*"1st International Symposium on Computing in Informatics and Mathematics (ISCIM 2011)"*
*in Collabaration between EPOKA University and "Aleksandër Moisiu" University of Durrës*
*on June 2-4 2011, Tirana-Durres, ALBANIA*

# Integration of Applications Based on SOA in Government Institutions

Ajri Shej[1], Betim Çiço[2]

[1]*SEEU Tetovo, Macedonia*
*Email:ajri.shej@gmail.com*
[2]*Computer Engineering Department, Polytechnic University of Tirana*
*Email:bcico@icc-al.org*

## ABSTRACT

Historically, services in the government institutions were developed in different hardware and software platforms. In these conditions, emerges for integration of these platforms is crucial for creating possibility to provide services for citizens and businesses based on the web (web services) in all government institutions without knowing government hierarchy. There are a lot of studies which tends to solve these kind of issues. The purpose of this study, based on analysis done in existing hardware and software systems, is to propose platform based on SOA which will support these web services. After final analysis of these web services we would recommend integrated platform which will enable government institutions to 'talk' to each other. This obviously will increase the efficiency between government institutions also will increase the citizens sanctification and efficiency. Web services based on this integration will allow quantitative and qualitative services, which will decrease the cost of the services provided in traditional way from the government institutions. These kind of services can be used for G2G, G2B, B2G services. Hardware and software platforms independence offers huge advantages for others systems which would like to be part of this integrated and interconnect platform.

Keywords: SOA, Web Services, government, integration systems..

## INTRODUCTION

It's bad enough being between a rock and a hard place. But many government agencies are being squeezed by three rocks: Budget cutbacks, mandates to deliver new services and ongoing dependence on legacy applications and platforms. Adding to the pressure are the growing demands on agencies to collaborate with other agencies and deliver their services across multiple channels. To confront these challenges, public-sector IT executives have become increasingly focused on the benefits of adopting a "Service-Orientation." Service orientation provides a wide range of benefits that are particularly applicable to the challenges faced by government agencies today, not the least of which is cost savings. By making existing IT functionality available to other applications via standardized interfaces,

SOA enables agencies to build new capabilities without starting from scratch. This "building block" approach can really help stretch budgets that are constrained by aggressive tax cuts and under constant scrutiny. Service orientation makes it possible for individual citizens and employees to interact online with a host of services. It also it less expensive for governments to link their business processes to those of their suppliers, vendors and other business partners. And if that weren't enough, the benefits of service orientation provides extraordinary flexibility by treating elements of business processes as components to be reused and combined in different ways to address changing needs and priorities. In more cases government agencies are gathering date which are not updated and synchronized with other agencies in the government hierarchy [1]. Heterogeneous environment of systems and application costs high level of unstructured and duplicated date in different levels of IT infrastructure which increase maintenance cost of such kid of IT systems. It is more then evident that heterogeneous system in government agencies has to 'talk' to each other in order to able to increase efficiency of government agencies, to improve services quality for businesses and citizens and to cut costs.

## 2.0 APPLICATION INTEGRATION

Integrating applications is a term that relates to the plans, methods and tools aimed at modernizing, consolidating and coordinating the computer applications in a corporate for the purpose of increasing efficiency and meeting the needs of business [2]. Usually, a corporation has applications and databases that do not communicate among themselves. A tendency of companies is to continue to use these systems while adding or migrate to new technology [3].

**Integration of applications is the separation and sharing, safely and orchestrated the data and processes between different applications of a company [4].**

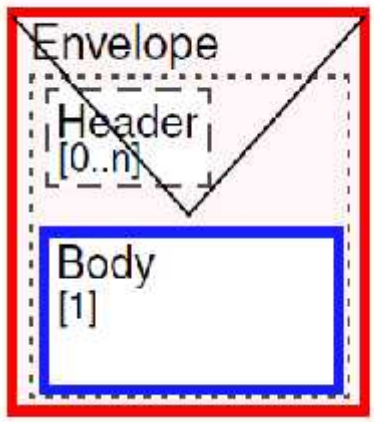
## 3.0 TYPES OF INTEGRATION

Integration architecture is usually built systematically in several layers. Integration usually start building architecture at the lowest layer and climb gradually. Omitting a layer is a short-term solution to speed up the process, but we will almost certainly have to pay back this time later. The most important types of integration are [5]:

