

# APPLYING SOA TO MOBILE SECURE EGOVERNMENT SERVICES – THE SWEB APPROACH

Silke Cuno

Yuri Glickman

Petra Hoepner

Linda Strick

Fraunhofer FOKUS,  
Berlin, Germany

{Silke.Cuno|Petra.Hoepner|Yuri.Glickman|Linda.Strick@fokus.fraunhofer.de}

## Abstract

*SWEB [1] is an EU international cooperation project that aims to develop a secure, interoperable, open and affordable government platform upon which secure cross-border mobile government services can be built. To ensure interoperability the SWEB platform is designed according to SOA concepts and implemented with Web services.*

## INTRODUCTION

Among the many promises of the digital revolution [2] is its potential to strengthen democracy and make governments more responsive to the needs of their citizens.

E-Government is defined by the World Bank Group [3]: “E-Government refers to the use by government agencies of information technologies (such as Wide Area Networks, the Internet, and mobile computing) that have the ability to transform relations with citizens, businesses, and other arms of government. These technologies can serve a variety of different ends: better delivery of government services to citizens, improved interactions with business and industry, citizen empowerment through access to information, or more efficient government management. The resulting benefits can be less corruption, increased transparency, greater convenience, revenue growth, and/or cost reductions.”

One of the main challenges with a great potential is the following goal [4]:

“E-Government can help to build trust between governments and citizens”.

To build trust, securing the eGovernment systems, services and service users is a necessary task.

Governments around the world work on the integration of the existing systems to provide them as eGovernment solutions. Most of these solutions have been started as functional approach that led to silo applications and generated islands of eGovernment solutions. Sooner or later these islands will be forced to interoperate on local, state, national or even pan-continental level. Interoperability will therefore be the main challenge to enable “cooperative government” for efficient information exchange over heterogeneous technology and organizational domain boundaries.

Interoperability is defined in the European Interoperability Framework (EIF) [5]:

“Interoperability means the ability of information and communication technology (ICT) systems and of the busi-

ness processes they support to exchange data and to enable the sharing of information and knowledge. An interoperability framework can be defined as a set of standards and guidelines that describes the way in which organizations have agreed, or should agree, to interact with each other.”

The second version of the European Interoperability Framework (EIF2.0) is expected in 2008 and will take into account the national interoperability frameworks and related activities that today either already exist in the Member States or are being prepared.

In a Gartner report [6] for the European Commission the main issues and barriers to interoperability were identified, as well as pan-European public services that need to be resolved in the near future by the EU Member States and industries. However, to enable cross-border public services usage, national governments shall provide their applications as basic public services, and therefore need to reflect all legal, organizational, process, semantic and technical measures necessary to do so. The figure below depicts the Generic Public Services Framework.

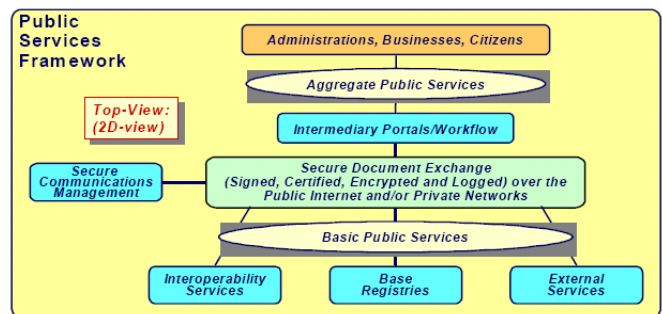


Figure 1: The Generic Public Services Framework  
(Source: Gartner Report [6])

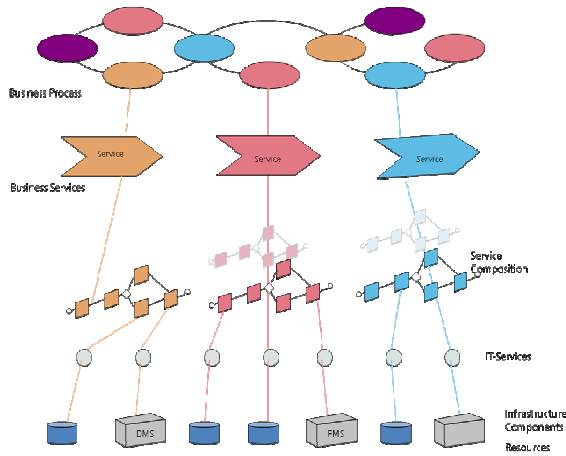
To support these goals EU Member States have to define an eGovernment architecture and infrastructure, which facilitates the integration and use of new and existing applications in a service-oriented way.

