A SOA-Based e-Government Data Integration

Suhail Madoukh[1] and Rebhi Baraka[2]
[1]Ministry of Telecommunications and Information Technology, Gaza, Palestine
[2]Faculty of Information Technology, Islamic University of Gaza, Gaza, Palestine

Abstract: *Data Integration presents a core issue in the Palestinian e-Government Technical Framework. The currently used data integration model relies on the Integrated Central Database which lacks quality attributes such as: interoperability and flexibility. We purpose a SOA-based approach for data integration that achieves the above attributes. We present and analyze the current architecture and implementation of the Palestinian e-Government Integrated Central Database model. We transform the current model into a SOA framework that is realized using Enterprise Service Bus (ESB) and Web Services. The proposed framework offers database replication and connectivity functionalities for the Central Database. The proposed framework is evaluated using a scenario-based software architecture evaluation method and proves that it achieves the framework goals of quality attributes: interoperability and flexibility. Moreover, a prototype of the framework is implemented and validates the framework correctness. A specific usage is presented and further proves that the framework accomplishes its functionality and quality attributes.*

Keywords: *e-Government, SOA, Enterprise Service Bus, Web Services, Architecture Tradeoff Analysis Method, Database Replication.*

Received November 11, 2012; Accepted March 13, 2013

IAJeT

1. Introduction

The Integrated Central Database is one of the core components in the Palestinian e-Government Technical Framework [2]. The currently implemented architecture of the database relies on replicating subsets of the government institutions databases into the Integrated Central Database. The main functionalities of the Integrated Central Database are replication and accessibility. Both functionalities suffer from the lack of vital features which are: i*nteroperability, flexibility and manageability*.

Problems of the current architecture appear when trying to replicate a government institution database with a different type database of the Integrated Central Database. The problem also appears when clients trying to access the Integrated Central Database over a transport, driver, or an API that is not natively supported by the Integrated Central Database. Another problem emerges from the inability to attain a central point of management for the operation of the Integrated Central Database.

The above shortcomings can be overcome by transforming the current architecture into a SOA-based one. Even though many researchers proposed SOA-based approaches for the data exchange and integration issue, their solutions have not addressed all three problems mentioned above and have been based on requirements that do not completely match our case.

The problem of this paper is how to build a SOA-based framework for the Palestinian e-Government Integrated Central Database that achieves *interoperability, flexibility, and manageability*. SOA provides a solution to shared and distributed services and it achieves high interoperability, flexibility, and standardization by utilizing the description, discovery, and invocation of services. We propose to use Web Services in the SOA architecture. Using Web Services in e-Government enables government institutions to provide additional services by defining a new service that emerges from other e-Government services [12]. To realize the concept of SOA model, one would use the Enterprise Service Bus (ESB) [15, 16].

ESB acts as the middleware glue infrastructure that holds SOA parts together and integrates and manages the communication between different Web Services, applications, and data sources. The three elements: Web Services, SOA and ESB are the basis for realizing the e-Government Central Database framework.

The proposed framework is composed of the following components: Central Database Enterprise Bus, Service Registry, Government Informational Service, Database Replication Service, Management Service, Security Assurance Service, Database Management Adaptor Service, and Service Orchestration. Direct access to the Central Database is performed from one service only; this would allow other services a transparent access to the underlying Integrated Central Database, and hence need not be aware of the database type or how it is implemented. The proposed framework would achieve the interoperability and flexibility quality attributes.

The rest of the paper consists of the following sections: Section 2 presents related works and the application of SOA in similar domains. Section 3 addresses and analyzes both the current e-Government technical framework and the Central Database model. Section 4 presents the proposed SOA-based framework of the Central Database. Section 5 presents the prototype of the framework, section 6 addresses the evaluation of the framework, and finally Section 7 presents the conclusion and future work.

2. Related Works

On the investigation for the e-Government data integration, the authors in [2] propose a conceptual SOA-based framework for the Palestinian e-Government Central Database. In this paper we present the realization of that framework.