1. Data-level integration
2. Application integration
3. Business process integration
4. Presentation integration
5. B2B integration

## 4.0 SERVICE ORIENTED ARCHITECTURE (SOA)

SOA (*Service Oriented Architecture*) is a concept for software architectures with a focus on business processes. The underlying principle behind the concept of an SOA is the idea that IT systems, software, devices and services will be integrated and "talk" to each other—even if they were never specifically designed for that. We propose that architecture which will support INTEROPERABILITY from logical aspect should be centralised, while its physical setting should be distributed. The proposed architecture will help the governemt to extend the value of the applications and business processes which currently running in the governemtn institutions. Proposed arichtecure is fully modular and interoperable, these features will allow government to select components on a build-as-you-go basis by adding components as new requirements need to be addressed[6]. SOA should be scalability, which allows the government to start small and grow as fast as the




```
<Envelope>
  <Header>
    <actor>http://...org/soap/actor/next</actor>
    <qos mustUnderstand="1">log</qos>
  </Header>
  <Body>
    <getMessage ns1-"urn:NextMessage" ...>
    <UserID type="string">JDoe</UserID>
    <Password type="string">OJDOEO</Password>
    </getMessage>
  </Body>
</Envelope>
```

business requires. Proposed architecture provides extensive support for business and IT standards to facilitate greater interoperability and portability between applications. Web service is defined as being network-accessible via SOAP and represented by a service description, the first three layers of this stack are required to provide or use any Web service [10]. The simplest stack would consist of HTTP for the network layer, the SOAP protocol for the XML messaging layer and WSDL for the service description layer. This is the interoperable base stack that all inter-government, or public, Web Services should support. Simple Object Access Protocol (SOAP) is a specification for the exchange of structured information in a decentralized, distributed environment [9]. Figure1 shows a conceptualized SOAP request message based on a scenario of a personal text message recorder, similar to a recording phone answering machine, where the user can listen to the recorded messages.

Figure 1. Example of a conceptualized SOAP message
(Source**:**WebSphere Application Server Version 6.1 and WebSphere Application Server Toolkit Version 6.1, IBM (2006))

XML documents are often used as a means for passing information between the service provider and service consumer. A sample portion of a simple XML document is shown in Listing 1, [9].

```
Basic XML document structure
1 <address country="USA">
2 <name>John Smith</name>
3 <street>43 Walcut St</street>
4 <city>Dublin</city>
5 <state>Ohio</state>
6 <postal-code>45561</postal-code>
7 </address>
```

SOA for the government interoperability should be built using web service standards, which already enjoy great popularity and are accepted in the industry. Above standards provide higher interoperability and adequate protection from 'locked' software, typical for vendors. Web services as self-describing components will enable their own services through specific Internet protocols[7]. The communication via Internet protocol means openness and availability of those services. Users of the system can access the registered web services from any computer connected via an Internet Network. The only that needs to be known is how to find the service we are looking for, i.e it is necessary to know how to discover it. Such discovering of web services may be performed with the help of UDDI (Universal Description, Discovery, and Integration) registers. Figure 2 presents the proposed logical architecture of the system which will support government interoperability.

