E-Government in Germany:

Status Quo and Perspectives

Alexander ELSAS
Institute for Information Systems, Johann Wolfgang Goethe-University
Frankfurt, 60054, Germany

Abstract

According to an UN survey, only 17 countries have reached a transactional stage of E-Government, no country has reached the fully integrated or seamless stage. A technological way to support reaching the seamless stage can be the utilization of the Web services framework to implement advanced, integrated E-Government applications. The actual situation in Germany is taken here as an example to illustrate the typical problems of recent E-Government developments.

Keywords: Web Services, SOAP, WSDL, UDDI, Service-orientation, E-Government

1. Introduction

The overwhelming success of the Internet technology as the foundation for E-Commerce and E-Business applications has caused considerable pressure on the public sector to modernize administrative processes and implement interaction with the citizens via the World Wide Web.

The so-called E-Government applications range from information-offering static web sites to sophisticated transaction-oriented applications, supporting the administrative processes and interaction with the citizens.

A first look at the dimensions of E-Government can be achieved by categorizing the involved partners:

- **Government-to-government (G2G) or administration-to-administration (A2A)**: internal processes of the administration.
- **Government-to-business (G2B) or administration-to-business (A2B)**: interactions between governmental organizations and commercial or business organisations, e.g. taxes.
- **Government-to-citizen (G2C) or administration-to-citizen (A2C)**: interactions between governmental organisations and citizens, e.g. marriage.

The degree of sophistication of the E-Government applications can be categorized in four levels:

1. Static web sites providing information.
2. Web sites with interaction.
3. Support of internal and external workflows of the administration.
4. Online voting.

According to [12] another classification of the state of development of E-Government is:

1. Emerging: A government web presence is established through a few independent official sites. Information is limited, basic and static.
2. Enhanced: Content and information is updated with greater regularity.
3. Interactive: Users can download forms, contact officials, and make appointments and requests.
4. Transactional: Users can actually pay for services or conduct financial transactions online.
5. Seamless: Total integration of e-functions and services across administrative and departmental boundaries.

Most countries are engaged in various E-Government projects on different levels of administration and sophistication. The reasons for

the implementation of E-Government applications are based on efficiency and cost developments. Most projects make use of proprietary, often self-developed, standards for communication and interaction. The problem is a lack of coordination and joint efforts between the different projects, which might lead to incompatibilities and integration problems, e.g. with so-called *one-stop-government* portals. This is a single website that acts as a citizen portal, giving access to information originating from different sources or allowing transactions with different local, regional, federal, or even foreign authorities.

Germany with its political system of federalism and local autonomy is taken as an example, because this political structure is a challenge for the implementation of an integrated, legislative boundaries bridging E-Government.

2. Status Quo

The *United Nations Online Network in Public Administration and Finance* (UNPAN) survey in 2001 ranks Germany as # 10 in the world. Germany is classified on the *transactional* level, together with 16 other nations, a *seamless* level has not been reached by any country.[12]

Rank	Country	Index
1.	USA	3.11
2.	Australia	2.60
3.	New Zealand	2.59
4.	Singapore	2.58
5.	Norway	2.55
6.	Canada	2.52
7.	United Kingdom	2.52
8.	The Netherlands	2.51
9.	Denmark	2.47
10.	Germany	2.46

Table 1: Global E-Government leaders

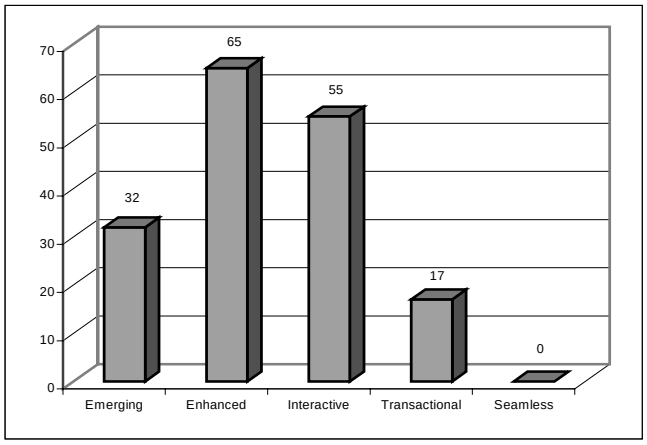

Figure 1: Country stages in 2001

Web sites

Nearly all German municipalities, counties, states, or federal authorities have their respective web sites.

