Semantic Web in e-Government

Tomas Sabol
Faculty of Economics, Technical University of Kosice,
B.Nemcovej 32, 042 00 Kosice, Slovak Republic
Tomas.Sabol@tuke.sk

Marian Mach
Dept. of Cybernetics and Artificial Intelligence, Technical University of Kosice,
Letna 9, 042 00 Kosice, Slovak Republic
Marian.Mach@tuke.sk

Abstract. *In order to improve efficiency and quality of services provided to the public (citizens and/or businesses), public administrations are constantly trying to turn their services into electronic service-oriented e-Government applications. Unfortunately, achieving service integration and interoperability in the domain of government services is quite a challenging task, since the domain is unique because of manifold horizontal as well as vertical semantic barriers caused by different interpretations and views. The emerging idea of the Semantic Web, and particularly Semantic Web Services, seems to offer a promising technology to overcome semantic fragmentation of the domain of public administration services.*

In this paper we argue for the employment of the Semantic Web technologies in the e-Government domain, introduce the current state of the research in the area, give a short overview of EU R&D projects in this area, and present a novel approach to e-Government service integration represented by the Access-eGov EU-funded project.

Keywords. e-Government, interoperability, web services, Semantic Web, ontology

1. e-Government

As the Communication of the European Commission on "The Role of e-Government for Europe's Future" states: *The public sector plays a very important role in Europe's social and economic model by supporting high levels of welfare for citizens, ensuring socio-economic cohesion and supporting the functioning of a competitive market environment.*. Just for illustration: government revenues add up to 45% of the EU's GDP. The public sector is the single largest purchaser in the economy. Public consumption amounted to 20.6% of GDP in 2002, up from 19.9% in 1998. Public administrations are also a major provider of services to business, influencing enterprises' ability to compete, and of services to citizens, enabling them to obtain education and training and to find jobs. The counterpart of ensuring the delivery of many public services is found in obligations on enterprises and citizens such as those relating to social security, environmental reporting and tax collection. The cost of administrative obligations is estimated at 2-3% of GDP and falls disproportionally on SMEs. E-Government is expected to make such activities more efficient – to cut red tape – and more effective – accessible, user-friendly, secure, targeted – which should boost economic growth throughout the economy as a whole.

However, it seems that public sector in EU is today at a crossroads, facing challenging economic and social conditions, institutional change and the profound impact of new technologies. Expectation is growing that, as it is a major economic actor for boosting economic growth, the public sector will play a strong role in realising the Lisbon strategy for economic, social and environmental renewal. Within the public sector, public administrations are facing the challenge of improving the efficiency, productivity and quality of their services.

On the other hand, there are clear indicators that ICTs are a powerful driver of growth and employment. A quarter of EU GDP growth and 40% of productivity growth are due to ICT . Differences in economic performances between industrialised countries are largely explained by the level of ICT investment, research, and use, and by the competitiveness of information society and media industries. ICT services, skills, media and content are a growing part of the economy and society. Expectations from ICTs in public sector are thus very high. And here e-Government comes into play.

The term e-Government (like e-Business, e-Learning) is associated with the dot com revolution of the start of the decade and has focussed on use of information and communication technologies (ICTs) and especially the Internet in public sector . Similarly as in e-business, there exist several models – for conducting business between government and citizen (G2C), government and businesses (G2B, B2G), or between different governmental institutions (G2G). There are several definitions of e-Government: "*the use of information and communication technology in public administrations combined with organisational change and new skills in order to improve public services and democratic processes and strengthen support to policies*" (EC, Communication of 2003 COM(2003)567), "e-Government is a government that applies ICT to transform its internal and external relationships" (UN), the use of ICT, and particularly the Internet, as a tool to achieve better government" (OECD).

Potential benefits of e-Government to citizens, businesses and public administration (PA) institutions can be summarised in the following points:

- Increased accessibility – for users (citizens and businesses), they can interact with government through new channels (Internet, information kiosks, mobile devices) and new ICT-enabled services routes (intermediaries, one stop shops, call centres etc.);
- Flexibility – the ability to interact at more convenient times, providing new types of services;
- Efficiency – more efficient government leading to better services and better use of available resources (doing more for less);
- Inclusion – the ability to reach a greater percentage of the target population (including disadvantaged groups – the elderly, handicapped, unemployed, people from rural areas etc.).

According to e-Government is currently in its third wave of development, where individual waves can be described as follows:

1) Wave 1: Promote access and development – new ICT introduced new capabilities and the opportunity for e-Government. The focus was on developing infrastructure and adapting the legislative framework in order to encourage adoption.
2) Wave 2: Provide services online – the primary emphasis was on developing customer interfaces to make existing services available online.
3) Wave 3: Transform the enterprise – the focus has changed from basic provision of services to impact, in terms of benefits to end users and government. An increasing emphasis is being put on back office automation and integration (between and within services). This is where most governments currently are (and some still delayed). Within the first part of this wave e-Government is mostly focused on automation of existing processes. In the second half of this wave governments are re-engineering their business (organisational and information processes) and subsequently restructuring the organisation itself.
4) Wave 4: Next generation government – governments move beyond the re-engineering of existing services to radically new ways of realising policy objectives enabled by ICT. Traditional channels are ICT-enabled in the back office and offered alongside electronic channels in a seamlessly integrated fashion. Re-organisation is radical and across organisational boundaries, the extent of cross-organisational restructuring spans governmental entities as well as the private and third (NGO) sector.

The third and fourth wave are sometimes referred as *t-Government* (transformative government) – using ICT to enable the business transformation of government, focused on customers. T-Government is characterised by three basic principles:

- t-Government is about transformation – changing fundamentally the way government works. It is more than moving services online.
- Transformation should be business-led and ICT-enabled. Technology should be neither the end nor the sole means of public service transformation.
- Transformation should have clear benefits for the customer (citizen, business, public servant).