A SOA-based approach to data integration achieves interoperability. Works such as [4, 5, 7 and 10] have stressed the importance of interoperability for the data integration process where database accessibility should be based on neutral mechanisms and should be independent of the underlying implementation.

A technical concept that employs SOA as an IT platform to handle different modalities, devices, and data streams in the operation room is presented in [17]. They propose SOA for networked medical devices and integrating legacy medical devices in this network.

A model for a service-oriented e-Government support platform for the integration of application and data known as (SoGoSP) is proposed in [10]. It integrates applications and data from various business systems deployed in e-Government external networks and e-Government internal networks. The model consists of four layers which include service integration layer, service support layer, common service layer and application layer.

A solution to data integration problem between heterogeneous databases is presented in [6]. The solution is based on constructing data center with Web Service technique and XML schema which can give a good solution to problems with business logic method invocation and transparent data exchange in low layer.

The requirements of abstracting, sharing and integrating multiple heterogeneous information management systems in e-Government are addressed in [17]. They introduce architecture of e-Government information management platform based on SOA framework.

3. Palestinian e-Government Central Database Model

The Central Database is one of the core components of the Palestinian e-Government technical framework. The framework [2, 11] has four layers: the *front end user interface layer* which presents the access interface that the end user interacts with, the *common service layer* which provides front end services that commonly needed by e-Services, *the data access layer* which addresses database access gateway either centralized or decentralized, and the *infrastructure layer* which includes physical and low level software components. The Central Database is one of the components of the *data access layer*.

The overall e-Government technical framework is not fully implemented. Also the communication between layers is not well defined in terms of access protocols and standards.

The Central Database comes as a solution to sharing and integrating data between various ministries. Figure 2 depicts the current implemented and used Central Database model. It relies on database replication and synchronization techniques as low level infrastructure to maintain the Central Database and to keep its content up to date.

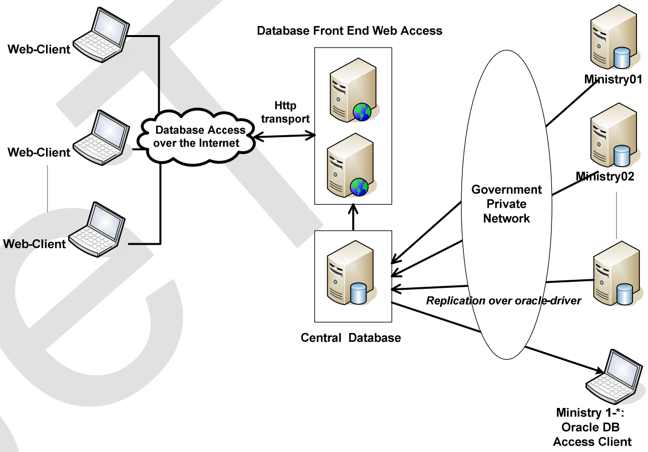
Web-Client
Web-Client
Web-Client
Database Access over the Internet
Http transport
Database Front End Web Access
Central Database
Government Private Network
Ministry01
Ministry02
Replication over oracle-driver
Ministry 1-*: Oracle DB Access Client

Figure 1. Current integrated central database model.

The main characteristics of this model can be classified into three categories as follows:

- Database Access: provides read-write for the owner ministry and read-only for other ministries. Access is done through the governmental private network using Oracle connectivity API with synchronous mode of invocation.
- Database Replication: facilitates replication between ministries databases and the Central Database using Oracle-based tools such as materialized views and database links.
- Management and monitoring: offers monitoring of the Central Database based on database parameters. Governance issues are limited to managing the main functionalities of the database access. Security policies are implemented at both network and database access level.