The stage of development of the various sites ranges from *enhanced* to *transactional*, with the majority in the stage of *interactive*.

Table 2 gives a short overview with selected best-practice examples.

Authority	Stage	URL
Town / State of Hamburg	Transactional	www.hamburg.de
Town / State of Bremen	Transactional	www.bremen.de
County of Ludwigslust	Transactional	www.kreis-lwl.de

Table 2: Best practice examples in Germany

Unfortunately there is no overall accepted standard for online identification: different citizen smart cards on the one hand and simple user / password identification on the other hand represent the range of technologies. This focuses the problem on compatibility and interaction issues between different authorities: The citizen smart card of Bremen won't work in Hamburg and vice versa.

Electronic Tax Declaration

The electronic tax declaration system *ELSTER* ("magpie", www.elster.de) is the success story within the German E-Government projects. More than 1,500,000 income tax declarations and 16,000,000 tax registrations have been processed since January 2000.

ELSTER implements a proprietary communication protocol, that can be integrated into the traditional tax declaration programs as a way of communication with the tax office.

ELSTERformular is a free forms software, offered by the tax offices. It allows the online tax declaration via the Internet, it uses the ELSTER protocol. The software is available via Internet download.

Voting

The first regular online elections via Internet have been for the student council at Osnabrück University in February 2000. The so-called *i-vote* project (www.internetwahlen.de) has gained a lot of experience with other elections on different levels of legislation, since then.

The results are mixed: the idea of nation-wide elections for the federal parliament via Internet, which was a vision at the beginning of the project, has been postponed.

Open Source Policy

The recent decision of the city council of Munich to replace the Microsoft office applications and server infrastructure with Open Source alternatives like OpenOffice or LINUX has gained a lot attention worldwide.[6]

In contrast, the "banking capital" Frankfurt/Main has decided to keep the Microsoft applications and network infrastructure. It is evident, that there is no overall *Open Source Policy* within Germany.

Conclusion

A lot of interesting or innovative projects in all fields of E-Government applications have been developed in Germany over the past few years. The UNPAN survey ranks Germany in the Top Ten of worldwide E-Government usage, it is one of the only 17 countries which have reached the *transactional* stage.

The main deficiency of the German E-Government activities is the lack of coordination and compatibility. The existing applications use proprietary protocols, it is not possible to implement interaction and communication between different applications without reengineering them.

This is a major obstacle on the way to reach the *seamless* state of E-Government, which is the next step in development.

3. Perspectives

As a technological way to support the *seamless* state of E-Government, the *Web services* framework is presented here. It is supported by the global IT players like IBM, Microsoft, or Sun in their respective products.

In contrast to traditional IT systems, which can be characterized as tightly coupled systems, Web services implement a loosely coupled approach.[8]

A Web services framework is consisting of three basic services:

- Communication,
- Service description and
- Service discovery.

These basic functionalities are implemented by protocols, the three basic standards within the Web services framework for these protocols are SOAP, WSDL and UDDI, all based on the common *Extensible Markup Language* (XML).[4]

Figure 2 shows the difference between the well known *Hypertext Markup Language* (which is the foundation of all web sites) and XML: HTML defines the presentation of the words *Alexander* and *Elsas* in a bold typeface, XML defines *Alexander Elsas* is a name.

HTML:	Alexander Elsas
XML:	<name> Alexander Elsas </name>

Figure 2: HTML vs. XML.

Figure 3 gives a graphical overview over the interaction concept of the Web service protocols, in the notation of an *UML component diagram*.

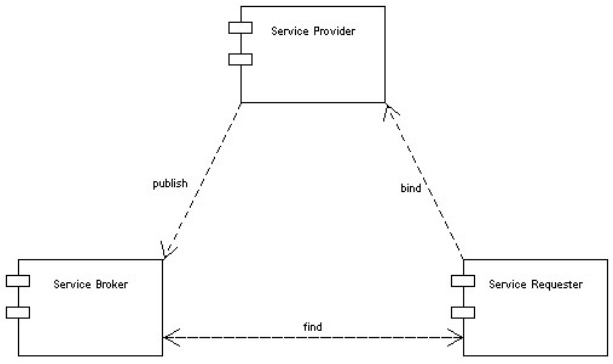

Figure 3: Service orientation

This architecture is often referred to as a *Service Oriented Architecture* (SOA), in the case of Web services, this SOA-approach is implemented with the before mentioned protocols:

- A specific Web service announces its WSDL definition to a UDDI registry (*publish*).
- A client searches the UDDI registry for a service's definition (*find*).
- The client sends messages or requests directly to the service via SOAP, based on the information of the WSDL definition from the UDDI directory (*bind*).

