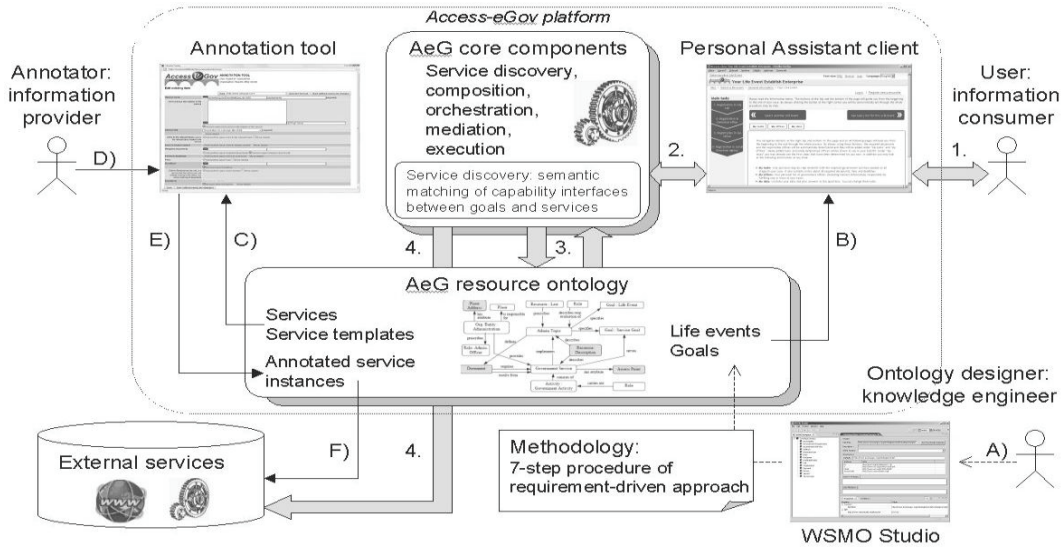

Figure 1: Architecture and control flow within the Access-eGov platform

Operations performed during the design time on the side of information provider (i.e. a public administration) are represented in Figure 1 by thin arrows and are labelled by capital letters. An ontology designer uses the WSMO Studio tool (www.wsmostudio.org) to create the resource ontology and customise it according to a given application case (A). The steps of the requirement-driven approach [7] can be employed to specify the life events and goals, as well

as the services and service types (templates). Existing ontology resources and standards, available in external ontology libraries or projects (some of them were described in [4]), can be reused to ensure the compatibility with other eGovernment solutions. The structure of life events and goals is then automatically populated to the PAC to be presented for information consumers (B). However, the services that should correspond to particular goals need to be created separately, by means of semantic annotation (steps C-F).

The structure of generic services and service templates, created in the resource ontology, is automatically populated to the AT (C). An annotator then uses the AT to semantically describe the services, i.e. to specify concrete values for particular non-functional properties, defined by the employed service template (D). A WSML representation of the annotated services is created automatically and is uploaded into the resource ontology as a set of service instances (E). The service instances may contain a reference to an external web service or to an existing web content (i.e. a portion of a web page). This reference is specified as a non-functional property during the step D. After uploading the service instance to the ontology, the reference is evaluated, the external resource is validated, invoked, and the returning data are set as default value for the service instance (F).

Operations performed by the information consumer during the run time are numbered and represented in Figure 1 by thick arrows. A citizen uses the PAC to browse the life events and goals (1). Some of the goals may require an additional input that concretises the citizen's needs - then the citizen provides answers to the customisation questions. The core system evaluates the responses obtained from the citizen for a given goal and dynamically creates a new thread of sub-goals, which is then returned back to the PAC (2). The process model of the goal is modified by the provided answers and its evaluation includes the procedures as service discovery, composition, orchestration, mediation, and execution. The core system communicates with the ontology to decompose a complex goal to sub-goals, to orchestrate, mediate, and compose the sub-goals into a workflow thread (3). For atomic goals that can not be decomposed to sub-goals, the semantic matching procedure is used to discover and dynamically resolve a set of proper services. The core system then transforms the resolved services, according to their type, to an executable form and invokes the referenced external services (4). The input values for the external services are populated from the input provided by users and/or calculated during the evaluation of the goal's process model (step 2). The output values provided by the invoked external services are returned to the process model, which is then modified accordingly and is presented to the citizen in the PAC (step1).