E-Government Action Plan - within the recent major Information Society development initiative of the EU "i2010 – A European Information Society for Growth and Employment" - calls for better economy as a competitive must in global economy, because countries that score high on public-sector openness and efficiency and e-Government readiness are also top on the economic performance and competitiveness scoreboards. The Action Plan focuses on five major objectives for e-Government with specific objectives for 2010:

- No citizen left behind - advancing inclusion through e-Government so that by 2010 all citizens benefit from trusted, innovative services and easy access for all;

- Making efficiency and effectiveness a reality – significantly contributing, by 2010, to high user satisfaction, transparency and accountability, a lighter administrative burden and efficiency gains;
- Implementing high-impact key services for citizens and businesses - by 2010, 100% of public procurement will be available electronically, with 50% actual usage, with agreement on cooperation on further high-impact online citizen services;
- Putting key enablers in place - enabling citizens and businesses to benefit, by 2010, from convenient, secure and interoperable authenticated access across Europe to public services;
- Strengthening participation and democratic decision-making - demonstrating, by 2010, tools for effective public debate and participation in democratic decision-making.

It has to be noted that those far reaching goals of e-Government mentioned above are still far away from 100% fulfilment. Expectations that e-Government would reduce the cost of service delivery have not materialized yet due to the early stage nature of most online government services and the lack of integration between the front end and back office systems. Interoperability in general has been identified as one of the main challenges of the future development.

Summarising the requirements on e-Government applications – flexible, user-friendly, integrated, interoperable, intelligent, cross-border/pan-European services – we can easily come to the conclusion that semantic technologies and artificial intelligence applications (e.g. Semantic Web, web services, (ontology-based) knowledge modelling, agent technology, knowledge management etc.) can play a major role in achieving these ambitious goals. But before going to the next, more technology-focused sections of this paper, we should remind the reader that "successful e-Government must do more than just use ICT and put administrative services on the Internet; successful e-Government implies reengineering administrative processes, reorganising and restructuring public organisations and shifting the focus towards a citizen and customer-oriented service provision" . To put it in other words, efficient use of ICT is necessary, but not sufficient condition, holistic approach is a *conditio sine qua non*.

In every case, expectations are high, e.g. vision of "the digital semantic government eco-system" outlined by participants of the "e-Government 2020. FP7 Research Consultation workshop" (Brussels, 26-27 October 2005) was quite ambitious (which is not necessarily the same as desirable): "*In 2020 information and knowledge will be virtually stored in 'cyberspace' and will be available anytime and from anywhere. There will be digital agents in the semantic eco-systems – such as the digital semantic person, the digital semantic enterprise or the digital semantic politician and the digital e-Government presence. Each of these agents will carry a magnitude of information. Technologies will have to be developed for collecting, storing and managing this information. Each of these agents also has different characteristics and roles. The digital semantic person will hold digital personal information: it will posses an architecture which will enable it to access its own data. Proactive governmental agencies will be able to provide the digital semantic person with highly personalised services. However, a semantic person will also be part of social groups (of other digital persons). This raises questions with respect to social roles and participation in community activities. The management of identity, anonymity and pseudonymity will have to be tackled …*"

2. Semantic Web

WWW (World Wide Web) represents a huge repository of information which can be retrieved and utilised (if user is lucky enough to find what he/she needs – but it is another story beyond the scope of this paper). Its success turns it into a phenomenon which in eyes of many people plays the role of the synonym of the Internet. Unfortunately, information is represented with no meaning associated – the meaning of retrieved information can be (re-)established only in the process of interpreting the information by humans. As a result, information scattered throughout the current (traditional) version of the web is almost totally useless for software, non-human users (machine agents).

In attempt to respond to this situation, the term "Semantic Web" was coined by Tim Berners-Lee and his colleagues referring to a "web for machines" as opposed to a web to be read by humans. In their understanding "*The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation.*"

The Semantic Web is the opportunity for providing, finding and processing information via the Internet with the help of machines (and mostly also for machines) which are capable of dealing with the semantics of the information. The idea is to transform information into something meaningful to actors who seek to enhance their knowledge in order to satisfy a specific concern or accomplish a specific task related to their particular context. The vision of the Semantic Web is based on the employment of semantic technologies that allow the meaning of information and the meaning of associations between information to be known and processed at execution time.

To fulfil the promises and enable semantic technologies to work, there must be a knowledge model (of some part) of the world that is used to provide meaning to information to be processed within an application. The knowledge model has the form of a semantic model which differs from other kind of models :

- Meaning is represented through connectivity. The meaning of terms, or concepts, in the model is established by the way they connect to each other.
- A semantic model expresses multiple viewpoints.
- Semantic models represent knowledge about the world in which systems operate and are shared across applications.
- Several interconnected models could be used to represent different aspects.
- Use of a model is often referred to as "reasoning over the model". The reasoning can range from a very simple process of graph search to intricate inferencing.

Although the role of a semantic model can be played by a simple taxonomy, nowadays use of semantically richer ontologies (ontological models) dominates.

Although most common definition states that "*An ontology is a specification of conceptualisation*", more detailed definitions can make things a bit clearer. One of them states that "*The subject of an ontology is the study of the categories and things which exist in some domain. The product of such study, called an ontology, is a catalogue of the types of things that are assumed to exist in a domain of interest from the perspective of a person who talks about the domain using some language*". From the practical point of view, an ontology is a network of connections defining explicit relationships (named and differentiated) between concepts. New knowledge can be derived by examining the connections between concepts. Simple ontologies are just networks of connections, richer ontologies include rules and constraints governing these connections.

The Semantic Web is not so much a technology as an infrastructure, enabling the creation of meaning through standards, mark-up languages, and related processing tools. To represent ontologies in a formal way, several languages can be used. The most common ontology languages are briefly described below (all the presented languages are supervised by the World Wide Web Consortium).

XML was widely accepted and used as a convenient information representation and exchange format. XML itself does not carry semantics, but is serves as the base syntax for the leading ontology languages. Later additions like XML-DTD (Document Type Definition) and XML-Schema, added some syntactic rules like enumerations, cardinality constrains, and data types, but still lacked even simple semantics like inheritance.

RDF (Resource Description Framework) is a standard way for defining simple descriptions. RDF supports semantics - a clear set of rules for providing simple descriptive information. RDF enforces a strict notation for the representation of information, based on resources and relations between them. The RDF data model provides three object types: resources, properties, and statements. Resource may be an object; property is a specific aspect, characteristic attribute, or relation used to describe a resource; statement is a triple consisting of two nodes and a connecting edge. The strength of the language is in its descriptive capabilities, but it still lacks some important features required in an ontology language such as inferences for example. However, ontology languages built on top of RDF as a representation and description format.

