An Overview Description of a Multi-channel eGovernment Open Interoperable Architecture – Communication Technologies and Constraints

Panagiotis Germanakos, Eleni Christodoulou, George Samaras
Computer Science Department, University of Cyprus
{pgerman, cseleni, cssamara} @cs.ucy.ac.cy

Abstract. The constant development of eGovernment in Europe and the rest of the world suggest a potential for more efficient, user-centered and multi-channel delivery of quality public services for all. It is an imperative requirement, nowadays, for the development and implementation of a common open platform that will enhance Cross-National eGovernment e-services and advance user interactivity. The key role of this platform is to provide all the necessary technological communication capabilities whereby all the governmental, external and business bodies will offer e-services in an integrated and interoperable manner. This paper will present an overview of an open interoperable architecture complying with eGovernance policies and strategies and identifying all technological mechanisms and procedures that will allow speedily transmission of information among Cities' G2G, G2B and G2C.

1 Introduction

In the new European socio-economic, governmental, and mobile reality, a common technological infrastructure that could link all the governmental and business organizations as well as the public, not only inside a country's borders but within the community as well has become a necessity [1], [2], [3]. Therefore, systems and architectures that can work together in an integrated and interoperable manner enabling secure delivery of customer-centered and multi-channel services [4], [5] through a single point of access [6], [7] in the area of eGovernment have to be developed enabling apt communication at all levels. In particular, an infrastructure including the identification of processes and mechanisms, with the latest technologies and advancements incorporated (wired and wireless), supporting the application and openness of current and future government platforms to transmit speedily and cost effectively information between Governments (G2G), Government and Businesses (G2B) and Government and Citizens (G2C) as well as supporting the type of applications installed for the interactive and efficient communication between the participating actors has to be identified.

The proposed technological infrastructure is designed to comply with a general legal and procedural framework as well as with a number of principles, rules and regulations [8] at a European level such as delivery of all the services 24 hours a day and seven days a week; improve of on line public service delivery for citizens and businesses by making it faster, more convenient, less constraining, more transparent and more user-friendly; simplification of administrative procedures and reduction of bureaucracy for citizen and business; share and exchange of data and information across governmental sectors; interoperability and cooperation between all public administrations to provide integrated services across organizational boundaries; establishment of standards for interfaces between departments that permit efficient and transparent communication with the outside world; back office integration; protection of personal data and so forth.

This paper gives a broad eGovernment infrastructure insight taking into consideration all the aforementioned implications. It is structured in 5 sections. Section 2 outlines serious pitfalls and an

initial infrastructure approach. Section 3 presents a multi-tier architecture overview. Section 4 describes the eGovernment open interoperable infrastructure, and section 5 concludes this paper.

2 Relevant Pitfalls Identified and Initial Infrastructure Approach

The fact that common communication strategies and activities among the Cities, and furthermore Nations, participants have not yet been standardized has resulted to the creation of serious problems leading primarily to the lack of integration and interoperability of the various systems and platforms implemented. Among others, a predominant problem identified is the huge economic investments that have been made to the stand alone legacy systems providing a particular service and not being structured in a way that could be incorporated with others. The main concern is not only to identify ICT technologies that could be ideal for separate systems, but since there is the major need for intercommunication to build the technological base that those systems could be attached to and with some necessary alterations to become interoperable and integrated. On top of the above, noteworthy are also the different problems that mobility applications suffer today and could be summarized to:

- *Local mobility*. Some successful mobile applications have been delivered in a local environment, but are not cost-effective when applied at a trans-national mobility level.
- *Limited mobility*. An impressive coverage has been reached in most EU countries. The GSM technology is an EU initiative with a high degree of expansion. Nevertheless, local problems, non-covered local areas or not enough capacity can invalidate mobile services in concrete situations.
- *Closed mobility*. Mobile services are restricted in most cases to GSM technology (both CSD and SMS). There is not a generalized use of complementary mobile technologies for local environments as well as non-terrestrial areas.
- *Interrupted mobility*. There is a lack in the availability of frameworks that make possible business models in which complex interactions between different sectors are performed by means of mobile applications. Undeniably, as the new 3G technology is evolving rapidly, suitable platforms that make possible an interesting business model to access Pan-European mobile services is a 'must' that has not been resolved yet.