It is worth mentioning that there are no old fashioned legacy applications or rigid connectivity access modes to the databases. This makes SOA transformation of the Central Database model feasible. Next we state the limitations of the current Central Database model which make SOA transformation a necessity. They are as follows:

IAJeT

- Replication between Central Database and ministries databases can only be achieved through Oracle databases. This imposes inflexibility tights ministries to using proprietary databases.
- Access to the Central Database is restricted to Oracle connectivity API which decreases the level of interoperability.
- Direct access to database procedures is achieved only through the government private network and through oracle standard SQL port which undermines the flexibility and accessibility.
- Central Database has a read-only access mode which undermines the capabilities of the database.
- Absence of a standard way for describing, finding and invoking procedures defined in the Central Database. This leads to less flexibility and more management overhead.
- Monitoring, management and security assurance is performed through Oracle built-in utilities. This tightens the Central Database to specific technology.
- Governance measures are limited to specific functionalities of the database access. It should be database-wide.

These limitations affect the interoperability and flexibility attributes that are necessary for governmental data integration in the Central Database. Section 4 presents a SOA-based solution that overcomes these limitations and achieves interoperability and flexibility attributes.

4. The Proposed SOA-Based Framework

SOA is adopted in the transformation of the Central Database due to its open architecture and standards that cope with heterogeneous systems and applications with high degree of interoperability and flexibility. A typical architecture of SOA includes three main roles that interact using standard messaging. The roles are *service provider*, *service registry* and *service client* [14]. The service is first published by the *service provider* to the *service registry* which is a repository that holds service interfacing information. The *service client* searches the *service registry* for a specific service, and gets its binding information. The client uses binding information to consume the service provided by the service provider.

Web Services are used as a tactical realization of the SOA architecture. A Web Service is a software component that another software application can access automatically on the Web [14]. The fast adoption of Web Services emerged from the maturity of XML-based Web Service standards such as SOAP and WSDL [1]. ESB provides the SOA solution with the necessary integration infrastructure for Web Services [15]. It integrates applications, services, and the registry. ESB is event driven and provides standard messaging between services. It routes and transports service requests to the appropriate service provider [9][14]. A typical ESB [13] provides capabilities such as: routing, message transformation, protocol transformation, service mapping, Service orchestration, transaction management, and Security [5, 14].

4.1. SOA-Based Integrated Central Database Requirements

To overcome the shortcomings (Section 3) of the current model, the requirements of the SOA-based Central Database are specified as follows:

- Accessibility Mode: The Central Database accessibility should be based on standard connectivity rather than proprietary commercial software access mode, such standards are XML, SOAP and WSDL. In this case, services would access the underlying database without using its access driver and hence services would be database independent.
- Replication: Government ministries need to replicate and synchronize heterogeneous databases such as Oracle, MySQL, MS-SQL and MS-Access with the Central Database. Hence different replication options should be provided for the diverse types of used databases and only the replication service should be aware of such diversity.
- Management and Monitoring: Management and Monitoring should be separated from the application logic and the database procedure accesses. Logging and performance recording should be also implemented.
- Governance: The Central Database should operate around the hour since individual ministries IT infrastructure lacks the ability to do so. The Central Database should be governed through QoS and Service Level Agreement (SLA) since different ministries rely on it for providing e-Services which should operate without interruption as well as provides a level of responsiveness.
- Security: Security must be assured and should be managed centrally and imposed on all database accesses through the different services. Security policies to be defined and enforced and must not be configured at the underlying database level only, but also at the service level.
- Reachability: Access to the Central Database should be allowed to both government and non-government institutions. The access should be allowed through the government private network as well as the Internet.

4.2. SOA-Based Integrated Central Database Architecture

To satisfy the requirements discussed in Section 4.1 for the proposed Central Database, we present the SOA-based framework which is realized using Web Services and ESB as depicted in Figure 4.

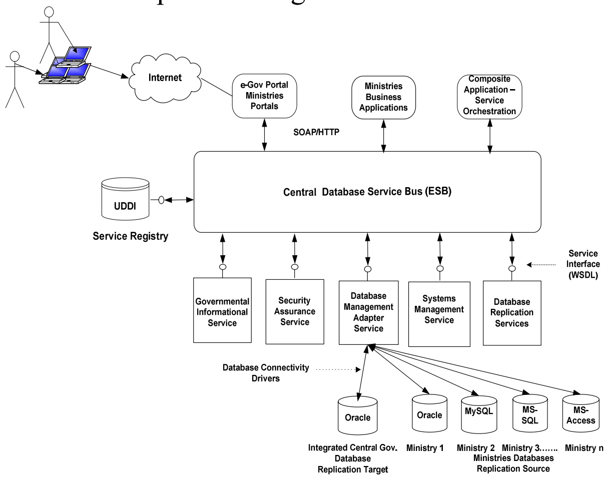

Figure 2. The SOA-based integrated central database framework.