RDF Schema (RDFS) enriches the basic RDF model, by providing a vocabulary for RDF, which is assumed to have certain semantics. Predefined properties can be used to model instance of and subclass of relationships as well as domain restrictions and range restrictions of attributes. Indeed, the RDF schema provides modelling primitives that can be used to capture basic semantics in a domain neutral way. That is, RDFS specifies metadata that is applicable to the entities and their properties in all domains. The metadata then serves as a standard model by which RDF tools can operate on specific domain models, since the RDFS meta-model elements will have a fixed semantics in all domain models.

OWL is the newest W3C recommendation for ontology definition. OWL enhances RDF vocabulary for describing properties and classes: relations between classes (e.g. subclasses), cardinality, equality, richer typing of properties, characteristics of properties (e.g. symmetry) and instances.

OWL is quite a sophisticated language. The most important feature is its capability for description logic (DL) reasoning (Description Logics are a family of logic-based knowledge representation formalisms designed to represent and reason about the knowledge of an application domain in a structured and well-understood way). The OWL language also provides three increasingly expressive sublanguages: OWL Lite, OWL DL, and OWL Full, each offers a different level of expressiveness at the trade-off for simplicity, thus offering a suitable sub language parts available for use according to expressivity needs.

An important constituent of the Semantic Web is represented by inference engines. Their aim is to reason over ontological models to prove statements and/or to deduce new knowledge from already explicitly presented knowledge. They are expected to make explicit those facts that are present in the ontology only implicitly. The reasoning over ontology can have the following purposes:

- *Validation*. Validating ontology means ensuring that the ontology is a good representation of the domain of discourse that should be modelled. Reasoning is extremely important for validation. For example, checking whether an ontology is internally consistent is crucial: obviously, no inconsistent theory can be a good representation of domain.
- *Analysis*. In analysis one assumes that the ontology is a faithful representation of the domain, and tries to deduce facts about the domain by reasoning over the ontology. In a sense of trying to collect new information about the domain by exploiting the ontology. Obviously, analysis can also provide input to the validation phase.
- *Harmonisation*. Myriads of ontologies can be used within the Semantic Web environment. Since each ontology represents a particular point of view, using different ontologies to represent meaning of information within a domain of interest results in mismatches in understanding and dealing with this information. In order to avoid it, semantic mappings between ontologies must be done.

At the beginning, the idea of the Semantic Web tried just to enhance the current version of the web. It started out with a document oriented approach. The basic idea was to make web pages identifiable by computers as information resources carrying not only information (readable only by humans) but the meaning of this information as well.

The meaning was added by annotating these pages with semantic mark-up. Ontologies here define a shared conceptualization of the application domain at hand and provide the basis for defining metadata, that have a precisely defined semantics, and that are therefore machine-processable. The idea of semantically annotated web pages with machine-interpretable description of their content aimed at automated processes of searching and accessing pages enabling human users to better utilise information stored on the web.

In addition to human users, the Semantic Web enables the participation of non-human users as well. These machine agents do not need to deal with whole web pages. Instead of this, they exchange chunks of data with each other. Although they can communicate using different protocols, technology of web services has become a dominant way of communication with and using services of applications in the web environment.

Formerly, the problem of interoperability of different agents was tackled by translation technologies, most commonly by field to field mapping. The Semantic Web enables agents to exchange chunks of data with meaning associated to the data using semantic technologies. Advanced applications can use ontologies to relate the information to a semantic model of a given domain. In this way semantic technologies offer a new way to integrate different applications. Nowadays, the field of semantic interoperability is the most addressed problem connected with the idea of the Semantic Web.

2.1. Formalisms for modelling web services

OWL-S (Web Ontology Language for Services). OWL-S is OWL ontology for semantic description of the web services. The structure of the OWL-S consists of a service profile for service discovering, a process model which supports composition of services, and a service grounding, which associates profile and process concepts with the underlying service interfaces.

Service profile has functional and non-functional properties. Functional properties describe the inputs, outputs, preconditions and effects of the service (IOPEs). The non-functional properties describe the semi-structured information intended for human users for service discovery, e.g. service name, description and parameters which incorporate further requirements on the service capabilities (e.g. security, quality of service, geographical scope, etc.).

Service model specifies how to interoperate with the service. The service is viewed as a process which defines the functional properties of the service (IOPEs), together with details of its constituent processes (if the service is a composite service). The service model functional properties can be shared with the service profile. OWL-S distinguishes between atomic, simple, and composite processes. OWL-S atomic processes can be invoked, have no sub-processes, and are executed in a single step from the requester's point of view. The simple processes are used as elements of abstraction, they are viewed as executed in a single step, but they are not invocable. Composite processes consist of simple processes and define their workflows using control constructs, such as sequence, split, if-then-else or iterate.

Service grounding enables the execution of the web service by binding the abstract concepts of the OWL-S profile and process model to concrete messages and protocols. Although different message specifications are supported by OWL-S, the widely accepted WSDL is preferred.

WSMO (Web Service Modelling Ontology). WSMO is a conceptual model for describing Semantic Web Services. It consists of four major components: ontologies, goals, web services and mediators.

Ontologies provide the formal semantics to the information used by all other components. WSMO specifies the following constituents as part of the description of ontology: concepts, relations, functions, axioms, and instances of concepts and relations, as well as non-functional properties, imported ontologies, and used mediators. The latter allows the interconnection of different ontologies by using mediators that solve terminology mismatches.

Goal specifies objectives that a client might have when consulting a web service, i.e. functionalities that a web service should provide from the user perspective. In WSMO a goal is characterized by a set of non-functional properties, imported ontologies, used mediators, the requested capability and the requested interface.

