

Web Services and e-Local Government

Çetin CÖMERT Halil AKINCI Necati ŞAHİN
Geodesy and Photogrammetry Department
Karadeniz Technical University
61080 Trabzon / TURKEY
e-mail: ccomert@ktu.edu.tr

Abstract

"Web Services" were first realized around the year 2000. Since then, they have collected a great interest from various communities. Although it has originally been foreseen for e-business applications, Web Services are now in the mainstreams of many companies that either produce or use Information and Communication Technologies. There are already a number of software companies providing tools to develop and deploy Web Services. Although it sounds like a new technology, Web Services has its roots on distributed systems and it is an evolution of the work on that area. They both have the same philosophy behind, which is the well-known concept of "distributed but cooperating" systems. But the way that the "cooperation" takes place is different. While the stress is on the distribution of the "data" in distributed systems, it is on the distribution of the "tasks" in Web services. In the Web Services view, every different system or component offers some services for some others. Any system does its job by just calling suitable services over Internet. From this respect, one can even derive the analogy between the philosophies behind Web Services and Object-Oriented. In general, Web Services seems to have a great potential for wherever there is a need for cooperation or "interoperability". In Today's World, the interoperability, the ability of different systems working together, is the only way to assure rapid, cost-effective and quality solutions or services. This need, backed up and in a sense provoked by Internet, and the Web have opened up the road to Web Services. The need for interoperability has long been felt in spatial data management communities as well. Therefore, these communities have recently directed their work towards the adoption of Web Services. The purpose of this work was to suggest Web Services for local governments to lower costs, improve service quality, and increase revenues in local government. This is an urgent need especially for local governments in Turkey. We have first examined major local government activities for the municipality of Trabzon in Turkey. And identified possible Web Services. We have also implemented some of these services using Cape Clear™ Web Services development and deployment software. We have found out that Web Services has a great potential for e-local government. In addition to improving qualities of services, and lowering costs, Web services would highly increase incomes by both closing the gaps and enabling new services.

Keywords: *Web services, e-government, e-municipality, interoperability.*

Introduction

One of the most striking developments of the passed decade was perhaps the Internet. People use Internet for almost anything today; they read news, search for information, get information, watch stock exchange, check the movies of the week, and so on. It has changed the life on the Planet. It has changed the way of doing business for instance. Rather than the traditional eight

hours a day, companies have to now offer 7/24 services. Organizations are using the Web to deliver their goods and services, to find trading partners, and to link their existing applications to other applications.

Today, for any sector, the need for integrating applications is more pressing than ever. This is felt rather strongly in the business-to-business (B2B) sector, where every different company has applications in their particular area of specialization. Nokia (2003) highlights this need for the mobile community. The business model of today necessitates the cooperation among these applications. Consider different parties involved in a business trip reservation for instance; at the requestors side there might be desktops, mobile phones, or PDAs, while at the providers side there would be airline, car rental, and hotel reservation systems. The ultimate business goal here is to provide the requestor with a high quality service that would enable him to effect his bookings in a quick and an easy way. For this to happen, all the systems involved should be able to "talk" to each other. And the provision of this "talk" should be easy and cost-effective for businesses to be able to make money.

The ability by which the different applications can talk and cooperate with each other is known as interoperability. The differences of applications may be in either hardware or software platforms, or in data formats. The need for interoperability has also been felt strongly in spatial data communities. There have been considerable work on this and many national and international bodies have been working on spatial data interoperability. Interoperability is also a must for National Spatial Data Infrastructures (NSDI) which are now underway in many countries. However, the spatial data interoperability solutions have been either inefficient, proprietary, or complex. That is why some of NSDI undertakings such as Canadian Geospatial Data Infrastructure (CGDI) and major bodies such as Open GIS Consortium (OGC) have directed their work towards Web services (CGDI, 2001; Reed, 2002).

Web services have emerged as the next generation of Web-based technology for interoperability. Web services are modular, self-describing, self-contained applications that are accessible over the Internet. Based on open standards, Web services enable constructing Web-based applications using any platform, object model, and programming language (Barefoot, 2002). A service is a collection of operations accessible through an application-programming interface that allows users to invoke a service, which could be a response to a simple request to create a map or a complicated set of image-processing operations running on several computers (Hecht, 2002).