A number of components constitute the proposed framework. Each component satisfies one or more of the requirements and leads to the achievement of the goals of the framework.

- Central Database Service Bus: Is considered the central platform of integration between different Web Services, and provides routing and transportation features for Web Service requests as well QoS feature for the framework. It is used and accessed by government institutions over the government private network as well as over the Internet for non-government institutions. It satisfies the reachability requirement of the framework.
- Service Registry: Is used to provide a search point of access to services and database definitions and metadata for all services provided by the Central Database model. The registry is based on Universal Description Discovery and Integration (UDDI).
- Government Informational Service: These Web Services provide access to basic informational queries and allow consumers to benefit from the government Central Database along with its presentation logic. This reliefs them from invoking services that interacts directly with the Central Database and return record sets that need to be manipulated by the developer. An example Web Service is a one that returns the social information of a citizen or the administrative record of an employee.
- Service Orchestration: This component is responsible for managing composite services. The composite service is invoked by a client and in turn it invokes and orchestrates different services to achieve the requirement of the composite service.
- Database Management Adapter: This adapter allows the Central Database Service Bus to accept requests for data sources from client systems and then invokes the relevant adapter to retrieve the data and return it in a standard format to the requester. It is used to hide the database management details from the rest of the Web Services. It communicates directly with the underlying data sources and provides database specific connectivity capabilities. This component achieves the accessibility requirement.
- Database Replication Service: This service is used to manage replication between the Central Database and ministries databases. It handles connection types, mode of replications, access permissions. This service achieves the replication requirement.
- Systems Management Service: This service is used to manage and monitor the Central Database service bus, and Web Services. It collects metrics, provides framework performance reporting capabilities. It achieves both governance and management requirements.
- Security Assurance Service: This service insures that security policies are adhered to. It is invoked by different services to add security layer to their functionality. Provided security functionalities include authentication, authorization, and non-repudiation. This service achieves the security requirement.

The interaction among these components is performed through the Central Database Service Bus which integrates the components and acts as the glue that connects them together. It routes, transports, formats requests and responses of services and provides service discovery through the registry.

5. The Framework Prototype

To provide a proof of concept of the framework, a prototype is implemented. The prototype provides a specific usage scenario for the framework and hence validates the requirements of the framework. Figure 5 depicts the top-level run time view of the prototype architecture. From the end user perspective, a user accesses the front-end Web interface which provides an access to the services of the Integrated Central Database (GovDB). The Web interface runs in the context of a Web application which interacts with the framework over Web Service interfaces. The functions available at the front-end interface are accessed through the Informational Services. For example they query the Central Database for citizen records and trigger the database replication service.

The prototype runs in the context of the Java Business Integration (JBI) environment which is the realization of the ESB. The ESB JBI environment used is OpenESB which is an open source ESB implementation (java.net/projects/openesb/). JBI includes the Business Process Execution Language (BPEL) Service Engine, Database Binding Component, and Composite Applications for

IAJeT

Replication and Informational Services. The implemented Informational Services are four: the Security Assurance Service (SecuritySvc), the Citizen Data Service (CtzDataSvc), Employee Data Service (EmpDataSvc), and Health Insurance Data Service (InsuranceDataSvc).