A *web service* description in WSMO consists of five sub-components: non-functional properties, imported ontologies, used mediators, a capability and interfaces. The capability of a web service defines its functionality in terms of preconditions, post-conditions, assumptions and effects. A capability may be linked to certain goals that are solved by the web service via mediators. Preconditions, assumptions, post-conditions and effects are expressed through a set of axioms and a set of shared all-quantified variables. The interface of a web service provides further information on how the functionality of the web service is achieved. It describes the behaviour of the service from the client's point of view (service choreography) and how the overall functionality of the service is achieved in terms of cooperation with other services (service orchestration). A choreography description consists of the states represented by ontology, and the if-then rules that specify (guarded) transitions between states. The ontology that represents the states provides the vocabulary of the transition rules and contains the set of instances that change their values from one state to the other. Like for the choreography, an orchestration description consists of the states and guarded transitions. In extension to the choreography, in an orchestration transition rules, that have as a post-condition the invocation of a mediator that links the orchestration with the choreography of a required web service, can also appear.

Mediators describe elements that aim to overcome structural, semantic or conceptual mismatches that appear between the different components that build up a WSMO description.

WSMO is formalized using the Web Service Modelling Language (WSML) which is based on description logic, first-order logic and logic programming formalisms.

WSDL-S (Web Service Semantics). WSDL-S is a small set of proposed extensions to Web Service Description Language (WSDL) by which semantic annotations may be associated with WSDL elements.

WSDL-S defines URI reference mechanisms to the interface, operation and message WSDL constructs to point to the semantic annotations defined in the externalized domain models. WSDL-S defines the following extensibility elements and attributes:

- *modelReference* element - allows for one-to-one associations of WSDL input and output type schema elements to the concepts in a semantic model;
- *schemaMapping* attribute - allows for many-to-many associations of WSDL input and output complex type schema elements to the concepts in a semantic model. It can point to a transformation (for example XSLT) from XML data to the external ontological data in RDF/OWL or in WSML;
- *precondition* and *effect* elements - are used on WSDL interface operations to specify conditions that must hold before and after the operation is invoked. The conditions can be specified directly as an expression with format defined by the semantic language or by reference to the semantic model;
- *category* element - provides a pointer to some taxonomy category. It can be used on a WSDL interface and is intended to be used for taxonomy-based discovery.

BPEL4WS (Business Process Execution Language for Web Services). BPEL4WS is a specification that models the behaviour of web services in a business process interaction. It is based on the XML grammar which describes the control logic required to coordinate web services participating in a process flow. An orchestration engine can interpret this grammar, thus it can coordinate activities in the process. BPEL4WS is a layer on the top of WSDL (Web Services Description Language). WSDL defines the specific operations and BPEL4WS defines how the operations can be sequenced. Every BPEL4WS process can be considered as a web service using WSDL describing the public entry and exit points for the process. WSDL data types are used within a BPEL4WS process to describe the information that passes between requests. WSDL might be used to reference external services required by the BPEL4WS process. BPEL4WS provides support for both executable and abstract business processes. The executable process models a private workflow. The abstract process specifies the public message exchanges between parties. The executable processes provide orchestration support while the business protocols (abstract processes) focus more on the choreography of the services.

Support for basic and structured activities is included. The basic activities might be receiving or replying to message requests as well as invoking external services. The structured activities specify what activities should run in what order – the whole process flow. These activities also provide support for conditional looping and dynamic branching. The structured activities might specify that certain activities should run sequentially or in parallel. *Containers* and *partners* are two important elements within BPEL4WS. A container is a variable for exchange in the message flow. A partner could be any service that the process invokes or any service that invokes the process. Each partner is mapped to a specific role that it fills within the business process. This is managed by containers.

In BPEL4WS, a set of activities can be grouped into a single transaction – it means that the steps enclosed in the scope should either all complete or all fail. Within this scope, the developer can then specify compensation handlers that should be invoked if an error occurs. BPEL4WS provides a robust exception handling mechanism through the use of throw and catch clauses, similar to the Java programming language.

2.2. Frameworks and tools for Semantic Web Services

OWL-S Tools. A set of disparate OWL-S tools exists, but not a complete execution environment based on OWL-S concepts. Instead of it, the tools have to be integrated by user. The set includes editor, matchmaker and annotator (several additional tools exist).

OWL-S editor is divided into three main parts: creator, validator and visualiser. The creator enables to create an empty OWL-S description either from a template or through a wizard. The validator part serves for validating the URIs used in the OWL-S descriptions and also validate the syntax of the ontologies. The visualiser part enables the user to visualise the descriptions and service compositions in a graphical manner by exploiting UML activity diagrams.

DAML-S Matchmaker is a Web Service that helps make connections between service requesters and service providers. The Matchmaker allows users and/or software agents to find each other by providing a mechanism for registering service capabilities. It calculates the syntactical and semantic similarity among service capability descriptions. The matching engine of the matchmaking system contains five different filters for namespace comparison, word frequency comparison, ontology similarity matching, ontology subsumption matching, and constraint matching.

ASSAM (Automated Semantic Service Annotation with Machine learning) WSDL Annotator is an application that assists the user in annotating Web Services. Annotations can be exported in OWL-S. WSDL files can be annotated with an OWL ontology with a point-and-click-interface, but the key feature is machine learning assisted annotation.

WSMX (Web Service Execution Environment). WSMX is an execution environment which enables discovery, selection, mediation, and invocation of Semantic Web Services. WSMX is based on the conceptual model provided by WSMO, being at the same time a reference implementation of it. It is the scope of WSMX to provide a test bed for WSMO and to prove its viability as a mean to achieve dynamic interoperability of Semantic Web Services.

Nowadays, some modules are not implemented or have limited functionality. The main components that have been already designed and implemented in WSMX are: core component, resource manager, discovery, data and process mediator, communication manager, choreography engine, and web service modelling toolkit.

Core component is the central component of the system connecting all the other components and managing the business logic of the system. Resource manager manages the set of repositories responsible for the persistence of the WSMO and non-WSMO related data flowing through the system. Discovery component has the role of locating the services that fulfil a specific user request. This task is based on the WSMO conceptual framework for discovery which envisions three main steps in this process: goal discovery, web service discovery, and service discovery. Currently, the service discovery covers only the matching of user's goal against service descriptions based on syntactical consideration.

Two types of mediators are provided by WSMX to resolve the heterogeneity problems on data and process level. Data mediation is based on paradigms of ontology mappings and alignment with direct application on instance transformation. The process mediation offers functionality for runtime analysis of two given patterns (i.e. WSMO choreographies) and compensates the possible mismatches that may appear.