## SOA AND WEB SERVICES

The World Wide Web Consortium (W3C) refers to SOA as “A set of components which can be invoked, and whose interface descriptions can be published and discovered”. [7] The Service-Oriented Architecture (SOA) approach emerged in response to existing integration problems, re-

lated to too many protocols, too many representations and too many adapters.

SOA is based on a service concept and includes not only the IT-perspective but also the business perspective. Services are designed and deployed to match the customers/citizens needs. Complex applications can be composed from services and exposed as services to humans and IT systems. SOA comprises the style of design that guides all aspects of the service life-cycle from the conception over the implementation to the deployment and maintenance, and reflects also mechanisms to define and to provide data exchange and business processes for the IT infrastructure. It provides a higher level of abstraction at which functionality can be specified, published and/or consumed compared to objects or components.



**Figure 2: SOA Architectural view**

In essence, SOA is an architectural approach that aligns business processes with business services, provides mechanism for service composition, allows for integration of legacy systems and integrates the infrastructure as shown in Figure 2. Both, processes and services need to be carefully coordinated to assure an effective SOA implementation.

While previous approaches had its focus more on the use of a specific execution environment technology, SOA focuses on the description of the business problem. The notion of service is not constrained by a specific type of technical implementation. It is transparent for a user of a service how the service is implemented, in other words, it does not matter if it is implemented using Java, Corba, .NET or a legacy language. The SOA communication infrastructure provides an easy way to use a service independent of its location. In a service-oriented architecture the infrastructure is called Enterprise Service Bus (ESB).

SOA's loose coupling principle is to avoid or at least encapsulate temporal, technology and organizational constraints, thus services can be easily orchestrated.

Loose coupling is a broad term that actually refers to several different elements of a service. Services can be used sole-

ly based on the published service contract and service-level agreement. The notion of loose coupling also avoids dependency so that, if a change is made to a service, the other services, which interact with this service, need not to be changed. The service itself publishes its interfaces in implementation-neutral formats and protocols, thus e.g. permitting location transparency.

The development of services shall not depend on a particular technology, product, or development platform. However, a commonly used technology for implementing services is Web services. The identification of a service is not an easy task, but it should not be tied to a single business process, so that it can be reused in many different processes and applications. This kind of reasoning is not new and represents solely the generalization of similar concepts, well known at the level of objects and components, now to the level of more complex autonomous applications, the services.

SOA provides flexibility that is a fundamental need in the e-government integration context due to the numerous applications. Usually these applications are distributed over many governments agencies, implemented on different operating systems and by different programming languages, and with distinct data models.

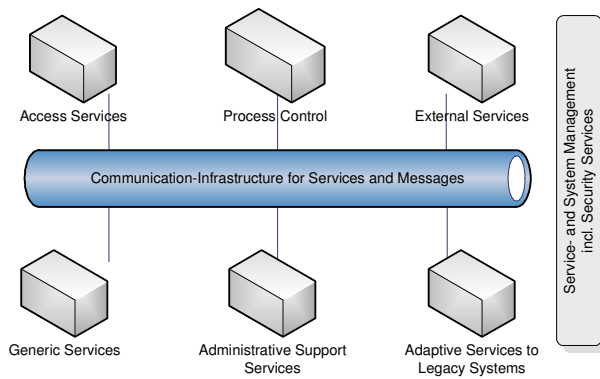
In addition to this flexible integration capability, SOA provides a universal mechanism to access and compose the services into a value-adding business solution. The process-oriented view in the context of eGovernment needs collaboration of services in order to support administrative business process chains.