SOAP

The *Simple Object Access Protocol* (SOAP) [9], a joint development of Microsoft and IBM and a few other companies, whose further development is in the hands of the W3C,[2] is a XML based protocol for messaging and remote procedure calls (RPCs, the execution of programs or program fragments on remote computers). SOAP defines how distributed applications can communicate in a message-oriented way.[4]

Figure 4 shows a simplified SOAP message, defining an electronic ticket for air travel; Figure 5 depicts the message-oriented communication with SOAP.[10]

```
POST /travelservice
SOAPAction: "http://www.cybertravel.com/checkin"
Content-Type: text/xml: charset="utf-8"
Content-Length: nnnn

<SOAP: Envelope xmlns:SOAP="http://schemas.xmlsoap.org/soap/envelope/">
    <SOAP:Body>
        <et:eTicket xmlns:et="http://www.cybertravel.com/eticket/schema">
        <et:passengerName first="Alexander" last="Elsas"/>
        <et:flightInfo
                        segment="FRA-MCO"
                        airline="UA"
                        class="C"
                        flight="945"
                        departureDate="2003-07-27"
                        departureTime="0830"
                        arrivalDate="2003-07-27"
                        arrivalTime="1709" />
        </et:eTicket>
    </SOAP:Body>
</SOAP:Envelope>
```

Figure 4: SOAP message for an electronic ticket

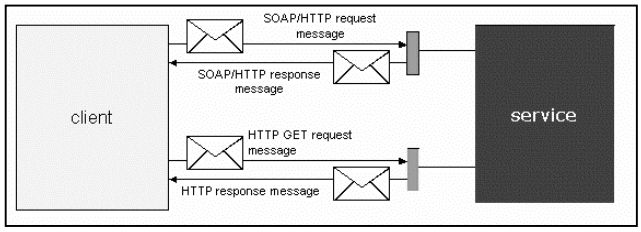

Figure 5: Message-oriented communication with SOAP

WSDL

The *Web Services Description Language* (WSDL) describes Web services as a collection of end-points for communication that can exchange messages. A complete service description consists of two parts of information:[6, 13]

- The *abstract interface* as a service description on the application-level.
- The *protocol-specific details* that have to be followed to gain access to services at the end points.

UDDI

The *Universal Description, Discovery, and Integration* (UDDI) can be seen as an online, automated "phone" directory of Web services. An UDDI registry holds three types of information about Web services:[4]

- Names and contact details (*white pages*),
- A categorization of business and service types (*yellow pages*),
- Technical details (*green pages*).

Additional Web Service Protocols

The protocols, that have been described so far, build the core protocols of the Web service technology stack. On top of these, industry-wide accepted core protocols, further specialized protocols are in different stages of development. These protocols address issues like workflow modeling within the Web services framework (e.g. *Business Process Execution Language for Web Services*, BPEL4WS) or integration into portals (e.g. *Web Services for Remote Portals*, WSRP). As the state of development of these protocols is not as stable and mature as the state of the core protocols,[11] they are omitted here. More details about these concepts can be found in [7] (BPEL4WS) and [14] (WSRP).

4. Summary and Outlook

The forthcoming E-Government applications should be based on common standards to allow for easier integration with other applications. The number of existing E-Government applications is still small, therefore upgrading them to a Web service based framework is not a problematic issue at the moment. The experiences gained with the existing applications should go directly into the next generation, which should be based on Web service standards.

This conclusion can also be drawn from recent research activities:

- The German Bertelsmann Foundation has formulated a recommendation for action to establish successful E-Government projects which is based on a recent best-practice study:[1, 5] An important point within the recommendations is the use of established standards.
- The SAGA project in Germany comes to the conclusion that the Web service concept and its related technologies, standards and products are an appropriate basis for the integration of the different administrative services.[3]

The first challenge for the forthcoming E-Government applications is to learn from the E-Commerce experiences of the last five years and to avoid the apparent shortcomings. Founding the applications on a Web services framework promises to be a feasible solution.