The Access-eGov platform in the scope of the described functionality was tested within the first trials on three pilot applications in Germany [2], Poland, and Slovakia. The testing was carried out from October 2007 to January 2008. Evaluation of the achieved results [4] demonstrated feasibility of the proposed solution as a platform for front-office service integration. However, several issues were identified and requested as important for further improvement. Significant enhancements were especially required on the graphical user interface of the PAC [2]. Easier navigation and browsing between the goals (tasks) accompanied with proper explanatory texts was identified as the main concern for users. To address the required changes, complete structural and visual overhaul was accomplished on the PAC. A screenshot of the enhanced user interface is presented in Figure 2.

Major changes implemented for the second trials include simplification of the left bar, which currently presents only top level elements (goals, tasks) together with an indication of state (Completed / Active / Future) for the goals. Navigation was enhanced by introducing guide buttons that enable to proceed further in the customisation workflow (i.e. the button ***Continue***

in Figure 2) or return to the previous step (the button *Leave personalization*). The tabs as *My tasks*, *My offices*, *My data* were included to provide a global survey of the goals, access points, and customisation answers entered in the scope of the whole life event during a single session. The user management functionality was introduced to enable the citizens to register and obtain a permanent user account that includes user preferences and personal data. Finally, a full-text searching capability was added to ease a selection of alternative services or offices.

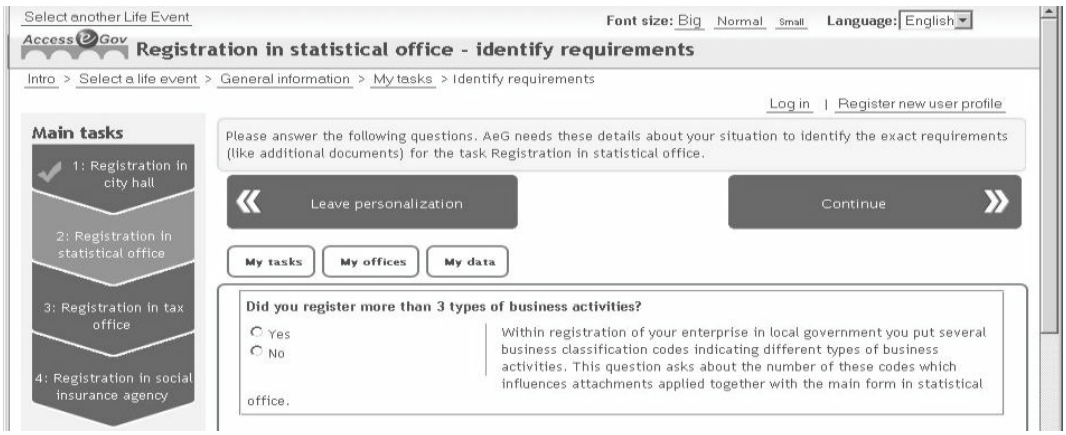

Figure 2: Personal Assistant client interface, customisation of a sub-goal

The enhancements of the PAC invoked the necessity to adapt the core components accordingly, even though the data structures and general functionality have not been fundamentally changed. The extensions that were required for the second trials (cf. section 3) have also influenced the updates of the inner components. Web service invocation was fully integrated into the Access-eGov platform. It is based on the specification of semantic annotation for WSDL (www.w3.org/TR/sawsdl/), which defines how to annotate the WSDL files that are published for the web services with the references to a knowledge model. The Access-eGov implementation primarily supports the SOAP/HTTP communication protocol; data transformations are handled by XSLT. This implementation was additionally extended with the support of transformation of semantic data into the HTML fragment using the same XSLT mechanism. This transformation is used in the PAC for presentation of the service results on the web user interface.

A web service interface for the data repository of the Access-eGov platform was implemented. This interface allows remote reading and writing of all the data elements stored in the resource ontology, i.e. the life events, goals, and published (annotated) services. Additionally, an inner component for resolving the goals was also wrapped as a web service; it enables an external application to find and access directly all the services registered in the Access-eGov platform for the specified goal (and specified information context). This web service interface can be used to implement the Access-eGov semantic registry of the traditional and electronic services for external applications.

3. Second trials of the pilot applications

The updated Access-eGov platform was tested from December 2008 to January 2009 within the second trials on the same pilot applications as for the first trials. However, the main differences were in the extension and scale of the covered area as well as in the ability of the system to integrate the external web services and data resources. The second trials were aimed

at proving the system functionality and behaviour on real-world settings, especially focusing on the interoperability of heterogeneous distributed services and information resources.