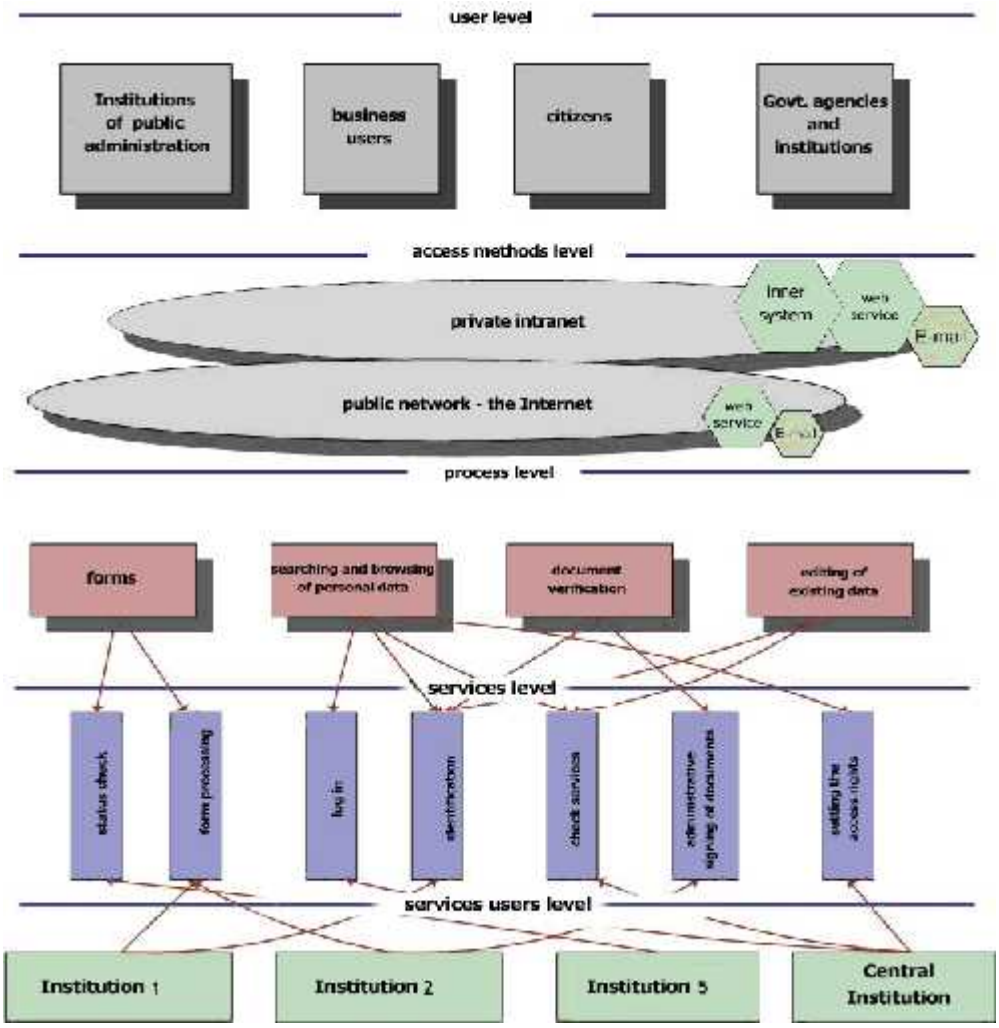



Figure 2. Logical architecture of the interoperability system

Proposed architecture uses a layered model and consist of the following layers (levels):