Another issue addressed by SOA is the facility to reuse; once a service has been developed it can easily be reused in other business processes. Services have standard interfaces facilitating the access and the composition of these services in new business processes. Services can also be defined at different granularity levels, for example: a fine grained component service can be used for composition and described by workflow mechanism. A coarse grained service, such as a business service, can consist of the composition of these fine grained service components, which can also exist in external systems.

To perform administrative applications a service-oriented software infrastructure provides the runtime environment and manages and integrates these applications and the resources needed. The following types of services construct this infrastructure [8] as depicted in Figure. 3:

Some examples of these service types are listed below:

- Access Services: access for citizen, business and administration providing various technologies (portals, mobile access);
- Process control: coordination and automation of administrative process chains;



**Figure 3: SOA for eGovernment**

- External services: services from external providers (e.g. ePayment, certification authority services);
- Generic services: registries for people and services;
- Administrative Support Services: administrative services used by more than one eGovernment application (e.g. virtual post office);
- Adaptive Services to Legacy System: integration and adaptation services that hide the legacy system and expose it as a service;
- Service- and System Management including Security Services: management and control of the service-oriented software infrastructure.

### EGOVERNMENT INTEROPERABILITY PROFILE

Governments, in general, are investing in nation-wide projects aiming to create eGovernment interoperability patterns. Most of them are encouraging the adoption of open standards such as: XML, SOAP, HTTP and SMTP. In many cases they give a special attention to Web services patterns; Web services are mentioned in documents of German SAGA [9], US Federal Enterprise Architecture [11], and EIF [5] to name some. It is widely accepted that Web services emerge as the most popular technology to implement the abstract concept of SOA, which supports eGovernment integration by platform neutral interfaces, standard based connectivity and loose-coupled services. For this reason Web service interoperability has to be technically supported by specific open standards.

The Web Service Interoperability Organization (WS-I) defines a profile (WS-I Basic Profile 1.2 [12]) of existing standards to provide implementation guidelines for how related Web services specifications should be used together for best interoperability. The profile incorporates XML Schema v1.0, SOAP v1.1, WSDL v1.1, and UDDI v2 and X.509 and related bindings and defines extensibility points within them.

The WS-I Basic Security Profile (BSP) [13] is an extension profile to the WS-I Basic Profile. This means it is consistent with the Basic Profile but defines additional functionality – e.g. to add conformant security features to the Basic Profile when needed. As an extension of the Basic Profile, the Basic Security Profile is designed to support the addition of security functionality to SOAP messaging, in an interoperable manner.

The WS-I Basic Security Profile 1.0 provides guidance on the use of WS-Security; it addresses Transport Layer Security, SOAP Message Security, Username Token Profile, Kerberos-, Rights Expression Language (REL) and SAML Tokens, Timestamps, X.509 Certificate Token Profile, XML-Signature, XML Encryption, Algorithms, Relationship of Basic Security Extension Profile to Basic Profile, and Attachment Security.

BPEL (Business Process Execution Language) defines an interoperable integration model between Web Services. It is an open XML-based language for the formal specification of executable business processes and abstract business processes defining business interaction protocols. It describes executable workflow orchestrations, which are designed to describe modern business processes in a smart manner. One of the key objectives while inventing BPEL was to standardize the definition format of process flows to actually make interaction with other business environments seamlessly and easy. The standardization was performed by OASIS and is named Web Services Business Process Execution Language (WSBPEL) v2.0 [14]. Most development environments provide a proprietary graphical notation and corresponding wizards to specify BPEL programs thus the BPEL programmer will not recognize the XML basics of the language.

### SWEB ARCHITECTURE

SWEB (“Secure, interoperable cross border m-services contributing towards a trustful European cooperation with the non-EU member Western Balkan countries”) is an EU international cooperation project that aims to develop a secure, interoperable, open and affordable government platform upon which secure cross-border mobile government services can be built. These services address governmental organizations, citizens and companies and will facilitate the creation of a more citizen-centred form of government.

The impressive penetration rates of mobile networks in Western Balkans give the unique opportunity to the countries in the region to use mobile services and accelerate their entrance in the digital society. This gives the advantage to the regional public administrations to skip a step and enter directly the mobile Government.