The *German pilot application* took place in the federal state of Schleswig-Holstein and was dealing with the scenario "Getting married" [2]. This application case was chosen as a prototype example, but the goal was to integrate the services of the different administrations of 1,120 municipalities located in the federal state. The task was to integrate the different web resources containing the service information and to make these accessible via a single platform but still leaving the data and its maintenance in the administrations' legacy systems. The second trial was especially focused on the semantic annotation of the services by metadata (i.e. contacts, opening hours, etc.) grabbed from the web sites of involved municipalities or provided as web services. The trial evaluation showed that integrating data from heterogeneous municipal web sites by adding semantic metadata to them and then grabbing it, requires a lot of effort. Specific tools for different content management systems (CMS) have to be developed and annotations have to be added manually in the relevant context. This effort can only be minimised if the data is already provided in a well-structured form in the CMS and if annotations can be partly or completely automated this way. Creating such annotations is still an advantage for the municipalities though, if the annotations can be used by several semantic web applications.

The web service interface has been tested by the IT service provider of another municipality in the field test. This test required mapping their internal information structures to the structures required by the web service interface and the underlying Access-eGov information architecture. This approach was more efficient because the structures have to be mapped only once and data can then be sent to the interface whenever it needs to be updated. No laborious manual annotation is required in this case. However, this approach is only possible if data is already provided in a well-structured form.

Furthermore, in the second trial, some improvements regarding the usability and stronger improvements regarding the believability of the PAC have been achieved. Think-aloud sessions, in which users were observed during their work with the system, also proved that navigation was facilitated in the revised system and that users had less problems understanding the provided content due to a better structuring of the process on the web site.

The *Polish pilot application*, focusing on the "Establishing a new enterprise" scenario, ran in the Silesian region around the city of Gliwice. Main aim of this pilot application was to provide a comprehensive and user-friendly interface that integrates all the relevant information (provided in Poland mostly in a form of traditional services) of rather complex process on one place, guides the citizens and businesses through the life event and related sub-goals. The "Establishing an enterprise" life event, modelled in this pilot, included sub-goals of registration procedure on various institutions as, for example, the local government, statistical office, tax office, and social insurance agency. In the run-time, the process is customised by answering questions on size, location, and type of the enterprise, etc., that constraints the selection of sub-goals and documents required from user as well as documents issued by institutions to user in the workflow thread of the life event.

Generally, the aim of the second trial was to cover the expected improvements coming from the evaluation of the first trial. In particular, the resource ontology was significantly updated by modelling additional business activities, as a reaction on the changes of the relevant Polish law. Moreover, the ontological model of the TERYT (i.e. Polish National Official Register of the Territorial Division of the Country) was implemented and integrated into the AT.

The *pilot application in Slovakia* covers the area of Kosice Self-governing Region (KSR) and the municipality of Michalovce as its part. This pilot application was aiming to support citizens during the process of obtaining permits for building a house, including services related to the land-use planning and final approval proceedings. The "Get a Building Permit" life event was modelled to provide for citizens a personalised guidance through the whole process, often very complex and difficult to comprehend even for laymen.

The second trial was designed to cover an extended area of Kosice and Michalovce with around 350,000 inhabitants. All five branches of the main construction office of KSR were engaged in the testing. The process for the second trial was enhanced by a construction supervision phase, modelled in the ontology. New service types as real estate agencies, project designers, and architects were included and respective services were annotated. Altogether, more than 200 of newly annotated services were created for the second trial. Another extension of the second trial was a creation of a new web service for online checking and provision of the land-use plan for a given municipality. Service integration with the Slovak cadastre portal (www.katasterportal.sk) was also tested but consequently it was excluded from the pilot due to the scalability restrictions on the portal side.

4. Evaluation of results

The results of the second trials were collected for all the pilots and evaluated during February 2009. To be able to identify a progress in the development, the same methods as in the first trials were used for technical testing, collecting feedback from the public, and evaluation [2]. Technical testing was focused on the correctness, speed, and overall performance of the platform. Results of all the performed technical tests indicate that the improvements on the inner core components for data repository management and service discovery increased the system response significantly (up to 4 seconds even for the most complicated requests on fully featured ontology structure).