Communication manager through its two subcomponents, the receiver and the invoker, enables the communication between the requester and the provider of the services. Choreography engine has to provide a means to store and retrieve choreography interface definitions, to initiate the communication between the requester and the provider in direct correlation with the results returned by the process mediator, and to keep track of the communication state on both the provider and the requester sides.

The web services modelling toolkit is a framework for rapid creation and deployment of homogeneous tools for Semantic Web Services. An initial set of tools includes a WSML editor for editing WSML and publishing it to WSMO repositories, a monitor for monitoring the state of the WSMX environment, a data mediation tool for creating mappings between ontologies, and a management tool for managing the WSMX environment.

Even if the reasoner is not a part of the WSMX development effort, a WSML compliant reasoner is required by various components such as data mediator, process mediator and discovery.

IRS (Internet Reasoning Service) III. IRS is a framework for Semantic Web Services that supports the publication, location composition and execution of Web Services based on their semantic descriptions. IRS supports the conceptual model defined by WSMO and also provides mappings for service descriptions provided in OWL-S. Although the approach is quite competitive to WSMX, choreography and orchestration do not follow WSMO specification and they are implemented in a non-standard way.

The main components of IRS are the IRS server, the IRS publisher and the IRS client. The server stores the descriptions of goals, mediators and web services along with domain ontologies. Discovery, composition, mediation, reasoning and invocation are all controlled by the server. Finally, the client provides a user-interface for goal-based web service invocation.

The publisher carries out the tasks required for publication. Publication has two roles in IRS. The first is where a web service represented by a URI endpoint is associated with a semantic service description known to IRS. The second is where standalone Java or Lisp code is wrapped to make it appear as a web service and then, as in the first case, the service is associated with a semantic service description known to IRS. Once a service has been published to IRS it is available to be used in the achievement of a user goal.

IRS has its foundation in an earlier IBROW project which made the distinction between tasks that need to be solved and problem solving methods that "provide abstract, implementationindependent descriptions of reasoning processes which can be applied to solve tasks in specific domains". Adopting the WSMO conceptual model, tasks in IRS are modelled as goals while problem solving methods are modelled as services. Discovery in IRS is based on matching the pre-conditions and post-conditions defined in the semantic descriptions of goals and services known to the IRS server.

METEOR-S. METEOR-S project proposes the application of semantics to existing web service technologies. In particular the project endeavours to define and support the complete life cycle of Semantic Web Service processes. The project extends WSDL to support the development of Semantic Web Services using semantic annotation from additional type systems such as WSMO and OWL ontologies. It is not based on an overall conceptual model and it is rather a collection of related discrete tools than a single, encapsulated architecture.

The development module provides a GUI based tool for creating Semantic Web Services using WSDL-S. The tool provides support for semi-automatic and manual annotation of existing web services or source code with domain ontologies. The publication and discovery module provides support for semantic publication and discovery of web services. It provides support for discovery in a federation of registries as well as a semantic publication and discovery layer over UDDI. The composition module consists of two main sub-modules - the constraint analysis and optimization sub-module (it deals with correctness and optimization of the process on the basis of quality service constraints) and the execution environment. The execution environment provides proxy-based dynamic binding support to BPWS4J execution engine for BPEL4WS.

The current implementation of METEOR-S allows for the creation of WSDL-S descriptions from annotated source code, the automatic publishing of WSDL-S descriptions in enhanced UDDI registries, and the generation of OWL-S descriptions, from WSDL-S, for grounding, profile and service.

3. Semantic Web in e-Government

The domain of e-Government is an example of a domain ridden by semantic problems – barriers in which the lack of interpretation of meaning of information in focus is the key obstacle for networked computer applications in administrative processes and services. This domain is unique because of its enormous challenge to achieve interoperability, given the manifold semantic differences of interpretation of, for example, law, regulations, citizen services, administrative processes, best-practices, and the many different languages to be taken into account within and across regions, nations and continents .

These semantic differences are related to a great variety of IT solutions currently in operation (on a local, regional, national, and international level), which will have to be networked (despite any effort of standardization). Semantic barriers of information exchange have a vertical and horizontal dimension (front-to-back-end and back-to-back interoperability). Because of the heterogeneity of IT infrastructures in administration and the tendency to operate closed systems and networks, it is not possible to rely on integrated systems within e-Government services.

Moreover, each of the local administrations has its own understanding of the domain (e.g. of the services to be provided to citizens and other clients) as well as of the interoperability needs. Domain specific standardisation as well as methods and tools may certainly help, but they will not unify the perspectives and the (professional) language of the actors involved .

On the other hand, many capabilities, which can be delivered by using semantic technologies, are applicable in the e-Government area. Some of the capabilities from almost twenty capabilities identified in are:

- concept-based search (precise and concept-aware search using knowledge representations across multiple knowledge sources);
- semantic data integrator (allowing data to be shared and understood across a variety of settings);
- semantic service discovery and choreography (re-use of existing services and the dynamic automation of processes);
- virtual consultant (understanding customers goals and offering products and services which can help them meet those goals).

Therefore, the combination of the two domains - e-Government and the Semantic Web - seems to be quite natural. The e-Government domain can provide an ideal test bed for current research in semantic technologies, and semantic technologies can be an ideal platform to achieve the vision of a knowledge-based, user-centric, distributed, interoperable, and networked e-Government.

As it is stressed in the current trends in e-Government call for joined-up services that are simple to use, shaped around and responding to the needs of citizens. On practical grounds, the integration of services is a basic requirement of service-oriented systems, which aim at gathering and transforming processes – needed for a particular user – into one single service and the corresponding back-office practices. Semantic Web Service technology provides an infrastructure in which new services can be added, discovered and composed on the fly. This technology combines the flexibility, reusability, and universal access with the power of semantic mark-up, and reasoning in order to make feasible the invocation, composition, mediation, and automatic execution of complex services with multiple conditional paths of execution .

From the administrative point of view, the Semantic Web and ontology-based approaches seem to promise a support for at least the following objectives:

- systematic management of dealing with all kinds of (electronic) informational resources;
- support for administrative processes crossing borders of organisations, systems and infrastructures;
- service quality improvement, e.g. responding to requests, information retrieval, and knowledge management with respect to different authors perspectives.