SWEB reuses some of the results of the eMayor project (www.emayor.org), a successful eGovernment security project in which five Municipalities - Siena, Psychico in Athens, Aachen, Bolzano, Seville - decided and tested two secure cross border e-government services. The eMayor platform is built from EJB components implementing basic services. Some of them, like components responsible for

notification of users, task management, Web GUI, etc. are reused for the SWEB platform after required updates. However, many components are implemented from scratch in order to realise innovative way of mobile access to the SWEB platform.

Moreover, there are architectural changes/updates to the eMayor platform that are necessary in order for the SWEB platform to provide reliable and secure mobile access to its users and efficient services composition. These updates concentrate on the architectural structure based on SOA, having a specific focus on technical and semantic interoperability as well as on mobile access and security services for public administrations. Technical interoperability in SWEB comprises service integration and interoperability aspects, i.e. the support of services working across multiple devices (mobile and fixed) platforms and computing networks, the usage of specialized adapters to integrate existing services with proprietary interfaces and newly developed services with different eGovernment platforms using standard-based open interfaces.

A main concern of different governments is the semantic interoperability, i.e. data models and interchange formats between administrations. SWEB will not define the generic data models, as this is regarded as an activity to be performed on national and pan-European level. For example an online ‘Semantic Interoperability Centre’ was established as part of the IDABC website, enabling publication of news and updates, providing links and information to current initiatives and serving as an XML information hub [15].

The already carried out analysis in eMayor made it clear that, in spite of the great diversity of public services, they have very similar process structures. One of the reasons for this is that the long term development of bureaucracy may lead to a sort of tradition, transferred from organization to organization. Anyway, public services are somehow quite similar everywhere. At first, they provide information for the citizen of any kind of administration issues. Then there are the requests for applications, for example when opening a new business or applying for a new passport. For this kind of applications the processes are quite similar, request order, including, if necessary, identification check, perform order, get payment and deliver application form. The order in the process might vary, but not significant. The difference is lying partly in its grade of process automation. Some of them may not have all activities automated; some need specific adaptation because they have specific requirements.

Based on these findings SWEB concentrates on the Security Services (as introduced in Figure 3) to support mobile and secure eGovernment applications. The architecture proposed is depicted in Figure 4.

According to the generic SOA for eGovernment in Figure 3, SWEB comprises various services with the focus on mobile requirements:

- Security and Access Services: Mobile and secure browser based access services for citizen, business and administration are the main access channels.
- Process control: Coordination of administrative process chains is provided by administration and orchestration services, which enforce a role based access to resources and coordinate the interaction between the various services.
- External services: External services are the PKI and Time stamping authorities required for secure authentication with public key certificates, electronic signatures and non-repudiation.
- Generic services: Services will register in a UDDI regi-

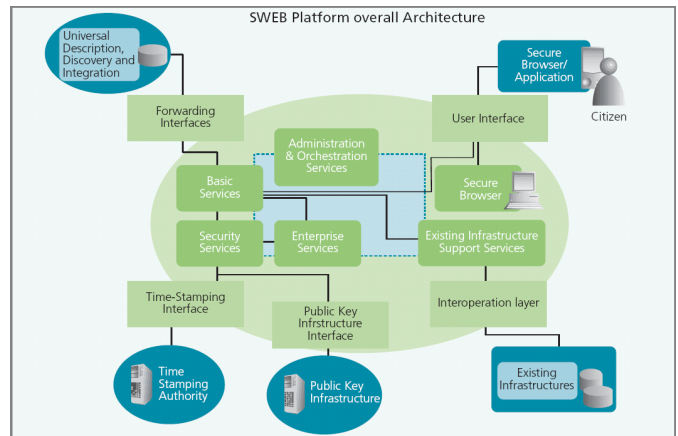


Figure 4: SWEB Architecture

stry.

- Adaptive Services to Legacy System: Integration and adaptation services are handled by the Interoperation layer to connect to the existing infrastructures of the participating municipalities.

In the currently executed implementation phase of SWEB the mentioned services are implemented in detail.