The need for interoperability is also strong for municipality and government applications. Due to the lack of interoperability, traditional governments and municipalities face serious problems. These are mainly poor quality and high cost of services, and low economical revenues. E-government has been proposed for solving these problems by making the interaction between government and citizens (G2C), government and business enterprises (G2B), and inter-agency relationships (G2G) more friendly, convenient, transparent, and inexpensive (World Bank, 2002). What is needed for e-government is an interoperability infrastructure (OeE, 2003). This point has also been our motivation in undertaking this work whose purpose was to determine the potentials of the Web services for an e-government interoperability infrastructure. For this aim, we have first investigated the activities of the municipality of Trabzon, Turkey. And determined the problems of the traditional system. Then we have determined, developed, and implemented a number of these services. We have concluded that Web services are very promising for the e-

municipality infrastructure. Since e-government and e-municipality share similar structures and problems, the findings of this work would also be valid within the context of e-government.

Web Services

Since it is a very new technology, there are many definitions of Web services around. According to World Wide Web Consortium (W3C) a Web service is as a software system identified by a URI, whose public interfaces and bindings are defined and described using XML (W3C, 2002). Kreger (2001) defines a Web service as an interface that describes a collection of operations that are network accessible through standardized XML messaging. Cerami (2002), defines a web service as any service that is available over the Internet, uses a standardized XML messaging system, and is not tied to any one operating system or programming language. Thomas (2003), on the other hand, defines Web services as a "Web API" that lets the applications communicate using XML and the Web.

Although differently phrased, various definitions of web services have some points in common. One of them is of web services being for the application-to-application communication. The other one is that web services are accessed over Internet. Finally, the definitions point out that web services are XML based and not for proprietary solutions.

Web services mean a shift from "human-centric" Web to a "application-centric" Web (Cerami, 2002). In its widespread use today, Web is human-centric, where the Web has been perceived as the means of merely disseminating information to as large an audience as possible. This view falls short in meeting the expectations of Today. Users of the Web do not only get information but also do business or develop applications over the Web. There are three components of "human-centric" Web; The Web, user, and the application programs. The way of achieving a goal or "doing the job" is the user-Web interaction. What usually happens is the user browses through many web pages, spending plenty of his time and hopefully gets some useful information. However this is not the end of the story, he might also need to use an application program to do the job. This means that the user might have to go through many lengthy searches and then transfer the results to the application programs. Besides being error-prone, this is an unacceptably time consuming process. In the "application-centric" Web, it is the applications that discovers and manipulates the information. In short, in "human-centric" Web it is the human who controls the processes and does the job, whereas in "application-centric" it is the applications that control and do the job by interacting with each other over the Internet and present the results to the users. Web services have emerged from the need for such applications. There are other names for the "application-centric" Web in the literature. For instance, Ryman (2000) uses the term "Service Web". Tim Berners-Lee, the original inventor of the Web, has coined the term "Semantic Web" and envisioned the Web services as actualization of the Semantic Web vision Cerami (2002). See Berners-Lee et al. (2001) for more on Semantic Web.

An example from the local government activities will be given below to highlight the need for Web services. Ryman (2000) also presents a very representative scenario that shows the potential of Web services.

Roles in a Web Services Architecture

Web Services has its roots on distributed systems and it is an evolution of the work on that area. They both have the same philosophy behind, which is the well-known concept of "distributed but cooperating" systems. But the way that the "cooperation" takes place is different. While the stress is on the distribution of the "data" in distributed systems, it is on the distribution of the "tasks" in Web services. In the Web Services view, every different system or component offers some services for some others. Any system does its job by just calling suitable services over Internet. Once a Web service is deployed, other applications and Web services can discover and invoke that service. From this respect, one can even derive the analogy between the philosophies behind Web Services and Object-Orientation.

There are three components of the Web services architecture (fig. 1). These are service provider, service broker and service requestor. Service broker is sometimes referred to as service registry (Cerami, 2002). The interactions among these components involve publishing, finding and binding Web services. In a typical scenario, a service provider hosts a network-accessible software module, the Web service. The service provider defines a service description for the web service and publishes it to a service broker. The service requestor uses a find operation to retrieve the service description locally or from the service broker and uses the service description to bind with the service provider and invoke the Web service implementation (Kreger, 2001). Service broker is responsible for service registration and discovery of the Web services. The broker lists various service types, descriptions, and locations of the services that help the service requestors find and subscribe to the required services.