To begin with, the architectural design and further construction of a related common platform implies the total reengineering of back- and front-office processes [9], [10] so to take advantage of the full capabilities of the latest technologies that may involve multi-device, multi-channel, multi-network, multi-user, multi-service, multi-country and multi-platform: open-standards based, with key technological constraints extended to reliability, interoperability, effectiveness, efficiency, openness, compliance with EU standards, personalization, interconnectivity, Federated Service Provision, flexibility for advancements adaptations, high connectivity speed, broadband connection, security, integration, transparency, mobility / wireless, and expandability.

More broadly, the particular infrastructure is composed of three main parts:

1. A User Interface (front-end), providing *a single point of access* to informational, interactive and transactional public and business services.
2. An integration middleware, based in XML [11], messaging and Web services. The middleware layer represents the nervous system of public and business service delivery, enabling service / user requests and data to be assembled from across governments and dispatched accordingly, and enabling related transactions to be conducted.
3. The back-end layer, whereby the actual internal and external services are stored and related requested information retrieved.

Vital needs / challenges this infrastructure is covering, providing additionally an easy-to-use support for desktop and mobile content management and maintenance, are focused upon: *any network* (combining both wired and mobile, i.e. GSM / GPRS / UMTS, wireless LAN, PSTN / ADSL), *any channel* (i.e. Web, WAP) or *device* (i.e. mobile phone, PDA, PC), *any user* (i.e. any age, any culture, any expertise), *any place* (i.e. local, regional, national, European), *any service* (i.e.

platform that can be tailored to any specific vertical application), and *any situation* (i.e. Government-To-Government, Government-To-Citizen, Government-To-Business).

3 Multi-Tier Architecture Overview

Since the current infrastructure is based on a multi-tier architecture an overviewed reference to the specific architecture is considered necessary and is further presented in this section. A Multi-tier architecture builds separate layers into applications so that maintenance and development concentrates to a particular layer. In essence, multi-tier architecture allows the distribution of components across multiple servers and access of data that is stored in multiple databases.

There are no specific rules as to how many tiers to use or what each tier should actually do, but are specific to particular system design and implementation according to various user requirements. However, there are some principles that should be satisfied if a multi-tier environment is implemented: (a) *scalable architecture and location transparency* (it should be possible for each application layer to be located (distributed) on a physical different piece of hardware (server)), (b) *information exchange* (each layer should exchange information only with its successor or predecessor layer), and *communication interface* (each service within a layer should have a clearly defined way of invocation. This means that a communication interface should exist to exchange data between layers).

In further support and enlighten of the general flow of the proposed infrastructure, an insight regarding the main parts of such a system design is given below:

- Client tier: Receives HTML, WML, XHTML etc. and delivers a friendly user interface (e.g. a standard Web browser) or receives an XML like data structure in the case of Desktop clients, external systems, etc. for further processing. The client tier obtains the content from the lower layer utilizing HTTP / HTTPS, SOAP [12], UDDI [13], XML exchange, etc.
- Presentation logic tier: It handles requests / responses from / to multiple clients. This is where HTML, WML, etc. are rendered and delivered for the presentation in the client tier. It can also use XML data to communicate with desktop clients, third party applications, etc.
- Application middle tier: It connects to any data repository (database, LDAP, file system) and gets data, which it manipulates and transforms according to business rules. The middle tier receives requests from the client application over the presentation tier, and retrieves data from the data repository. Business logic is the code that processes retrieved data according to the requests received from the client application. The Presentation and Application middle tier are usually encapsulated in the server components.
- Application Back-end: Usually it consists of the various data repositories (database, files system, LDAP), which exist separately from the client and the middle tiers.

The basic advantages in comparison to the traditional 2-tier approach are in the performance and scalability. Multi-tier architecture performs substantially better than the 2-tier architecture and is much more scalable. The performance and scalability improvements are a result of moving the business logic from the client to the server where it is possible to perform tasks in parallel, thus getting the most out of server and network resources.

Moving the business logic to the servers has also considerable security advantages. Local communication between servers is more secure than communication with the client workstations. Furthermore, it is much easier to control access to sensitive data since requests originate from known sources. Consequently, critical business processes that handle sensitive data are run on the server.

An additional advantage of the multi-tier architecture is that modifications to the business logic require minimal or no changes at all to the user interface or the database. The most dominant platforms suitable for multi-tier application development currently in the market are .NET [14] and J2EE [15].

4 eGovernment Open Interoperable Infrastructure Description

Based on the abovementioned considerations, a presentation of an eGovernment multi-channel infrastructure is presented trying to convey the essence and the peculiarities encapsulated, and further answering to the question why such a technological platform approach it could be proved most appropriate and integrated for the satisfaction of the citizens' and business' needs and requirements on the services level.

