An ontologically principled service-oriented architecture for managing distributed e-government nodes

K. Votis\textsuperscript{a,}\textsuperscript{*}, C. Alexakos\textsuperscript{a}, B. Vassiliadis\textsuperscript{b}, S. Likothanassis\textsuperscript{a}

\textsuperscript{a}Department of Computer Engineering and Informatics, University of Patras, GR-26500 Rion, Greece
\textsuperscript{b}Computer Science, Hellenic Open University, 13-15 Tsamadou st., GR-26222 Patras, Greece

Received 8 March 2006; accepted 8 March 2006

Abstract

The need for managing distributed systems is prominent in e-government. Different applications and platforms that cover the overall range of the e-government implementation area need to interoperate in order to provide integrated governmental services to the citizens. This paper proposes an ontologically principled service-oriented architecture for the administration and integration of distributed nodes in an e-government network. The goal of the proposed design is to improve effectiveness and coherence by taking advantage of the enabling technologies of service-oriented computing, web services and ontologies. A two-level semantic mediator model is proposed to both provide an integrated description of entities and map actual information to them. This architecture was used on a prototype system developed for managing distributed educational directorates in the prefecture of Achaia, Greece. The pilot use of the system led to efficient decision making since it managed to mine information that was previously ‘buried’ in the local governmental infrastructure nodes.

© 2006 Elsevier Ltd. All rights reserved.

Keywords: Integration of distributed e-government nodes; SOA; Ontologies

*Corresponding author.

E-mail address: botis@ceid.upatras.gr (K. Votis).

1084-8045/$ - see front matter © 2006 Elsevier Ltd. All rights reserved.
doi:10.1016/j.jnca.2006.03.006
1. Introduction

The e-government is a means that aims to relax the citizens’ interface with over-complicated and slow processes of local and regional governmental organizations (Martin, 2005). The advent of easy to use and cost-effective technologies such as the web has paved the way towards user friendly e-government systems; however, these technologies alone are not adequate to handle the complexity of such systems. The e-government processes are complicated and complex information systems with monolithic architectures are currently handling them (Ramnath and Landsbergen, 2005). This complexity is transferred to the users each time they transact electronically with such a system. Users of such systems are either citizens or employees of governmental bodies. Although the need for simple and faster communication between government and citizens is supported by both EU and local initiatives, only a few cases have been a success, especially in South-eastern Europe. Even past efforts in e-government worldwide have produced mixed results. Singapore (Ke and Wei, 2004) has managed to put on-line many public services with relative success. Similar efforts in the US (Norris, 2003) and Australia (Martin et al., 2004) have produced mixed results. The software industry and governmental organizations are still seeking for the right technological and business solutions (Chamberlain and Castleman, 2003).

In order to meet increased expectations by the public, administrations need to deploy a variety of channels for their service delivery-channels that allow users to consume their services anytime, anywhere (Stoltzfus, 2005). New developments in IT will hopefully allow public sector agencies to meet these challenges by adapting their front and back office; new ways of interaction through a variety of channels, restructured services that accommodate users’ needs, and re-organized business processes within and between separate administrative bodies are some of these new ways. However, complexity needs to be handled first.

Complexity can be handled initially by decomposing complex processes into simpler ones. This way, a monolithic architecture is transformed into a construct comprised of simpler building blocks. This entails the use of the appropriate technology to manage, combine, interface and diffuse these building blocks to ensure an adequate quality of service. Service-oriented computing (SOC) enables the design of such architectures (SOA—service-oriented architecture) (Singh and Huhns, 2005). This paradigm was not available a few years ago but the advent of web services (WSs) (one of its enabling technologies) changed all this. With services as building blocks, even a large and complicated system design can be efficient and flexible. By reducing complexity, system-user communication (a parameter so valued in e-government) is easier.

In this work, we propose a new architectural approach for managing distributed e-government services using advanced ontology mechanisms. We base our proposal on a scenario of distributed government agencies offering simple or complex services to citizens. The proposed scheme requires a SOA that enables efficient service coordination, integration of data residing in different agencies and increases interoperability.