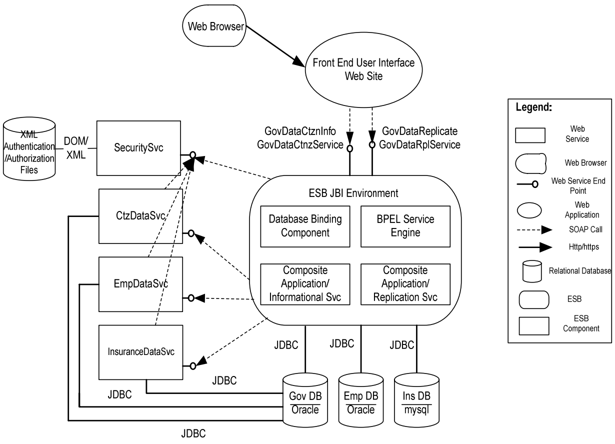

Figure 3. Prototype architecture.

The prototype deals with three databases: The Governmental Database (GovDb) which is Oracle 10g, the Employee Records Database (EmpDb) which is Oracle 10g and its data source is the General Personnel Office, and the Health Insurance Database (InsDb) which is MySQL and its data source is the health insurance system. Access to these databases is done over JDBC. The interaction between Web Service consumers and providers is done over SOAP/HTTP transport. Direct access to GovDb is performed using stored procedures which provide more security where database access is not allowed directly and tables and views are not exposed to clients; only predefined procedures with specific input/output are allowed in accessing the GovDb.

The prototype architecture can be associated with the SOA-based Framework proposed and explained in Section 4.2. The framework as seen in Figure 4 is composed of different components that can be mapped to the parts of the prototype. The importance of the association between the prototype and the framework is that it verifies that the prototype implementation scope includes the main components of the framework. The parts of the prototype and their counter parts in the framework are as follows:

- The ESB JBI Environment corresponds to the ESB.
- The Composite Application/Informational Service corresponds to the Service Orchestration.
- The Front-end interface corresponds to the e-Gov Portal.
- The services CtzDataSvc, EmpDataSvc, and InsuranceDataSvc correspond to Informational Services.
- The SecuritySvc corresponds to Security Assurance Service.
- The Composite Application/Replication Service corresponds to the Replication Service.

6. The Framework Evaluation

The framework evaluation is based on Architecture Tradeoff Analysis Method (ATAM) [8] in which the architecture should be analyzed to disclose its strengths and weaknesses in light of quality attributes, while eliciting any risks [3]. The quality attributes we are evaluating the proposed framework for are: interoperability and flexibility. The methodology for evaluating the framework depends on setting scenarios for these goals. The requirements to conduct the evaluation are evaluation team and stakeholder staff. The Evaluation team typically probes the architectural approaches used to address the important quality attribute requirements specified in the scenarios. The goal is to assess whether these quality attribute requirements can be met. In our case the evaluation team is the administrators of the current Integrated Central Database model and those use it in their applications and systems within the governmental institutions. Figure 6 depicts the overall process performed for performing the framework evaluation based on ATAM method. The evaluation process relies on the evaluation teams, business drivers and constraints, the framework quality attributes, and framework architecture approach. The results of the evaluation are the scenarios and how far the quality attributes are fulfilled. If the scenario presents a non-risk for the quality attributes, the framework is considered achieving the quality attribute.

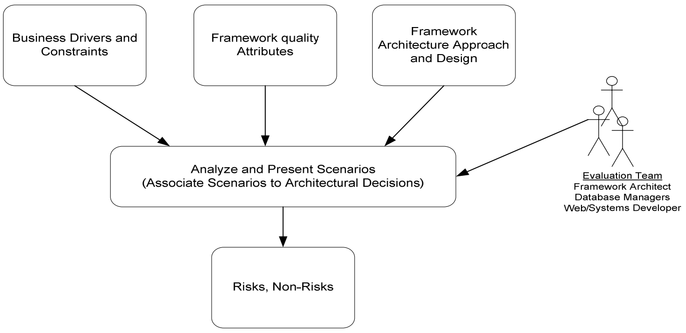
Business Drivers and Constraints
Framework quality Attributes
Framework Architecture Approach and Design
Analyze and Present Scenarios (Associate Scenarios to Architectural Decisions)
Evaluation Team
Framework Architect
Database Managers
Web/Systems Developer
Risks, Non-Risks

Figure 4. Framework evaluation based on ATAM.