### FUTURE ACTIVITIES

SWEB currently implements the Web services that will realize a SOA for governmental applications supporting mobile access and taking into account the specific security requirements. The Web services will be tested in pilot environments with two services:

- Residence Certification Service: as a specific example for a secure municipal document exchange service, in which a public organization and individual citizens can securely communicate e/m-municipal documents.
- Electronic/Mobile Invoicing, which has a critical role in all the stages of handling Value Added Tax (VAT) procedures for EU Member States. Through e/m-invoicing, tax administrators will be able to implement

new tools and procedures to carry out alternative controls.

Before the tests of the platform in the pilot environment will start a special training for civil servants of involved municipalities will be organized. This training will not only teach users to use the platform but also to get first feedback about the platform from them. The feedback will be used on the next iteration of the implementation to refine the platform and eliminate defects.

## ACKNOWLEDGMENTS

This paper is based on the work undertaken and in progress in the context of the project SWEB (044979), which is partially funded by the European Commission in the European Union's "IST" program. The authors would like to acknowledge all SWEB partners.

## DISCLAIMER

This document has been produced with the financial assistance of the European Community. The views expressed herein are those of SWEB Consortium and can therefore in no way be taken to reflect the official opinion of the European Commission. The information in this document is provided as is and no guarantee or warranty is given to state that the information is fit for any particular purpose. The user therefore uses the information at their sole risk and liability.

## REFERENCES

- [1] SWEB Project Homepage, at <http://www.sweb-project.org/>
- [2] eGovernment Handbook, Center for Democracy and technology, at <http://www.cdt.org/egov/handbook/>.
- [3] World Bank Group: A Definition of e-Government, at <http://www1.worldbank.org/publicsector/egov/definition.htm>.
- [4] OECD: Checklist for e-Government Leaders, September, 2003, at <http://www.oecd.org/dataoecd/62/58/11923037.pdf>.
- [5] IDABC, European Interoperability Framework for pan-European eGovernment Services (EIF), Version 1.0, November 2004, at <http://europa.eu.int/idabc/en/document/3761>.
- [6] Gartner, Preparation for Update European Interoperability Framework 2.0, Final Report for IDABC, 4 June 2007, at <http://ec.europa.eu/idabc/servlets/Doc?id=29727>
- [7] W3C, "Web Services Glossary", 11 February 2004, at <http://www.w3.org/TR/ws-gloss/>.
- [8] R. Reinecke, „Software-Infrastruktur für eGovernment, Einsatz von Anwendungsfällen“, eGovernment Computing, 11/2006 (in German).
- [9] KBSt unit, Federal Ministry of the Interior of Germany: SAGA - Standards and Architectures for e-Government Applications, Version 3.0, October 2006, at [http://www.kbst.bund.de/cln\\_047/nn\\_837392/SharedDocs/Anlagen-kbst/Saga/saga\\_\\_3\\_\\_0\\_\\_en.html](http://www.kbst.bund.de/cln_047/nn_837392/SharedDocs/Anlagen-kbst/Saga/saga__3__0__en.html).
- [10] United Kingdom, e-GIF 6.1, 18 March 2005, at <http://www.govtalk.gov.uk/schemasstandards/egif.asp>
- [11] USA, Federal Enterprise Architecture (FEA), at <http://www.whitehouse.gov/omb/egov/a-1-fea.html>.
- [12] WS-I, Basic Profile Version 1.2, Working Group Approval Draft, 2007-10-24, at [http://www.ws-i.org/Profiles/BasicProfile-1\\_2\(WGAD\).html](http://www.ws-i.org/Profiles/BasicProfile-1_2(WGAD).html)
- [13] WS-I, Basic Security Profile Version 1.0, Final Material, 30 March 2007, at <http://www.ws-i.org/Profiles/BasicSecurityProfile-1.0.html>.
- [14] OASIS, Web Services Business Process Execution Language Version 2.0, OASIS Standard, 11 April 2007, at <http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.pdf>.
- [15] IDABC, Speaking the same language: semantic interoperability for European services, April 2005, at <http://europa.eu.int/idabc/en/document/4272> and IDABC, IDABC Semantic Interoperability Strategy: The European XML Clearinghouse, Feasibility Study, December 2005, at <http://ec.europa.eu/idabc/servlets/Doc?id=24406>. and Semantic Interoperability Center at: <http://www.semic.eu/>