1. User level
2. Manner of access level
3. Process level
4. Service level
5. Level of service providers

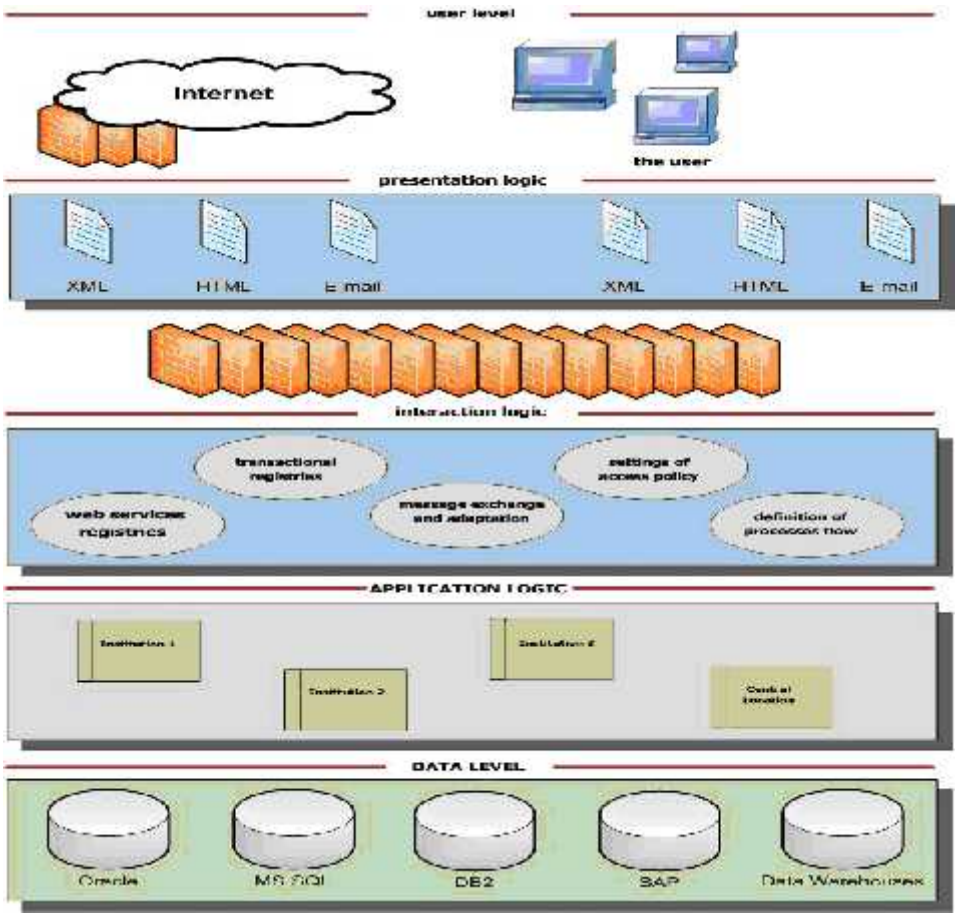
**The user level** in the proposed architures provides access through the use of adequate presentation logics to the SYSTEM for all types of functional and system users.

In line with the safety policies these users have different access manner to the System. The manner of access is defined in the level of access manner. It can be via the public and open Internet network (e-mail, publicly available web portals of the SYSTEM), a private intranet network (via web sites, e-mail as well as integration with the existing information systems in institutions). The access level, in line with the defined policies of types of users, also defines the access manner of each of the users.

## 5.0 SYSTEM ARCHITECTURE

Due to the complexity of the system, proposed architecture should be realized as a layered architecture model. At the lowest level we propose to have a distributed system of heterogeneous databases which have no mutual bases management system, and consequently, as such they are not of direct interest to the SYSTEM. Aberration from this statement is the database which is supposed to record the use of services provided by the SYSTEM. Access to the separate databases we proposed to be performed with the application logics of the modules which belong to the internal information systems of the institutions included in the interoperability process. Application level should provide interfaces to the level of integration logics. The general SYSTEM architecture is presented in Figure 3.

Figure 3. System architecture




The level of integration logics is of key importance for SYSTEM realization. This level is to provide the creating of services through media tools for work flow tracking which will be connected to the existing modules of the internal information system and their transformation into web services. The integration level should also publish the obtained web services in corresponding registers of web services depending on the access privileges. This level also manages the access privileges, as well as the exchange and modification of messages from diverse sources, in case there is a need for their converting into a comparable format. Finally, this level is in charge of management of the services offered by the SYSTEM in the shape of transactions if the need arises. In one sentence, this level provides the functionality of the services in the SYSTEM. Those services have to be available to various user categories. For the needs of SYSTEM protection, we propose a firewall to be installed behind this level after which the level of presentation logics follows. This level will be realized in the shape of a portal offering: list of web services through access to service registers, integration of the web services by e-mail or directly as a

remote procedural call of application (RCP) in a standardized format (XML), but also as an ordinary HTML text for a certain set of services, i.e users. In order to be able to have high level of security and achieving maximum protection from unwanted incursion into the system, we propose the external users of the system to be separated by another additional protective firewall.

## 6.0 NON FUNCTIONAL REQUIREMENTS OF THE INTEROPERABILITY SYSTEM

The non-functional requests of the system present criteria which point to the operative attributes of the system. Those are availability, confidentiality, interoperability, security and robustness of the system.