The main features that support each of the quality attributes are presented which are inducted from the scenarios. Table 1 and Table 2 present the main features of the framework that provides interoperability and flexibility enhancement. Features listed in both tables are the summaries for the scenarios based on ATAM methodology.

The features listed in both tables show that the framework achieves its goals which are interoperability and flexibility. The interoperability, which allows using diverse type of database, is achieved by having different database types as part of this system and can be part of the replication as well

IAJeT

Table 1. Framework Interoperability Supporting Features.

#	Interoperability Feature
1.	The framework supports diverse services implemented in various platforms and languages.
2.	The framework allows replicating heterogeneous database types.
3.	The framework is using BPEL for business process which can orchestrate Web Services that uses SOAP and WSDL for service interfacing regardless of the underlying platform or development languages.
4.	The framework allows having service users and providers to use different implementation languages and platforms.
5.	The middleware integration approach used in the framework is ESB. It allows connecting diverse applications, technologies, and data formatting.
6.	The authentication mechanism is centralized and realized using Web Service.
7.	The framework is designed to support standard message-level security; but it is left to the service providers to implement such features to further enhance the security of the service.
8.	ESB is responsible for integrating legacy systems to the framework, which provides interoperability with old legacy applications.
9.	The standards used in the framework and provide interoperability between framework components when interacting with each others are: WSDL, SOAP, UDDI, and BPEL which provide capabilities to systems developed with Web Services technology.
10.	Not all Web Services platforms implement the same version of the additional standards such as UDDI, BPEL, WS-Security and hence achieving interoperability faces some obstacles when using such standards. But since the framework is under a centralized administration, these obstacles can be mitigated.

Table 2. Framework Flexibility Supporting Features**.**

#	Flexibility Feature
1.	Since the framework is composed of diverse components, most of them are Web Services, which are self-contained and loosely coupled; the changes required for any service do not affect other services.
2.	Services in the process (composed services) can be changed without affecting other services in the BPEL workflow, as far as input/output types of the service are unchanged.
3.	The identity information is not hard-coded in security services implementation. Depending on the realization of the Web Service, access information can be stored in an XML file or a database source.
4.	The Web Services to be implemented are coarse grain and hence self contained and can operate independently, so they provide loose coupling and enhances flexibility.
5.	The credentials are used in Security Assurance Service; they are easily managed and not hard coded. They are stored in an XML format, which is flexible for manipulating and easy for understanding.
6.	Access to services by consumers can be from the Internet as an open network as well as from the private governmental network.
7.	Using a new data source in the framework or adding another database type can be achieved with little efforts. A minimal change is required in the code that access the database if such database connectivity is not supported by the JDBC.
8.	The framework provides support for both synchronous and asynchronous Web Services. It is left for the requirement and operation of the service to use either of them. The services implemented in the prototype are synchronous services.

as resource for different governmental information services. The flexibility, which allows different ways for performing a specific task, is achieved by accessing the information services over HTTP transport which generally uses port 80 which is normally not filtered by internet firewalls, and hence the access to the Central Database can be both from internal government private network as well as over the Internet. Section 7 presents a usage scenario of the framework and outlines the interaction between different components of the framework.

7. A Usage Scenario

Here we consider a usage scenario of the implemented prototype of framework. Figure 7 depicts the flow of events in this scenario and the interaction direction between different components.

- The end user, e.g. a citizen, would like to check his social status in the Citizen Population Registry, for example, in order to verify that the birth data of his new born baby is added to the social section of his identification card. He accesses the citizen information section of the e-Government portal using his login credentials.
- The Web application running at the portal would use SOAP messaging and HTTP to invoke, say, the *Citizen Information Operation* which is part of the *Government Informational Services - CtzDataService*
- The *Government Informational Service* would then interact with the *Security Assurance Service* to insure security policies and access for this context is allowed.
- The *Security Assurance Service* and based on username/password pairs and IP authentication would permit the *Government Informational Service.*
- The *Government Informational Service* would invoke the *Database Access Service* for information retrieval from the database.
- *The Database Access Service* accesses the Central Database using the JBI database binding component or the database driver.
- The database returns the requested records to the *Database Access Service.*
- The *Database Access Service* returns the response to the *Government Information Service.*
- The *Governmental Information Service* manipulates and processes the results and returns it back to the Web application at the e-Government portal
- Finally, the Web application formats and presents the required information to the citizen.