A framework is proposed for developing a semantic mediator model that manages user requests coming from different access points uniformly. This model is comprised of two levels or dimensions. The first dimension describes generic and domain specific concepts (entities) that are an integrated description of data. The mapping of actual information to the entities is realized by the second dimension of the model. A global ontological schema is used to integrate all source schemas, acting as a mediator between the various data sources. This architecture was used on a prototype system developed for managing
distributed educational directorates in the prefecture of Achaia, Greece. The pilot use of
the system led to efficient decision making since it managed to integrate and coordinate
information that was previously ‘buried’ in the local governmental infrastructure.

The rest of this paper is as follows: Section 2 presents relevant literature in e-government
projects while Section 3 discusses the state-of-the-art in service-oriented enabling
technologies utilized by the proposed architecture. Section 4 presents the proposed
service-oriented approach and its implementation architecture. Section 5 showcases the
example use case in the prefecture of Achaia and discusses the results from the pilot
operation, and finally conclusions are drawn in Section 6.

2. Literature review

The introduction of new information and communication technologies in public
administration is an integral part of the overall government policy for the decentralization
of authority and the re-organization of public services. Despite the implementation of a
number of projects in the last decade, information technologies have not yet penetrated
the public administration to a satisfactory level. This delay has created a vicious circle with
the preservation of the traditional bureaucratic and inefficient structures, mechanisms and
mentalities (e-Europe, 2005).

While south-east Europe efforts in e-government are just beginning to be realized,
electronic government is already a major priority in Europe and western world countries.
Numerous e-government solutions in European countries are widely supported by WSs
and ontologies as a way for agencies, other associations, businesses and citizens to make
queries and discover the information available in their systems (Bradier, 2005; e-Government
Unit, 2005). Efforts such as the (Terregov project, 2005) adopt the
principles of a SOA based on interoperable components with dynamic support for finding
services. The idea is strengthened by the fact that information, services and administrations
are spread over several information systems. The architecture contains a set of
collaborative tools for e-government WSs that are semantically enriched. Another
significant effort is the Ontogov—Ontology-enabled e-government service configuration
project (ONTOGOV, 2006) that aims to develop, test and validate a semantically enriched
(ontology-enabled) platform that will facilitate the consistent composition, re-configura-
tion and evolution of e-government services.

Although increasing efforts have been devoted to the automation of the Greek Public
administrations system, the level of electronically provided services is still far from the
European standards. A number of information systems are already operating or under
development in the public sector. One of the most important Greek programs was
“Kleisthenis” (Kleisthenis, 2004), which was an operational program for the moderniza-
tion of public administration via interventions of technical, organizational and educational
character. It financed studies for the best use of new technologies in public services and for
the creation of the necessary common infrastructure. Another local effort, the “Ariadne”
project (Ariadne project, 1992) aimed at building the necessary infrastructure in order to
provide one-stop public services to citizens and businesses. The infrastructure consisted of
more than 1000 access points for citizens (one at each municipality), a call center and a
portal. Both of these efforts had marginal effects on the public, however more funding is
under way for e-government development.
3. Service-orientation enabling technologies

The growing data demands, the infrastructure complexity and the great heterogeneity of governmental organizations and public administrations required the introduction of a new architecture—one designed for extensibility and flexibility. Organizations are now turning to SOA based on WS and semantic technologies to make existing applications, components, and data available for reuse and to simplify the consumption of these reusable assets. Although SOA go a long way towards providing interoperability in distributed, heterogeneous environments, managing semantic differences in such environments remains a challenge. Several implemented architectures and infrastructures claim to be SOA. The enabling technologies of SOA include WS and semantic technologies such as ontologies. Before continuing with the details of the proposed methodology, we briefly present their basic characteristics of the enabling technologies.