The Semantic Web has been in the focus of the e-Government research community since the idea of the Semantic Web has emerged (also because the Semantic Web seems promising in alleviation of integration and interoperability problems) . It is backed up by a bunch of research projects (a list of some of these projects is given in the next section) trying to validate principles of using semantic technologies in the e-Government area as well as to develop tools, infrastructures, components, etc. necessary to bring the idea of the Semantic Web to reality.

However, after years of intensive research and impressive scientific results, what the Semantic Web now really needs is real-world use cases (lifespan of which goes beyond the end of research projects), in order to demonstrate its added (business) value and to communicate to non-research audience that the Semantic Web technologies, although still a bit far from being mature, are already capable of setting in real life environment.

From the technical perspective, the main challenges related to the Semantic Web are to identify the objects which will need semantic mark-up, to provide (or generate) the appropriate mark-up, and to understand the processes which will use those objects and the related semantic meaning. However, in system development projects not only the life cycle of resources and their mark-up come into focus, but also the organisational aspects of information provision and use as well as the reasoning behind identifying, organizing, and sharing information (Klischewski et al. 2004). In this context, there is a lack of papers, published within the field of e-Government research, dedicated to how to make strategic use of semantic technologies in opposition to papers focusing only on technical aspects of employing semantic technologies.

3.1. EU R&D projects in the area of the Semantic Web and e-Government

The evolution towards integrated IT-based public services shows the necessity to adopt new ways of interacting with and between public service institutions. Since the member countries of the European Union are confronted with the challenge of providing also cross-border public services, the way needs to be paved for new approaches suited for them. Many of past and current EU-funded initiatives mainly focus on semantic enrichment of electronic services and their aggregation and orchestration towards combined “complex e-services”. This section provides a brief overview of some of these initiatives (information on other initiatives can be accessed through).

ONTOGOV (IST 507237: Ontology-enabled e-Gov Service Configuration). The overall objective of the project is to develop, test and validate a semantically-enriched (ontology enabled) platform that will facilitate the consistent composition, re-configuration and evolution of e-Government services. The platform enables public administrations: (a) model the semantics related to their e-Government services, (b) ensure consistency of the model when re-designing e-Government services due to an external (e.g. change in legislation) or internal trigger, (c) use this model in order to semi-automate the software configuration of e-Government services, (d) re-configure services in a manner that ensures the consistency of the service model, and consistency between the service and the related software, (e) have access to a knowledge enriched infrastructure mapping the entire history and life-cycle of e-Gov services (service design, configuration, and re-configuration). Its ontology work is heavily bound to OWL-S. Home page: http://www.ontogov.com

TERREGOV (IST 2002-507749: Impact of e-Government on Territorial Government Service). The project addresses the issue of interoperability of e-Government services for local and regional governments. Taking the view that government services are offered by a number of administrations interacting one with each other and that local administrations often act as a front office to citizens, the project's goal is to make it possible for local governments to deliver online a variety of services in a straightforward and transparent manner regardless of the administration(s) actually involved in providing those services. The project makes use of current state-of-the-art W3C technologies, such as OWL-S for describing ontologies. Semantic registries register web services and BPEL files will be used to orchestrate and compose procedures. Home page: http://www.terregov.eupm.net/my_spip/index.php

QUALEG (IST 507767: Quality of Service and Legitimacy in e-Government). The project aims at enabling local governments to manage their policies – they should be able to measure the performance of services they offer, to assess the satisfaction of citizens and to re-formulate policy orientation. The key enabling blocks of the project software solution are: (a) a WSDL based workflow management system which brings together the interoperability features of web services and the business process design and enactment features of workflow management, (b) a semantic engine for web services/workflows coupled with an ontology management system. It aims to jointly publish semantically rich web services interacting with legacy applications and information sources. Home page: http://www.qualeg.eupm.net/my_spip/index.php

EU-PUBLI.COM (IST 2001-35217: Facilitating Co-operation amongst European Public Administration Employees). The project attempts to achieve interoperability amongst public administration organisations by defining a 'Unitary European Network Architecture' into which the collection of distributed, autonomous systems of each public authority can be brought together into a common co-operative environment. In turn, this acts as a framework for the development of new and the re-engineering of existing European public administration processes in order to be made suitable for facilitated co-operation. The developed software components are based on the web service paradigm. The orchestration engine subsystem uses BPEL4WS technology to coordinate the workflow of composite macro-processes and the execution of its atomic components. Home page: http://www.eu-publi.com

DIP (FP6 – 507483: Data, Information and Process Integration with Semantic Web Services). The project's objective is to develop and extend the Semantic Web and web service technologies in order to produce a new technology infrastructure for Semantic Web Services - an environment in which different web services can discover and cooperate with each other automatically. DIP strives to develop Semantic Web Services as a scaleable and cost effective solution to the integration problem. Project results are represented by a set of tools used in implementing and realising parts of the open source WSMX architecture. Tools include Ontology management suite (consisting of editing, reporting, mapping, merging, versioning tools, etc.), WSMO Studio, Web service modelling toolkit etc. Although the project is not dedicated to field of e-Government, one of case studies should be implemented in this field. Home page: http://dip.semanticweb.org/

SEMANTICGOV (FP6-2004-IST-4-027517: Services for Public Administration). The project aims at building the infrastructure (software, models, services, etc) necessary for enabling the offering of Semantic Web Services by public administration through the use of the Semantic Web. Through this infrastructure, SemanticGov will address longstanding challenges faced by public administrations such as streamlining cooperation amongst agencies both within a country as well as amongst countries, easing the discovery of public administration services by its customers, facilitating the execution of complex services often involving multiple agencies in inter-workflows. The project intends to utilise the infrastructure represented by WSMO, WSML, and WSMX to implement components supporting the aim of the project. Home page: http://www.semantic-gov.org/

Other relevant EU projects are: GUIDE, www.guide-project.org; eMAYOR, http://www.deloitte.com/dtt/section_node/0%2C1042%2Csid%25253D97625%2C00.html; HOPS, http://www.bcn.es/hops/, BRITE, http://www.briteproject.net/; R4EGOV, http://www.r4egov.info/; OneStopGov, http://www.onestopgov-project.org/; Athena, http://www.athena-ip.org/ (although with application in business sector), eGOVRTD2020, http://www.egovrtd2020.org/, SAKE, http://www.sake-project.org/ etc.