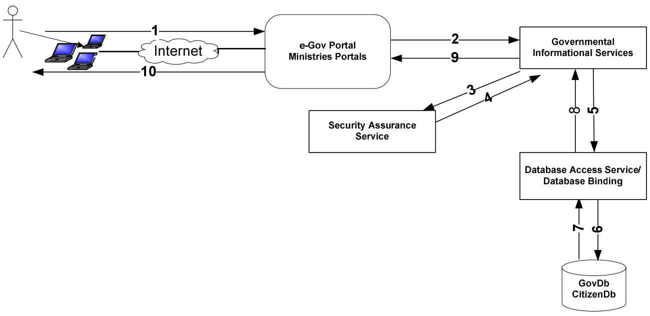

Figure 5. Usage Scenario for the Proposed Framework.

IAJeT

The presented scenario outlines how the implemented prototype fulfills one of the main functional requirements, namely the accessibility, as well as two of the quality attributes which are interoperability and flexibility. First, interoperability achievement is clear in this scenario, this is because the Web application and the Government Informational Service are database type independent, and so if the low level database that holds the Citizens Population Registry is changed from e.g. Oracle to MySQL, then change is not required for the Web application that access the Governmental Database. Second, flexibility is achieved by using standard HTTP transport to carry messages between the Web application in e-Government portal and the Government Informational Service (either over the government private network or the Internet); the HTTP transport is generally allowed and not filtered by firewalls; where in the current model of Central Database such access would be carried over Oracle-Sql port which most of the time needs security reconfiguration to allow it, also the portal access to the Central Database is restricted to be from the government private network.

8. Conclusions and Future Work

The current Integrated Central Database model, a core part of the Palestinian e-Government Technical Framework, was presented and analyzed. A new Central Database model based on SOA solution was proposed that overcomes the shortcomings of the currently used model that lacks interoperability and flexibility. The proposed framework is based on SOA and was realized using an ESB and Web Services. The framework provides the main functionalities which are the access to the Integrated Central Database, and the replication between the diverse database types. The main components of the framework are: ESB, Web Services, databases, e-Government portals, governmental business applications, and front-end applications.

In the evaluation of this work, two methods were used: ATAM based and proof-of-concept. The ATAM based evaluation method was used to evaluate the framework architecture. The proof-of-concept was realized through implementing a prototype for specific functions of the framework. The implementation included a usage case of three informational services, replication service, security assurance service, and service orchestration using BPEL for the informational services. The prototype was the proof-of-concept to validate the solution of the framework and showed it accomplishes its requirements.

The main contribution and impact of this research is to show that SOA solutions can be applied to the Integrated Central Database model, also to align SOA concepts to the e-Government domain problems.

Future work for the proposed framework would be fully implementing the framework. Enhancing the framework by adding support of services auto-composition. Providing semantic support and capabilities to the framework. Adding support for integrated and distributed database backend, instead of having just one Integrated Centralized Governmental Database. Enhancing the framework to support and achieve the following quality attributes: manageability, availability, reliability and fault-tolerance.

References

[1] Alonso G., Casati F., Kuno H. and Machiraju V., “Web Services: Concepts Architecture and Applications,” Springer, Berlin, 2004.

[2] Baraka R., and Madoukh S., “A Conceptual SOA-Based Framework for e-Government Central Database,” International Conference on Computer, Information and Telecommunication Systems, Amman, Jordan. May 14-16, 2012.

[3] Bianco P., Kotermanski R., and Merson P., “Evaluating a Service Oriented Architecture,” Software Engineering Institute, 2007.