3.1. Web services

WSs provide a standard means of interoperating between different software applications, running on a variety of platforms and/or frameworks. WSs describe a collection of operations that are network-accessible through standardized Web protocols, and whose features are described using a standard XML-based language (Leymann et al., 2002; Kreger, 2001). There are numerous other definitions of what WSs are. Some of them describe a WS as interfaces, but they impose some language or transport method. For example the W3C consortium definition (W3C Web Services Working Group, 2004) imposes the use of SOAP (Box et al., 2000) and WSDL (Christensen et al., 2001). WS Description Language (WSDL) provides a model and an XML format for WS description. WSs are described in two fundamental stages: an abstract and a concrete one. The abstract description, describes WSs in terms of their exchanged messages. At the concrete level, bindings specify transport and wire format details, endpoints associate bindings to network addresses and finally, endpoints that implement a common interface are grouped together. Other definitions introduce the idea of software components or modules, like the UDDI consortium definition (OASIS UDDI, 2006), and others the interface (Sheth and Miller, 2003). In order to provide a model and a context for understanding WS and the relationships between the various specifications and technologies an instance of SOA, called WS Architecture was born (Alonso et al., 2004).

The abovementioned technologies have become de facto standards for the creation and use of WS, with significant industry commitment to them. However, there are some limitations to these capabilities. The UDDI standard does not offer a rich enough description of a service: if we think of scaling to the WWW, it is easy to imagine that searching a UDDI registry would typically result in a large number of candidate services being returned. A more detailed selection would then need to be executed by contacting each of the service providers for further information, a process which could rapidly become intractable. Similarly, while WSDL describes the input/output format of a WS, it is silent with regard to, for example, any impact execution of the service might have in the physical world or on the progress of a business process.

To overcome these limitations, we require a richer, finer-grained description of WS. Fortunately, more or less in parallel with the development of WS technology, the Semantic Web has seen rapidly increasing interest and activity. Semantic Web technology provides a
way to describe SOA elements in a human- and machine-readable way which facilitates automated discovery and composition of elements into new applications. The approach is backed by the W3C and other standards bodies. The Semantic Web will enable the accessing of Web resources by semantic content rather than just by keywords. Resources (in this case WS) are defined in such a way that they can be automatically ‘understood’ and processed by machines. This enables the automation of service discovery, acquisition, composition and monitoring procedures (Sycara et al., 2004).

3.2. Ontologies

Ontology has been a very popular buzzword during the last 3–4 years, although it does not refer to anything conceptually new, nor is it very precisely defined. The general definition states that an ontology is a model of the real world, created for future multiple uses (Staab et al., 2001). Ontologies can be used to describe the concepts and relationships in all domains of human life and work, particularly whenever a common language between human and machine, machine and machine, or even human and human is necessary. Thus, an ontology seems to be a self-describing, communication-enabled conceptual model. One of the definitions states that an ontology is a shared formal specification of the conceptualization of a knowledge domain. Any ontology has to fulfill a set of criteria to be properly developed. It has to be clear, coherent, extensible, encode minimal bias, and possess minimal ontological commitment (Gruber, 1993). Only by developing a proper conceptual model the entire requirements can be fulfilled. Ontologies are the cornerstone of interoperability and the task of their creation is important.

In practice, authors who consider an ontology as a sophisticated vocabulary insist on the fact that ontologies are not limited to the restricted set of relationships between nodes that is usually found in a thesaurus. According to them, an ontology is a taxonomy of concepts, which can be linked together by an open number of relations. These relations may have explicit logical properties such as transitivity, reflexivity, and symmetry. Each concept must be named by at least one or possibly several terms, chosen in one or several languages. Concepts may be described and characterized using an arbitrary number of attributes and descriptors (Davies et al., 2004).

The ontology can be divided into two parts: Taxonomical Box (T-Box) and Axiological Box (A-Box). The former one represents concept hierarchies and their properties while the latter contains individuals of these concepts. A property can be considered as a role of a concept with respect to some other concept or as a parameter of an individual that distinguishes it from other individuals of the same class.