The current architecture, depicted in Fig. 1, is composed of three interrelated parts / tiers. Each *tier* for the purpose of the infrastructure functionality may be composed of *components* and each component may be broken down into *elements*, as detailed below:

4.1 Front-End Tier

It is the primary tier and user access interface of the system directly communicating with the Middleware exchanging multi-purpose data. It consists of four components each one assigned for a different scope:

- *Multi-Device*: Enables the attachment of various devices on the infrastructure, such as mobile phones, PDAs, desktop devices etc. identifying the characteristics of the device and the preferences as well as the location of the user (Personalization / Location based).
- *User Customization layer*: This component comprises of all the access-control data (for security reasons) and all the information regarding the user profile. It is based on LDAP standards. The type of information that is associated to the user concerns: preferences, geographical data, device model, age, business type, native language, context, etc.
- *Multi-Service access point*: It is the enter point for the user enabling the login to the gateway. This component is directly communicating with the Authentication & Authorization component of the Middleware where the actual verification for the user is taking place. Through this single point the user has access to any service.
- *Multi-channel*: Due to the variety of multi-channel delivery i.e. over the Web, telephone, interactive kiosks and so on, this component consists of the elements that will make use of the different characteristics of the channels and will be the components in charge of handling requests of the services associated to each of the applications.

4.2 Middleware Tier

This is the main tier of the architecture. At this level all the requests are processed. This processing varies from security, to authentication, to integration, to interoperability and so forth. This tier accepts requests from the Front-end and after the necessary processing; either sends information back or communicates with the next tier (Back-end) accordingly. The Middleware is comprised of the following components:

- The first component is made up of eight elements each one responsible for executing a different job. These elements are:
  - *Identity management*: It is responsible for the management of a user's identity entering the system and what resources that person has access to. The backbone of identity management is a system of directories and directory-enabled applications.
  - *Personalization / location based*: It is responsible for the custom tailoring information to the user. To Web based applications, it returns a page that has been customized for the user, taking into consideration their habits and preferences. The personalization

may be done by the user, the system or both. The notion "location-based" is referring mostly to mobile systems that support user identification based on its location .

- *Session / Transaction Control*: This element analyzes and controls all the parameters associated with every link opened between the platform and a user. It is responsible for resuming sessions and contexts when necessary and establishing adequate communication with other components such as multi-device and multi-channel residing at the Front-end.
- *Routing & Messaging*: It is a broker that works with existing messaging transports in order to add routing intelligence and data conversion capabilities. A rules engine analyzes the messages and determines which application should receive them, and a formatting engine converts the data into the structure required by the receiving application.
- *Authentication & Authorization*: The verification of a user's identification number or password that is used to gain access to the system granting them the right or permission to use the assigned system resources (direct communication with the Multi-Service access point component of the Front-end).
- *Security & Certification*: On the infrastructure level, it is the protection against unauthorized access to the system and the assurance that programs or routines running will be inaccessible for unauthorized users. On the application level, various security levels could be assigned for different users depending on the rights they have to use specific information or perform specific activities. Some data accessible for one user might not be for another.
- *Forms Engine*: Responsible for the structure presentation of the data with the creation of on-screen data forms for entry, update and so on.
- *Workflow Services*: Responsible for the automatic routing of documents regarding particular services assigned to a user. Workflow combines rules, which govern the tasks performed by the user, and coordinates the transfer of the information required to support these tasks.