[4] Chen D., “Enterprise Interoperability Framework”, In Proceedings of Enterprise Modeling and Ontologies for Interoperability EMOI - Interop, vol. 200, 2006.

[5] Curl A., and Fertalj K., “A review of enterprise IT integration methods,” Proceedings of the 31st International Conference on Information Technology Interfaces (ITI '09), Dubrovnik, pp.107-112, June 22-25, 2009.

[6] Guo Q., Zheng H., Lie J., and Wang X., “Design and Implementation of Data Management Center Based on Web Services,” Proceeding of the Ninth International Conference on Hybrid Intelligent Systems (HIS '09), Shenyang, vol.2, pp. 322-326, Aug 12-14, 2009.

[7] Josuttis, N., “SOA in practice The Art of Distributed System Design,” O'Reilly, 2007.

[8] Kazman R., Klein M., and Clements P., “ATAM: Method for Architecture Evaluation,” CMU August 2000.

[9] Keen M., Adinolfi O., Hemmings S., Humphreys A., Kanthi H., and Nottingham A., “Patterns: Implementing an SOA using an Enterprise Service Bus,” IBM Redbook, 2005.

[10] Ma H., “A Service-oriented e-Government Support Platform for Integration of Application and Data,” Proceeding of the Second International Conference on Information

IAJeT

Technology and Computer Science (ITCS), Kiev, pp. 398-401, July 24-25, 2010.

[11] Madoukh, S. and Yaghi, S., "Technical Framework for Palestinian e-Government," IT-Forum-3, Islamic University of Gaza, Gaza Strip, May 12-13, 2009.

[12] Medjahed B., Rezgui A., Bouguettaya A., and Ouzzani M., "Infrastructure for e-Government Web Services," IEEE Internet Computing, vol.7, no.1, pp. 58- 65, 2003.

[13] Ortiz, S., "Getting on Board the Enterprise Service Bus," Computer , vol. 40, no. 4, pp. 15-17, April 2007

[14] Papazoglou M., "Service-oriented computing: concepts, characteristics and directions," Proceedings of the Fourth International Conference on Web Information Systems Engineering (WISE 2003), pp. 3-12, Dec 10-12, 2003.

[15] Papazoglou M., Traverso P., Dustdar S., and Leymann F., "Service-Oriented Computing Research Roadmap," International Journal of Cooperative Information Systems, vol. 17, no. 2, pp. 223-255, 2008.

[16] Strahle M., Ehlbeck M., Prapavat V., Kuck K., Franz F. and Meyer J.-U., "Towards a Service-Oriented Architecture for Interconnecting Medical Devices and Applications," Joint Workshop on High Confidence Medical Devices, Software, and Systems and Medical Device Plug-and-Play Interoperability (HCMDSS-MDPnP), Boston, MA, pp. 153-155, June 25-27, 2007.

[17] Yunliang J., Xiongtao Z., Qing S., Jing F. and Ning Z., "Design of E-Government Information Management Platform Based on SOA Framework," First International Conference on Networking and Distributed Computing (ICNDC), Hangzhou, pp. 165-169, Oct 21-24, 2010.

Suhail Madoukh received his M.Sc. in Information Technology from the Islamic University of Gaza in 2011 with excellent degree, and B.Sc. in Computer Engineering from the Middle East Technical University in Turkey in 1994 with high honor. He works as an Assistant Deputy in the Ministry of Telecommunication and Information Technology, Palestine. He has over 18 years of expereince in the filed of IT and Telecom Management, IT Training, Computer Networks and Systems Administation.

Rebhi S. Baraka is an assistant professor of computer science and vice dean of the Faculty of Information Technology at the Islamic University of Gaza, Palestine. He received his PhD degree in Computer Science in 2006 from Johannes Kepler University, Austria, He received his M.Sc. degree in Computer Science in 1996 from De La Salle University, Philippines. He received his B.Sc. degree in Electronics and Communications Engineering in 1991 from University of the East, Philippines. His research activities are in the area of Distributed Systems, Web Services Semantics, Semantic-based Discovery of Web Services, Arabic Ontology Building, and Semantic Web.