Other authors and standardization groups use the term ontology to name some kind of object-oriented database schema that can be expressed in a XML jargon such as RDF-S and OWL. A good example of this vision is offered by the work of the OWL-S Coalition (2006). OWL-S is a schema specified in the OWL language, which is proposed for setting-up web-service registries. OWL-S appears as an extensible object-oriented schema, and as such could be specified in UML. An OWL-S repository is a registry of web-services. It is a database structured according the OWL-S schema that contains descriptions of web-services. In this context, Web-services are objects (instances of the Service Class) rather than concepts. They must have a unique identifier (a URI) and do not require to be named using human-understandable names. In other words, OWL-S has nothing to do with a taxonomy of concepts, and an OWL-S registry is a specific database of WS. The OWL-S
coalition therefore uses the term ontology with a very different meaning than as used by those mentioned before who consider ontologies as controlled vocabularies for naming concepts. Furthermore, it allows for defining both T- and A-Boxes, and also for combining multiple ontologies using import closure.

A large number of public-funded research projects have been launched during the last 4 years aiming at applying ontology-related methods and tools to various application domains. The number of resulting operational applications appears to be very limited, if any. Terminology-oriented applications do not generally use the word ontology.

The notion of ontologies helps the SOA architect prepare generalizations that make the problem domain more understandable. In contrast to abstraction, generalization ignores many of the details and ends up with general ideas. Therefore, when generalizing, we start with a collection of types and analyze commonalities to generalize them.

4. A SOC approach to e-government service provision

4.1. Network of e-government service centers

Suppose that several government agencies (e-government Service Center—ESC) are connected through an integrated E-government network in order to provide services to the citizens. Services are either simple (provided by a single ESC) or complex (requiring the collaboration of several ESCs). Such an infrastructure is depicted in Fig. 1. Each ESC has a Portal for user access and it can supply the whole set of available services or a part of them. Moreover, some services may reside locally to the ESC and others remotely. In general, a service may reside also out of an ESC, to local subsystems that contain useful data about the e-government infrastructure. A unique point of access redirects the user to the preferred ESC Portal but, in general, the user must be able to access the services he/she is subscribed to and his own profile from every Portal.

Fig. 1. An ESC eco-system; distributed E-government nodes providing simple and complex services.
In order to manage the different portals, a User Ontology component is used to provide a central repository to store and keep consistent all the information concerning the user. Each Portal can refer to the User Ontology to retrieve these kinds of data. Nevertheless, each Portal has its own user management capability, based on a local database. This necessitates the development of opportune procedures of synchronization between the Portals’ user management services and the User Ontology. Each ESC integrates a Portal and a local scripting module consisting of the corresponding mapping scheme referring to the User Ontology. Service provision is transparent to the user; the user may connect to any Portal in order to receive the same services as if he/she was connected to the central one.

A Semantic Mediator Model is used to integrate all source schemas, acting as a mediator between the different pieces of distributed information (various data sources). The Semantic Mediator Model consists of semantics describing the data structure of the distributed resources in a unified and clearly defined model in order to simplify the data search process. When the mappings between the source schemas and the Semantic Mediator Model have been created, the user can browse the global schema and discover what information is present in the system and where the information is located.

Collaboration among ESC portals is performed in terms of SOAP messaging. Each portal implements one or more WS for accessing the corresponding stored data, and the descriptions associated with these services are gathered into a Central WSDL Repository (CWR). WSDL provides an XML-based grammar for describing a web service interface. Portals providing services will publish on the CWR their WSDLs, and portals seeking services will retrieve the WSDLs. To maintain the confidentiality and integrity of data exchanged among portals, Web Service Security (WS-S) (OASIS Web Services Security (WS-S), 2006) provided by OASIS organization can be used. The WS-S standard set of SOAP extensions can be used when building secure WS to implement message level integrity and confidentiality.