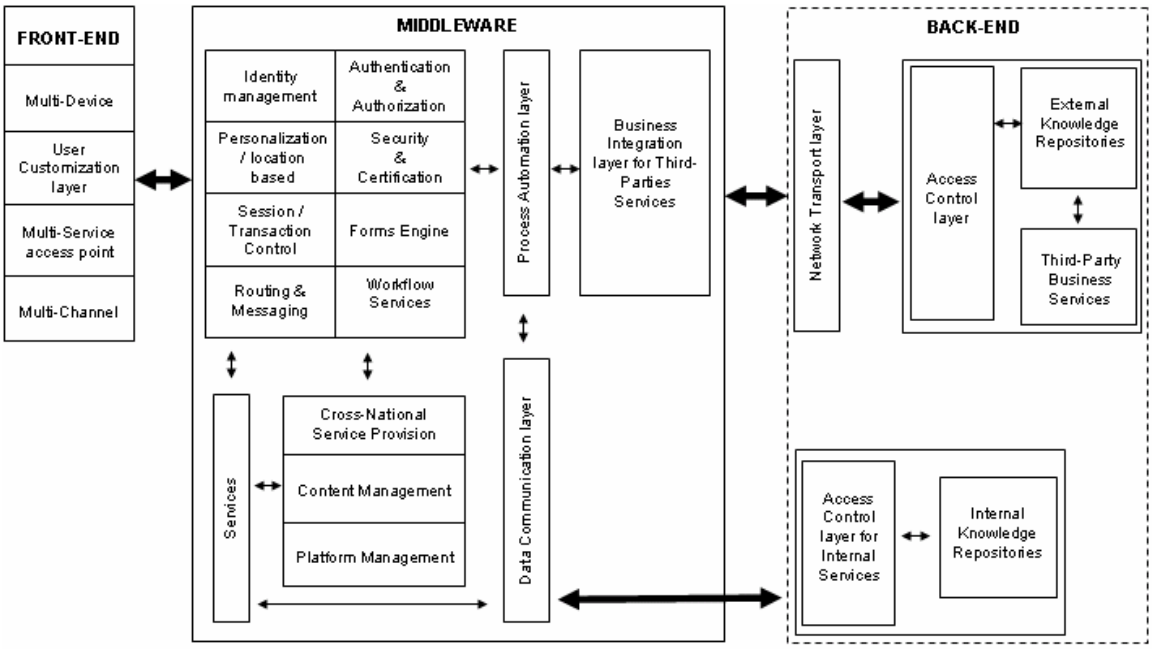

Fig. 1. Multi-channel Open Interoperable Infrastructure

- *Services*: The second component in the infrastructure is the Services component. It is responsible for providing information with regards to the internal services of the infrastructure and refers to the specific domains of e-Administration, e-Business and e-Citizen. It can communicate with the Back-end tier of the system, through the Process Automation layer and the Integration layer components, with the Third-Parties Services to extract required data. The Services component communicates directly with the first infrastructure component described above.
- The third component communicates with the Services component and the elements of the first infrastructure component. It is made up of the following elements:
  - *Cross-National Service Provision*: This element is responsible for increasing the feasibility and efficiency of cross-national services. It may be based on a multimedia distribution platform, that makes possible the replication of data between several Application Service Providers (ASP) in a network (WAN, using i.e. frame-relay or satellite technologies).
  - *Content Management*: When a user enters the system it manages / allocates the corresponding information based on their personalization profile.
  - *Platform Management*: It manages the current status of the platform controlling i.e. the number of transactions to the server, activities / tasks priority, performance issues, usability, communications, backup services, users monitoring, etc.
- *Process Automation layer*: It is responsible to provide automation of workflows, managing the sequence of activities and invoking resources.
- *Data Communication layer*: Responsible for the smooth data communication within the infrastructure. This component is of great importance since it enables the integration of data using mostly XML. The Data Communication layer component and the Process Automation layer component make up the middle-layer between the Middleware and the Back-end tiers.
- *Business Integration layer for Third-Parties Services*: This component refers to Third-Parties business services (services that are external to the infrastructure either they are new or they are legacy systems). It is responsible for the integration of the business processes, the identification and the apt communication of those services with the Middleware applying specific XML schemas, metadata, ODBCs, etc. It communicates with the Process Automation layer component and the Network Transport layer component of the Back-end tier.

4.3 Back-End Tier

This is the last tier of this architecture design and it contains transition and integration mechanisms with the external business parties as well as the databases of all the services' systems (external and internal). More particularly it consists of:

- *Network Transport layer*: It is responsible for the smooth transition of the external networks to the current infrastructure. It enables the communication of two or more inhomogeneous architectures. It is found between the Business Integration layer for Third-Parties Services component and the Access Control layer component of the Third-Party Services.
- *Third-Parties*: This component represents the external business parties to be attached on the infrastructure. It communicates with the Network Transport layer component and consists of three elements:
  - *Access Control layer*: It gives access to the identified service onto the external network and furthermore to the various business systems.
  - *External Knowledge Repositories*: These are the databases containing the data of the external business systems.

- *Third-Party Business Systems*: These are the actual external business systems. Such systems might be Customer Relationship Management system, Funds system, Business Budget system, Employee Development system, Managing Resources system, Business Accounting system, Business Inventory Control system and so forth.