If a really *seamless* state of E-Government shall be reached, there is another challenge to be taken into account. This challenge is the issue of integration between E-Government and E-Commerce applications.

With the introduction of seamless E-Government the need for collaboration with the E-Commerce world arises. An example can be found in the E-Government process of registering a car in Germany: From a citizens point of view it would be most convenient to integrate the licence plate producer into the process *Car Registration*.

The need for communication and thus integration between E-Commerce and E-Government applications is evident; Figure 6 shows this scenario.

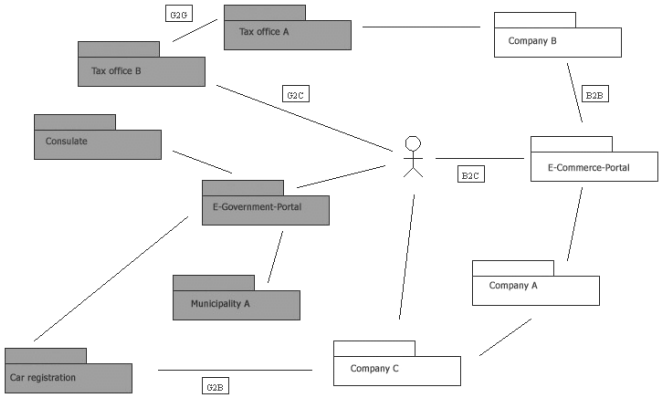

Figure 6: Integration with E-Commerce applications

The Web services framework, as it is independent from the application context (E-Government or E-Commerce), allows for the easy integration of these two worlds.

The support of all major IT companies for the Web services framework is another point for the utilization of the concept.

References

[1] Bertelsmann Foundation (ed.): Balanced E-Government; 2002; URL: http://www.begix.de/en/studie/download.html (7.9.2002)

[2] Box, Don: A Brief History of SOAP; 2001; URL: http://www.xml.com/pub/a/2001/04/04/soap.html (6.8.2002)

[3] BundOnline 2005 (ed.): One Stop Government; URL: http://www.bund.de/BundOnline-2005/SAGA/Glossar-.6343.htm?suche=O* (7.9.2002)

[4] Curbera, Francisco; Duftler, Matthew; Khalaf, Rania; Nagy, William; Mukhi, Nirmal, Weerawarana, Sanjiva: Unraveling the Web Services Web – An Introduction to SOAP, WSDL, and UDDI; in: IEEE Internet Computing, March / April 2002, p. 86 – 93

[5] Friedrichs, Stefan; Hart, Thomas; Welzel, Carolin: 10-Punkte-Plan für gutes E-Government - Ein Fahrplan zur Verwaltungsmodernisierung und Stärkung der Bürgergesellschaft; 2002, URL: http://www.begix.de/10_Punkte_Plan.pdf (7.9.2002)

[6] Galli, Peter: Why Munich Dumped Microsoft for Linux; URL: http://www.eweek.com/article2/0,3959,1110809,00.asp (4.7.2003)

[7] IBM (eds.): Business Process Execution Language for Web Services version 1.1; URL: http://www-106.ibm.com/developerworks/webservices/library/ws-bpel/ (4.7.2003)

[8] Kleijnen, Stans; Raju, Srikanth: An Open Web Services Architecture; in: acm queue, March 2003, p. 38 – 46

[9] Mitra, Nilo (Eds.): SOAP Version 1.2 Part 0: Primer; 2002, URL: http://www.w3.org/TR/soap12-part0/ (26.8.2002)

[10] Shohoud, Yasser: Introduction to WSDL; URL: http://www.learnxmlws.com/tutors/wsdl/wsdl.aspx (7.8.2002).

[11] Stal, Michael: Web Services: Beyond Component-Based Computing; in: Communications of the ACM, October 2002, p. 71 – 76

[12] UNPAN (eds.): Benchmarking E-government: A Global Perspective; URL: http://www.unpan.org/e-government/Benchmarking%20E-gov%202001.pdf (4.7.2003)

[13] Weerawarana, Sanjiva; Chinnici, Roberto; Gudgin, Martin; Moreau, Jean-Jacques (Eds.): Web Services Description Language (WSDL) Version 1.2 – W3C Working Draft 9 July 2002; URL: http://www.w3.org/TR/wsdl12 (6.8.2002)

[14] Wege, Christian: Portal Server Technology; in: IEEE Internet Computing, May / June 2002, p. 73 – 77