4.2. Semantic Mediator Model structure

The integration of various levels uses ontologies for the introduction of the semantic information, related to e-government resources. In order to have an overall integrated view of the resources through their representation using ontologies, it is essential to have a semantic description for both the structure services provided by the ESCs systems and their functionalities.

One of the main issues in designing and developing the Semantic Mediator Model is to make it maintainable and extensible while assuring consistency. The main idea is to separate the ontology into two dimensions.

The first dimension cuts the concept space into a Generic Ontology and a set of Domain Specific ontologies. The Generic Ontology forms the core ontology and it is the same for any ESC portal is used for. In other words, this ontology set can be considered as an intrinsic part of the system. The Generic Ontology describes entities and concepts that are similar to the whole system. For example in the educational use case described in Section 5, teachers, students, school buildings, classes and laboratories are entities of the Generic ontology that provides a relational model about the main elements that comprise an educational system (Fig. 2).
The Domain Specific one is created specifically to map the individual domains of the system and to allow the potential extension of the information model. It uses the entities of the Generic Ontology in order to describe the specific features and structure of the data that is residing in each ESC in an integrated form as depicted in Fig. 3, again for an ecosystem of Educational ESCs. Both the Generic Ontology and Domain Specific Ontologies are comprised of a semantic model that provides an integrated description of the data about the educational systems that is stored in the various systems involved in the ESC Network. This structural approach might seem as a constraint at first, in our point of view however it should make users understand better the structure of knowledge underlying the system and, more importantly, keep it reasonably maintainable.

The Semantic Mediator Model uses the second dimension that includes two parts of the information. The first part maps information of the entities that are described by the Generic Ontology and Domain Specific Ontologies to the actual data structure that resides in the distributed ESCs. The second part consists of a scheme that presents the choreography and the topology of the data that is separated among the ESC systems. This optimizes the search procedure in particular systems by bounding the number of query requests to the systems of the ESC network that actually have the requested data. Furthermore, the topology includes references to the CWR exposing the necessary information for the dynamic invocation of the WS provided by the systems of the ESC Network.

The Semantic Mediator Model is depicted in Fig. 4. The two-dimensional approach provides the desirable unification of ontologies’ descriptions along with the mappings and the topology of the actual data in the ESC network. This configuration contributes to the creation of a machine-readable and machine-understandable model that can be used by the appropriate infrastructure to discover what information is present in the system and where this information is located.

4.3. Mapping the heterogeneous data sources

The proposed methodology is based on the creation of an ontological model to support locating and accessing data residing in heterogeneous databases throughout the distributed e-government nodes (actually the ESCs). In order to realize this architecture, an integration of the heterogeneous data sources is needed with the capability to present the user with a unified view of the disparate data schemas for browsing and querying.
We chose to create a global schema (ontology), integrating all source schemas, which acts as a mediator between the various data sources. This functionality is implemented by the mappings that are included in the second dimension of the Semantic Mediator Model. The user can also issue queries to the mediator that are automatically translated to the respective platforms and schemas of the data sources, where they are executed. Using the mappings from the various source schemas to the global schema, it is also possible to

Fig. 3. Example of a domain ontology.

Fig. 4. Semantic Mediator Model.
automatically generate transformation scripts for the transformation of instances between different schemas.

We used a semantic approach to the integration problem instead of abstracting a global database schema from the source schemas. The meaning of the data is captured in the Semantic Mediator Model and the data in the sources is given meaning by creating mappings between the sources and the ontology. The Semantic Mediator Model is an integrated virtual view of the information present in the ESCs. Information integration typically involves data sources from different data schemas. The differences in these data schemas imply the necessity of not only creating a simple (syntactic) translation from one platform to another, but also a semantic mapping, linking the entities in the dispersed schemas based on the correspondence to their meaning. When the central ontology has been created, it is possible to create mappings between the source schemas and the ontology, thereby semantically linking the terms in the sources to each other. This link provides a meaning to the data residing in each data source which in turn relate to the meaning of the data as described in the ontology model (Fig. 5).