The majority of these projects focuses on the technical level and thus still lacks a citizen-centred point of view that could be taken by implementing software components tailored to assist the citizen when applying for a public service. In these projects, citizens' needs take a background position compared to technical aspects that are very often predominating. Therefore, new approaches in e-Government have to put the emphasis on easy service accessibility for customers.

4. Access-eGov project

The FP6-2004-27020 Access-eGov Project (funded by the EC under the Sixth Framework Programme in the Information Society Technologies Programme) is coordinated by the Technical University of Kosice, started in January 2006 and is expected to last 36 months. Total estimated effort behind this project is 410 person-months. Access-eGov aims at development of component-based enhancements of existing e-Government infrastructure based on Semantic Web technologies and distributed architectures (service-oriented and peer-to-peer). These components will enable e-Government service providers (on all levels of public administration - local, regional, national, and European) to easily introduce any (new) e-service to the world of e-Government interoperability. Once the service is registered in the Access-eGov system, it may be localised, contracted and used (in case of e-service) automatically through agents and other IT components. For service users (citizens as well as businesses) Access-eGov will increase accessibility and facilitate connectivity of the existing e-services across organisational and regional borders, provide more information necessary for the use of traditional PA services and thus enable “integration” of traditional and e-services into “hybrid scenarios”. And since not all users feel comfortable when dealing with a myriad of PA services, a virtual personal assistant will guide users through this scenario.

4.1. Conceptual overview

A pyramid depicted in Fig. 1 represents the approach taken by the Access-eGov project. The pyramid consists of five layers creating a vertical hierarchy. Each layer can provide services for external applications (through a set of APPs interfaces) or can be used by the upper layer in the hierarchy.

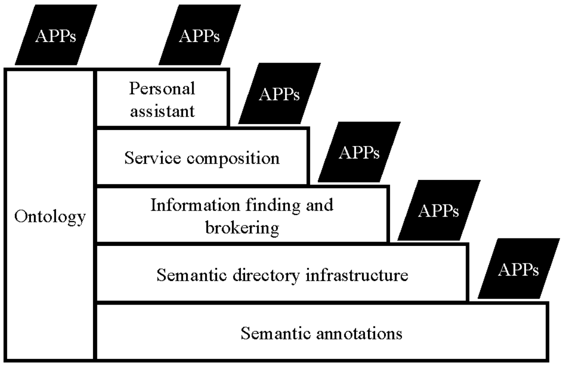

Figure 1. Five layers of Access-eGov system

Since the project uses semantic technologies to be able to search for appropriate government services and to ensure their semantic interoperability, the omni-presence of ontologies is not too surprising.

In order to register a government service within the Access-eGov system, it is necessary to provide a *semantic annotation* of the service – i.e. information describing the service at several levels. In particular, at least the following information should be provided:

- what the service is about (aim of the service, what the service can be used for);
- how it can be used or accessed (any contractual information, e.g. what information and in what format is necessary to supply as an input to the service);
- who is a target group of the service – i.e. who is eligible for the service (for example, it can be based on such characteristics as permanent address, citizenship, age etc.).

The project takes the position not to be invasive in existing solutions. This enables to index not only services currently without any semantic information attached but also services, which were developed with semantics in mind - so they already have a semantic description. This description will be replicated and it will be ensured that the Access-eGov keeps the valid version of the description.

All semantic descriptions are expected to be stored in a decentralised *semantic directory infrastructure*, which represents a heart of the system. Its role is to serve as 'yellow pages' of all the services provided by governmental organisations. Basically, the infrastructure contains a list of services, available publicly to citizens and business entities. A record of a service consists of a reference to the service (where the service is located) and semantic description of this service. The services registered in the system can be of three types:

- automatic services available electronically (e.g. web services);
- services accessible electronically, but manual communication with user is necessary (e.g. filling and submitting some form located on the web);
- services not accessible electronically (e.g. user has to visit a PA institution), only some information is accessible on the web.

Information finding and brokering layer serves as a data provider/search agent enabling to benefit from semantic description infrastructure. It enables to find which service types can be principally offered to a user ("user" can be a person or another software system/component). For this purpose it performs on-line communication with semantic description infrastructure each time when information is necessary. Its role is to support search - for a particular service in question (and some constraints, e.g. permanent address of user) to create appropriate queries (using all relevant ontologies) and to communicate them to the lower layer.

The aim of *service composition* layer is to generate a complex plan (scenario) how to cope with the given life event or a business episode. In order to satisfy a particular need in the given life event, a plan (containing information like: which services should be used, in what sequence, in what way etc.) will be assembled. Basically, it can be assembled in two ways: top-down and bottom-up. The former involves identification of the appropriate generic process definition for a particular life event, its subsequent pruning, and selection of particular services.

The life event (or some business episode) is represented as a general scenario, how to solve a need resulting from given life-event or business episode. The life event scenario is a general one - and should be adjusted/configured to a particular user. In order to serve needs of the particular user, the following steps should be taken:

- to adjust a general event scenario to the particular user in order to specify which steps must be performed;
- to select a service from those offered for each step in generated user-specific scenario (e.g. based on user address, birth place, etc.);
- to generate a plan for the user leading to an expected outcome of his/her life event.

The plan generated for a particular event represents a guide for the user, which should be followed. Execution of the plan will be "supervised" by a temporary virtual *personal assistant* assigned to the user by Access-eGov. The personal assistant is able to actively guide the user through the scenario. The role of the personal assistant is to execute a process instance (activity by activity).

Personal assistant performs some activities electronically (it accesses some e-Government services electronically on behalf of the user – i.e. the assistant invokes a web service of one institution and transforms the response appropriately to supply it to another service of some institution). Some activities, however, have to be performed by the user himself. The user is informed on the progress made (mapped to the process activities). Additionally for some activities, the user is asked to carry out some tasks, e.g. to take the form (delivered by the assistant), fill it in (some items may be pre-filled by the assistant), print it, sign, send by post or go to relevant public administration institution. Personal assistant represents a temporary 'one-stop shop' dedicated for one particular user and one particular life event and is generated in a dynamic way.