### 6.1 Availability

Availability is defined as time in which a certain service is accessible to its users (citizen, legal entity or public administration).The availability of the functions of the interoperability system can be presented with the availability of the services which at a certain moment are used by the participants in the system.

According to the interoperability design, the availability of each of the services depends on:

1. The consistence (hardware operability, the operative system and interfaces) of the service registers;
2. The quality of services of the network infrastructure
3. Information systems of the service providers

The detailed action plan should be elaborate to implement the following events (exceptions) in the specific solution:

1. Activation of the procedure for disaster recovery in case of failure of the service register or failure of a process of a service provider
2. Providing a notification to the user in case of ban to further use of the existing service (due to a change in the legal regulations)
3. Providing of an alternative service, or automatic recommendation for request of a new service, in case of inavailability of a certain service
4. The pervious events should also function in case of complex services (provided by several service providers)

### 6.2 Confidentiality

Confidentiality presents a measure of authenticity of the information. The design of the interoperability system architecture provides the application of safety measures and policies which increase the confidentially of information. The overall communication between the interoperability system and the existing national system should be confidential and protected. That is why the traffic has to be

cryptographically protected, so that it would be impossible to correctly interpret the messages even if they are discovered. For this purpose mechanisms must be implemented such as application for cryptographic protection SSL or TLS-based products (e.g. RSA, PGP). Protection mechanisms are also to be provided for the protection of the communication between the interoperability system and the external users. Regarding the fact the interoperability system infrastructure will include Internet as manner of access to services, the safety of the communication must be guaranteed through appropriate security mechanisms.

### 6.3 Interoperability

Interoperability is a system characteristic which points to the possibility of connecting different systems with the purpose of information exchange and use of that information.

The interoperability of functions in such a system can be represented with the independence of services through which the users exchange data from various existing systems and they are used by the participants in the system.

The service oriented architecture itself, envisages interoperability as a basic concept in the functioning of the system.

### 6.4 Security

Security must be factored from the beginning of an SOA, from model, assemble, deploy, and manage. This is a very important point and one aspect that are easily overlooked in a SOA. SOA security requirements should be included through all of the SOA implementation layers. Beginning at the service consumers, through the business processes, service definitions, service components, and finally service implementation, security needs to be considered.

SOA security includes:[8]

6. Identity
7. Authentication
8. Authorization
9. Auditing
10. Confidentiality, Integrity and Availability
11. Administration and Policy Management

### 6.5 Robustness

Robustness is a system characteristic presenting the ability of the system to respond to changes, obstacles in the operability or other abnormal behaviours which could cause halt in the operation of the system.

# 7.0 CONCLUSION

- SOA proves that is in line with service delivering task within and out of the government institutions. Using such kind of system, government institutions, citizens and businesses will have securely access of all the registers and databases in the state institutions via single connection, using predefined, regulated procedures, schemes and protocols.
- SOA can operate on independent platform and location, these advantages enables development of systems which will be able to serve all government institutions. SOA will eliminate data and business process redundancy across state institutions.
- We conclude that SOA meet the requirements needed to have cross institutions interoperability and is big step further of effective way of government administration in technical and business point of view.

# REFERENCES

[1] An analysis of Service Oriented Architecture -Rajiv Ramaratnam, Massachusetts Institute of Technology (2007)

[2] Web services and Enterprise Application Integration - Chris Wong

[3] Elements of a Successful Enterprise Application Integration (Artikull Syntel)

[4] Guidelines for Application Integration (Patterns & Practices) - Microsoft Press 2003 Microsoft Corporation Version 1

[5] SOA Approach to Integration: XML, Web Services, ESB, and BPEL in Real-World SOA Projects-Matjaz, B. Juric (2007)

[6] Service-Oriented Architecture: Concepts, Technology, and Design - Thomas Erl Pretence (2005)

[7] Web Services Analyst Perspectives (2006).

[8] Understanding SOA Security Design and Implementation, IBM 2007.

[9] SOA-Based Enterprise Integration: A Step-by-Step Guide to Services-Based Application Integration by Waseem Roshen McGraw-Hill/Osborne 2009.

[10] Web Services Conceptual Architecture, IBM 2001