4.4. Proposed architecture

The proposed scheme requires a specific architecture that makes it possible to integrate the data residing in different systems of the ESC Network in a flexible and interoperable way. Our approach introduces a framework for developing the Semantic Mediator Model, the User Ontology and managing the accesses and search queries from the users to the network. This framework, presented in Fig. 6, involves the following functional elements:

The Model Repository where the ontologies and the mappings information contained in the Semantic Mediator Model are described using XML-based ontology representation.
standards. More particularly, the main ontology language used is OWL, a language based on the field of computational logics called Description Logics. The mappings and the choreography are described in a custom XML schema.

The User Ontology Node stores all the appropriate information about the access rights of the users to the integrated data. The User Ontology Node provides information to both the ESC portals and to the Query Manager in order to guarantee an adequate level of security that is required in e-government collaboration networks.

The Model Composer provides the mechanism and the graphical interface for composing the Semantic Mediator Model and creates the mapping information between the ontologies of the heterogeneous data schemes.

The Query Interface provides the appropriate connection interface to the ESC portal, the place where the user is composing the search queries.

Finally, the Query Manager is the platform which accepts the queries from the Query Interface and initiates the ontology-based search process.

4.5. Querying the dispersed data sources

In the overall system architecture, the reporting service is provided by a specific Query Manager based on the Sesame storage and query server (OpenRDF, 2006) which interacts with the ESCs portals and with the individual data sources. For this reason, conceptual queries on classes in the ontology should be implemented considering the users requests and the Semantic Mediator Model. These conceptual queries are, with the use of mappings to database schemas, automatically translated into instance queries that can be executed on the database platform of the data source.
The query itself consists of the following five parts:

- A name, identifying the query.
- A “Select” clause, which specifies which (indirect) properties to retrieve.
- A “From” clause, which specifies which class in the ontology is queried.
- A “Where” clause, which specifies which additional conditions the instances need to satisfy in order for them to be selected.
- An “On” database clause, which specifies which database to query.

The following steps constitute the query implementation strategy:

1. Search the repository for instances which match the keywords.
2. Get the class for each instance.
3. Find mapping information for a class.
4. Get the super-class(es) and/or sub-class(es) of a certain class.
5. Return all instances of a class.
6. Return relevant instances of an instance.

An example of the transformation of a RDQL query to a SQL query is presented in Table 1.

The following steps are involved in the querying process:

1. The user selects the query to be executed along with its parameter values using the portal interface.
2. The ESC Portal identifies the query and sends the query name along with the parameter value to the Query Interface over a SOAP interface.
3. The Query Manager requests the query from the query interface.
4. The Query Manager transforms the query to the appropriate form for questioning the heterogeneous databases.

Table 1
An example of RDQL to SQL query transformation

<table>
<thead>
<tr>
<th>RDQL query</th>
<th>SQL query</th>
</tr>
</thead>
<tbody>
<tr>
<td>SELECT ?TeacherName, ?TeacherSurName</td>
<td>SELECT teachers.tname, teachers.tsurname</td>
</tr>
<tr>
<td>WHERE classes.classtitle like ‘B1’ AND classes.teacherID = teacher.ID</td>
<td></td>
</tr>
</tbody>
</table>
5. The Query Manager locates, in the second dimension of the Semantic Mediator Model, the appropriate ESCs that stores the desired information.

6. The corresponding WSDLs are retrieved from the CWR and the Query Manager dynamically invokes the web methods for querying each system.

7. The results are returned, translated to a suitable format by the Query Manager, and ultimately displayed to the user by the ESC Portal.

5. **A use case in the prefecture of Achaia**

This section discusses the implementation of the proposed architectural model for the management of distributed educational departments (primary and secondary educational directorates) in the prefecture of Achaia, Greece. The solution incorporates powerful adaptivity for the management, authoring, delivery and monitoring of services. The enabling technology is based on SOC and results in significant reduction of the difficulty to access distributed resources within the prefecture. Dominant technologies are used such as WS and ontologies.