4.2. Proposed architecture

At the time of writing this paper (September 2006), a technical architecture of the above mentioned structure is being designed . The result is expected to be a flexible service-oriented architecture that goes beyond existing e-Government systems and overcomes restrictions of the existing solutions.

A main pillar of the structure is a peer-to-peer network due to its capability of adapting to failures without requiring the intermediation or support of a global centralized server or authority. A core role in communication with participating nodes is played by mediator nodes, which have the following responsibilities:

- Interface to service providers and legacy services (using layered wrappers, one layer of which can be tailored to a specific legacy application);
- Management of life and business events;
- Management of the registry infrastructure in order to facilitate the efficient service storage and location;
- Interface to personal assistants.

Apparently, mediators represent the most complex part of the Access-eGov architecture. Since participating services are mostly in the form of web services (and wrappers of legacy services can simulate this kind of accessing services), a semantic approach to web services is expected to be utilised. The choice of the project is WSMO which was designed from scratch addressing exclusively the needs of Semantic Web Services. Based on this selection, using WSML as a complementing mark-up language seems to be logical. Presumably, during the course of implementing our architecture, WSMX will be extended to fit the project's exact needs. More on the project can be found at http://www.accessegov.org/.

5. Conclusions

The paper provides an overview of e-Government aims, expected benefits, and challenges. e-Government has become also an important research area and quite a significant funding on EU level (FP5, FP6) as well as on national level (in some countries) went into this area. In the early stage of development focus was put on development of online services, later on reengineering and back-office modernisation became a priority. And recently the move towards electronic identity management, e-Participation, networked solutions and interoperability has become visible. Service integration and domain interoperability are principal requirements in the development of service-oriented e-Government applications. The Semantic Web, which aims to alleviate integration and interoperability problems, and in particular Semantic Web Services are addressing exactly these issues and appear to be promising technologies (which, however, does not mean, that it is an easy task).

The paper gives an overview of ontology languages, formalisms for modelling web services, and frameworks and tools for Semantic Web Services. A brief description of several relevant EU funded R&D projects in the area of e-Government and the Semantic Web is provided as well - including the Access-eGov Project in which the authors of the paper currently participate.

6. Acknowledgements

The work presented in the paper was partly supported by the EC within the FP6-2004-27020 Project "Access to e-Government Services Employing Semantic Technologies" as well as by the Slovak Grant Agency of the Ministry of Education and Academy of Science of the Slovak Republic within the 1/1060/04 Project "Document classification and annotation for the Semantic web".

7. References

[1] Communication from the Commission to the Council, the European Parliament, the European Economic and Social Committee and the Committee of the Regions, COM(2003) 567. The Role of eGovernment for Europe's Future, {SEC(2003) 1038}

[2] Communication from the Commission to the Council, The European Parliament, The European Economic and Social Committee and the Committee of the Regions. "i2010 – A European Information Society for growth and employment", COM(2005) 229

[3] Beyond e-Government. The world's most successful technology-enabled transformations. Booz, Allen, Hamilton, 2005.

[4] Communication from the Commission to the Council, the European Parliament, the European Economic and Social Committee and the Committee of the Regions, i2010 eGovernment Action Plan: Accelerating eGovernment in Europe for the Benefit of All, COM(2006) 173, {SEC(2006) 511}.

[5] Makolm A.: Creating Public Value Instead of Administrative Burden: The Austrian Supplementary Register for Other Concerned Parties. In: Electronic Government. Communication Proc. Of the 5th International eGov Conference 2006, Krakow, Poland, September 4-8, 2006, p. 351-358.

[6] The Danish e-Government Interoperability Framework. http://standarder.oio.dk/English/

[7] http://imu.iccs.ntua.gr/sweg/index_files/Program.html

[8] Berners-Lee T., Hendler J., Lassila O.: The Semantic Web. A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities. Scientific American, 284 (5), 34-43, May 2001. http://www.sciam.com/

[9] Semantic Technology. TopQuadrant Technology Briefing, March 2004, http://www.topquadrant.com/tq_white_papers.htm

[10] The World Wide Web Consortium. http://www.w3.org/

[11] Web Ontology Language for Services (OWL-S), http://www.daml.org/services/owl-s/1.1/

[12] Web Service Modelling Ontology (WSMO), http://www.wsmo.org/TR/d2/v1.1/

[13] Web Service Semantics (WSDL-S), http://www.w3.org/Submission/WSDL-S/

[14] Business Process Execution Language for Web Services (BPEL4WS). http://www-

128.ibm.com/developerworks/library/specification/ws-bpel/
[15]OWS-S Tool Directory. http://www.daml.org/services/owl-s/tools.html
[16]Web Service execution Environment. http://www.wsmx.org/
[17]Internet Reasoning Service. http://kmi.open.ac.uk/projects/irs/
[18]Semantic Web Services and processes. http://lsdis.cs.uga.edu/projects/meteor-s/
[19]Semantic Web Meets eGovernment, 2006 http://www.aaai.org/Press/Reports/Symposia/Spring/ss-06-06.php
[20] Klischewski R., Jeenicke M.: Semantic Web Technologies for Information Management within e-Government Services. Proc. of the 37th Hawaii Int. Conference on System Sciences (HICSS'04), IEEE Computer Society Press, January 2004.
[21]Gugliotta A., Cabral, L., Domingue, J.: Knowledge Modelling for Integrating e-Government Applications and Semantic Web Services. http://imu.iccs.ntua.gr/sweg/papers/SS0606GugliottaA.pdf#search=%22Knowledge%20Modelling%20for%20Integrating%20e-Government%20Applications%20and%20Semantic%20Web%20Services%22
[22]A list of IST eGovernment projects. http://europa.eu.int/information_society/activities/egovernment_research/projects/egovernment_projects/index_en.htm
[23] Kolter J., Schillinger R., Dobmeier W., Pernul G.: An Architecture Integrating Semantic E-Government Services. In: Electronic Government. Dommunication Proc. of the 5th International eGov Conference 2006, Krakow, Poland, September 4-8, 2006, p.163-170.