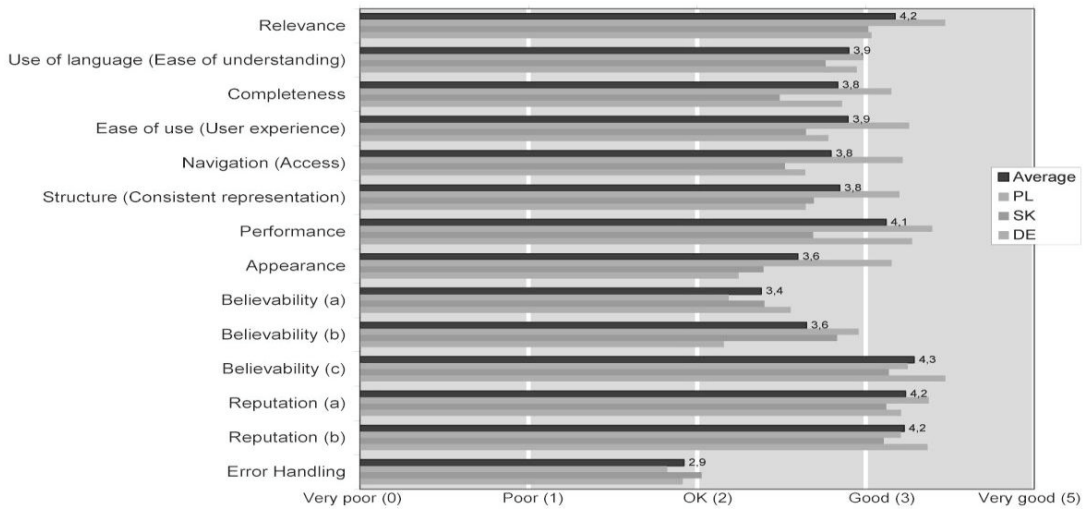

Figure 3: Overall results of online questionnaire

Most of the criticism of users during the first trials was towards the user interface, i.e. the usability of the PAC. It is obvious, since this tool is a front-end and "the most visible" component of the platform. It was the reason why the PAC's user interface was completely redesigned and enhanced according to the feedback obtained from users (cf. section 2). A

chart presenting the results collected from user's answers on the online questionnaire regarding the usability of the updated PAC is depicted in Figure 3. The levels of believability include a) ability to identify a provider (author) of the presented information, b) ability to determine which of the presented links lead to an external web site, and c) overall correctness of the provided information. The reputation covers a) a conviction to take the Access-eGov as a good (relevant, significant) source of information, and b) trustfulness of the information provided by the PAC.

The usability evaluation of the second trial was quite successful, since average results of all the investigated aspects were better than those achieved in the first trial. Biggest improvement, in comparison with the first trials, was achieved in believability (increased by 15%) and in completeness of presented information (10%). Moreover, the people involved in the testing expressed rather positive feedback, stating that the solution is easy-to-use and provides very useful information for the modelled life events. In addition to the interface usability, a serious concern was given during development also on the accessibility criteria of the user interface. The updated PAC conforms to the "single A" level of the W3C WCAG eAccessibility criteria (www.w3.org/TR/WCAG20/).

Acknowledgements

The Access-eGov project No. FP6-2004-27020 is co-funded by the European Commission.

References

[1] Apostolou, D. et al, Towards a Semantically-Driven Software Engineering Environment for eGovernment, in: TCGOV 2005, LNAI 3416, 2005, pp. 157-168

[2] Bednar, P. et al, Semantic Integration of eGovernment Services in Schleswig-Holstein, in: Electronic Government, 7th International EGOV Conference, Springer, LNCS 5184, 2008, pp. 315-327

[3] Cabinet Office, e-Government Interoperability Framework, http://www.govtalk.gov.uk

[4] Furdik, K., Mach, M., Sabol, T., Practical Experiences with Enhancing Semantic Interoperability in eGovernment using WSMO, in: MeTTeG 2008, Halley Editrice, 2008, pp. 23-35

[5] IDABC e-Government Observatory: European Interoperability Framework for pan-European e-Government Sevices. Version 1.0, 2006, http://ec.europa.eu/idabc/

[6] Interoperability for Pan-European eGovernment Services. Commission of the European Communities, COM(2006) 45 final, Brussels, February 13, 2006

[7] Klischewski, R., Ukena, S., Designing semantic e-Government services driven by user requirements, in: Electronic Government, 6th International EGOV Conference, Proceedings of ongoing research, project contributions and workshops, Trauner Verlag, Linz, 2007, pp. 133-140

[8] Koussouris, S. et al, Building a Local Administration Services Portal for Citizens and Businesses: Service Composition, Architecture and Back-Office Interoperability Issues, in: Electronic Government, Springer, LNCS 4656, 2007, pp. 434-437

[9] Leben, A., Vintar, M., Life-Event Approach: Comparison between Countries, in: Electronic Government, Springer LNCS 2739, 2003, pp. 434-437

[10] Mach, M., Sabol, T., Paralic, J., Integration of eGov services: back-office versus front-office integration, in: ESCW 2006, Budva, Serbia - Monte Negro, 2006, pp. 48-52

[11] Skokan, M., Bednar, P., Semantic orchestration of services in eGovernment, in: Proceedings of Znalosti (Knowledge) 2008, STU Bratislava, 2008, pp. 215-223

[12] Skokan, M. et al, Integration of governmental services in semantically described processes in the Access-eGov system, in: Proc. of the ECOM&EGOV-08 conference, Wisla, Poland, 2008, pp. 415-419