The system increases the opportunities for the management board of the prefecture to take concrete strategic decisions regarding the educational departments (public primary and secondary schools) of the area as well as to depict their regional status. Administrators can use the application to make critical decisions in a variety of administrative areas, including forecast budget needs and educational trends, enrollment management to achieve an appropriate mix of students, improve or construct new educational facilities. They have the opportunity to use data from multiple sources in order to gain useful insights. It is associated with a methodology for the classification of management and monitoring processes and with the integration and interoperability of heterogeneous systems and applications that are required, independently of the level that they reside in. Before going into details, the background of the deployment context is briefly presented.

5.1. **Background**

The Greek educational system remains highly centralized. Central government through the ministry of Education and its departments formulates and adopts education policy. The Greek school curriculum is national and compulsory. There are standard numbers of hours, content and textbooks provided by the National Ministry of Education for all levels of schooling.

The most characteristic feature of the Greek educational system is its high degree of centralization necessary for the close control of curriculum, teaching methodology and teaching staff. The Ministry of National Education and Religious Affairs and its Pedagogical Institute are responsible for managing among others, the school curriculum, timetables, educational and human resource (Ifanti, 1995).

The National Educational school system (Primary and Secondary) consists of three individual levels, each one with its management requirements and particularities:

- **Pre-school education**: It is optional and available for children from age of 2 to 6 in public or private kindergartens and public or private Nursery Schools.
- **Primary education and lower secondary education**: It is compulsory for all children 6–15 years old.
Post-compulsory secondary education: It consists of two school types, the unified upper secondary general education schools—“Eniaia Lykeia” with duration of studies of 3 years and the Technical Vocational Educational Schools—TEE with a duration of studies of either 2 years (A’ level) or 3 years (B’ level).

Schools are internally non-hierarchically structured with the faculty deciding on all issues while school directors have only a coordinating role. However, in practice the centralized nature of the educational system does not allow the faculty to have any real autonomy to decide on issues concerning school life. Furthermore, local authorities (municipalities) are responsible for school buildings. So besides central government, the administration in the prefectural level is also responsible for some educational resources via the so-called Directorates of Primary and Secondary education. These Directorates are responsible for:

- the coordination of the Education Offices, the supervision of the school directors and the coordination of the schools located functioning in their prefecture;
- the supervision and coordination of the maintenance of school buildings and the improvement of school equipment and workshops; allocation of teaching staff to schools; supervision of private schools;
- the submission of proposals both to the prefect of their prefecture and to the Ministry of National Education and Religious Affairs for the improvement of educational activities in their prefecture.

The administrative tasks and responsibilities of the Heads of Education Offices are similar to those of the Directors of Education in the prefecture. In addition, issues concerning teachers’ status are dealt with by a Regional Council in each prefecture and by a Central Council in the Ministry of National Education and Religious Affairs at a national level. The Councils consist of five members. The President of the Regional Council is the head of the Directorate of Education in the prefecture, two members are heads of Education Offices or, if there are none, teachers with the highest grade in the career structure (A’ grade), and two more members are elected as representatives of the teaching staff. The Central Council comprises three heads of Directorates of Education with corresponding alternates and two elected representatives of teachers (Dimitrakopoulos and Mauromatis, 2002). It is obvious that the coordination of this system is difficult since it is comprised of three levels of education and two management authorities (one national and one local).

5.2. Design and deployment of a service-oriented e-government system in a prefectural level

The use case selected for the system was focused on the pilot running of Government to Government activities related to managing the educational activities of the prefecture of Achaia (the third largest in Greece in terms of population) for the academic year 2004–2005. This critical and huge process was at that moment managed by human actors without the support of any specialized software application. Applying an e-government solution was a big challenge because numerous processes and data from different actors needed to be integrated: from analyses of test information to data discovery.
The design begun with the development of the Semantic Mediator Model. Database schemas such as the “school buildings”, “students”, and “classes” domains were used as the basis for the construction of the model, as depicted in the examples provided in Section 4.