- The last component refers to the internal services, communicating with the Data Communication layer component and it consists of the following two elements:
  - *Access Control layer for Internal Services*: It gives access to the internal services. At this point no intermediate component is needed, like the Third-Parties component does, since these services are included into the infrastructure itself and no further alternations or interoperability procedures are needed.
  - *Internal Knowledge Repositories*: These are the databases of the internal services systems containing all the necessary data retrieved upon request.

The proposed Multi-channel Open Interoperable Infrastructure described above will allow the Government and all related actors to provide online information and services through a single point of access to all the citizens and businesses increasing the effectiveness and quality while at the same time cutting the cost and saving time.

5 Conclusion

In this paper has been presented an eGovernment open interoperable infrastructure based on the Cross-National G2G, G2B, and G2C intercommunication needs and requirements. Related pitfalls and architecture technological implications have been mentioned as regards. The key role of the infrastructure has been exploited and reported as to provide a common technological platform whereby all the internal governmental or external business systems will run smoothly in an integrated and interoperable manner. Transparency and personalization issues will be addressed, increasing not only the effectiveness and efficiency of the systems but also participation and interaction of large number of users since there are provided citizen-centric solutions with extended customization options. Legacy stand-alone systems will be reshaped and further reengineered regarding their front-end mechanisms and processes execution leading to a more reliable, secure and optimized performance while at the same time complying with the legal framework, rules and regulations set by the European Union regarding intercommunication and data protection.

Acknowledgements

A major part of the work reported in this paper is performed in the context of the INTELCITIES (Intelligent Cities, IST-2002-507860) project, which officially started in January 2004. The INTELCITIES consortium is led by the Manchester City Council and the City of Sienna, and includes, among others, partners such as the Computer Science Department of the University of Cyprus, NetU Consultants Ltd., NOKIA Corporation, Deloitte., CISCO Systems, FZK-ITAS, TeDIS-VIU, GoPro, SYSTEMA Technologies S.A, Forseback IT and Euro Intelligence, public service providers and professional chambers.

References

1. Communication from the Commission (2003) The role of eGovernment for Europe's future, COM(2003).
2. Communication from the Commission (2004) Mobile Broadband Services, COM(2004).
3. Communication from the Commission (2004) Mobile Services in eGovernment.
4. Interchange of Data between Administrations (2004) Multi-channel delivery of eGovernment services.

5. European Commission (2004) Multi-Channel Delivery of eGovernment Services, June 2004.
6. K. Herbert , and M. Hagen One-Stop-Government in Europe: An Overview.
7. D. Gouscos, G. Laskaridis, D. Lioulias, G. Mentzas, and P. Georgiadis (2002) An Approach to Offering One-Stop e-Government Services – Available Technologies and Architectural Issues, EGOV2002, LNCS 2456, pp. 264-271.
8. M. Finger, and G. Pecoud (2003) From e-Government to e-Governance? Towards a model of e-Governance, Electronic Journal of e-Government, Vol. 1, Issue 1 (2003), 1-10.
9. J. Millard, I., and S. Jonas (2004) Reorganization of Government Back Offices for Better Electronic Public Services – European Good Practices, Final report to the European Commission.
10. D. Swedberg, and J. Douglas (2003) Transformation by Design: An Innovation Approach to Implementation of e-Government, Electronic Journal of e-Government, Vol. 1, Issue 1 (2003), 51-56.
11. M. Greunz, B. Schopp, and J. Haes (2001) Integrating e-Government Infrastructure through Secure XML Document Containers, Proceeding of the 34th Hawaii International Conference on System Sciences, 2001 IEEE.
12. Box, D., D. Ehnebuske, G. Kakivaya, A. Layman, N. Mendelsohn, H.F. Nielsen, S. Thatte, and D. Winer (2000) Simple Object Access Protocol (SOAP) 1.1 Official Specification, W3C, May 8, 2000, [online], http://www.w3.org/TR/SOAP.
13. uddi.org (2000) UDDI Technical White Paper, September 6, 2000.
14. Microsoft .NET Framework Developer Center, [online], http://msdn.microsoft.com/ netframework/
15. An overview of the J2EE Connector Architecture, [online], http://java.sun.com/j2ee/connector/overview.html.
16. C. Panayiotou, and G. Samaras mPERSONA (2004) Personalized Portals for the Wireless User: An Agent Approach", Journal of ACM / Baltzer Mobile Networking and Applications (MONET), special issue on "Mobile and Pervasive Commerce".