In a typical scenario exploring our system’s technical feasibility, an administrative manager wishes to analyze the appropriate information that will help him decide about the relocation or the construction of a new school unit within the prefecture. In the past, this was a typically time consuming procedure because human actors had to collect all the required information from all public schools in the respective area and then to analyze more than 20 parameters regarding the students, the quality and safety of school units, the availability of classes and other different values residing in different agencies. The current system collects all this information automatically and presents an analysis screen that helps decision making. Each activity initiates a Web service operation for the corresponding task test procedure. In this scenario, the user has logged into the system and then proceeded to finding the key concepts ‘building plots’, ‘School building status’, ‘Available Resources’, ‘Classes’ and ‘Students’. A set of related information is selected based upon an aspect of user requirements for functionality. This information is available in the Semantic Mediator Model. Having collected sufficient information to perform the task, the Query Manager feeds the collected data into the system and handles executions according to user preferences as described in Section 4.5. Fig. 7 illustrates this scenario using a UML diagram.

The user is able to access analysis reports for these executions and results which also contain graphic results as presented in Fig. 8.

There is also a user-friendly map of the Achaia area that has been integrated within the system and provides a valuable human-interface interaction as depicted in Fig. 9. This map presents the current status of school units and building plots. The system is a user and task-centric platform that shifts effort away from users so that they can complete their tasks more efficiently.

5.3. Pilot use results

Statistics and detailed reports provided valuable insights as they impact strategic decisions. For example, the system generates reports regarding the safety conditions of
schools. Statistics showed that about 50% of school buildings within the prefecture of Achaia are not equipped with fire safety systems. When the number of classes per school was calculated, it was found that approximately 85% of the schools had an overabundance in classrooms (this meant that the number of students decreased during the last few years in the prefecture). In about 15% of the schools there were fewer classrooms than classes, forcing the principal to use one classroom for more than one grade. A possible reason for this is that some schools had more classrooms than classes, where more classrooms were needed to accommodate secondary learners.

Summarization reports and statistics permit the administration to quickly conclude that some facilities in the prefecture Achaia had shortage of classrooms at some schools, as some of the classes exceed 30 learners. Special purpose classrooms were also not readily
available. Inadequate sport facilities and sport equipment deprive learners of the opportunity to spend the time after school hours developing other human capacities. These were some of the initial results that were mined from the integration of heterogeneous data sources in the educational system in the prefecture of Achaia. Since the system is mainly used for local inter-government coordination, it is hoped that the integration of other national and local authorities will reveal more needs and problems that would otherwise be hidden.

6. Conclusions

Searching in distributed information resources and coordinating/orchestrating service provision is challenging problem and it seems that semantic web technologies, such as SOA, offer a viable solution. This work discussed an ontologically principled SOA for the management of distributed e-government nodes. We introduced a two-level ontology scheme that enables efficient integration by mapping concepts to actual information in a distributed environment. This ontology scheme and the associated architecture provide an integrated approach towards achieving integration of different and distributed systems. The proposed architecture was used in a pilot system that manages educational departments (primary and secondary educational directorates) with good results.

Our proposal provides a means of addressing many of the problems associated with commonly used e-government paradigms by adding to the overall system in terms of flexibility and interoperability. Future work includes the implementation of the ontology scheme, taking into account the specific features of different E-government services.

Acknowledgements

This work was partially supported by the Greek Prefecture of Achaia. The authors would like to thank G. Aggelopoulos, Educational co-prefect of the prefecture of Achaia, Greece.

References

Ariadne project, 1992. At URL: http://www.ariadne-t.gr/
Christensen E, Curbera F, Meredith G, Weerawarana S. Web service description language (WSDL) version 1.1, at URL: http://www.w3c.org/


OWL-S Coalition, at URL: http://www.daml.org; 2006.