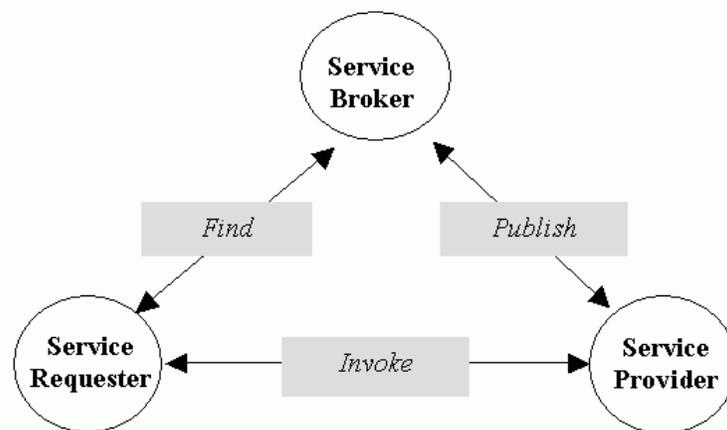


Figure 1. Web services architecture (Nagappan et al., 2003)

Web Services Technologies

Web Services can be developed using any programming language and can be deployed on any platform. Web Services can communicate because they all speak the same language: the Extensible Markup Language (XML). Web Services use XML to describe their interfaces and to encode their messages. XML-based Web Services communicate over standard Web protocols using XML interfaces and XML messages, which any application can interpret.

The three core XML-based technologies for building and enabling Web services are:

- Simple Object Access Protocol (**SOAP**) defines a standard communications protocol for Web Services.
- Web Services Description Language (WSDL) defines a standard mechanism to describe a Web Service.
- Universal Description, Discovery and Integration (UDDI) provides a standard mechanism to register and discover Web Services.

When a service provider wants to make the service available to service requestors, he describes the service using WSDL and registers the service in a UDDI registry. The UDDI registry will then maintain pointers to the WSDL description and to the service. When a service requestor wants to use a service, he queries the UDDI registry to find a service that satisfies his needs and obtains the WSDL description of the service, as well as the access point of the service. The service requestor uses the WSDL description to construct a SOAP message with which to communicate with the service (Systinet, 2003).

The need for Web services in Traditional governments

In Turkey, the traditional governments have many problems that stem from both insufficient and improper use of ICT. "Insufficient use" refers to the traditional means such as manual archiving systems. "Improper use", on the other hand, refers to the lack of an interoperability infrastructure within and among the government agencies. In this work, a number of such problems have been identified for the Municipality of Trabzon in Turkey. Although they have been determined for a municipality, they are also valid for such sectors as G2G, G2C, G2B in the Country. World Bank (2002) has also identified similar problems for many developing and even developed countries around the World. To identify these problems, some of the activities of the various departments of the Municipality were closely investigated (Şahin, 2003). The selected activities were the ones that involve spatial data and are most characteristic within the context of e-municipality or e-government. Identified problems were classified under five different though interrelated categories. These are the lack of auto-control mechanisms, high economical losses, high cost of services, poor service quality, and low efficiency. Some of these problems can be clearly seen in many activities of the Municipality. One such activity is laying out the Zoning Plan Forms (ZPF), which is explained below.

A Zoning Plan Form (ZPF) geometrically shows the location of a land parcel in the zoning plan, and includes construction conditions of the zoning plan block that covers the parcel (Figure 3). ZPF data are required by many applications of municipalities. For instance, obtaining a ZPF is the first step in getting a building permit. Laying out a ZPF involves combining Cadastral and Land Title data with the zoning plan data. In Turkey, Cadastral and Land Title data are under the responsibility of Cadastre and Land Title Offices, which are Central Government organizations. Zoning plan data is handled by the ZPO of the municipalities. In short, the activity involves combining the data from different providers. Figure 2, shows a ZPF, involved data and their legal and effective providers. By the "legal provider" we mean the legal responsible and owner of the data. By the "effective provider" we mean the supplier who supplies the data to the municipality in traditional system.

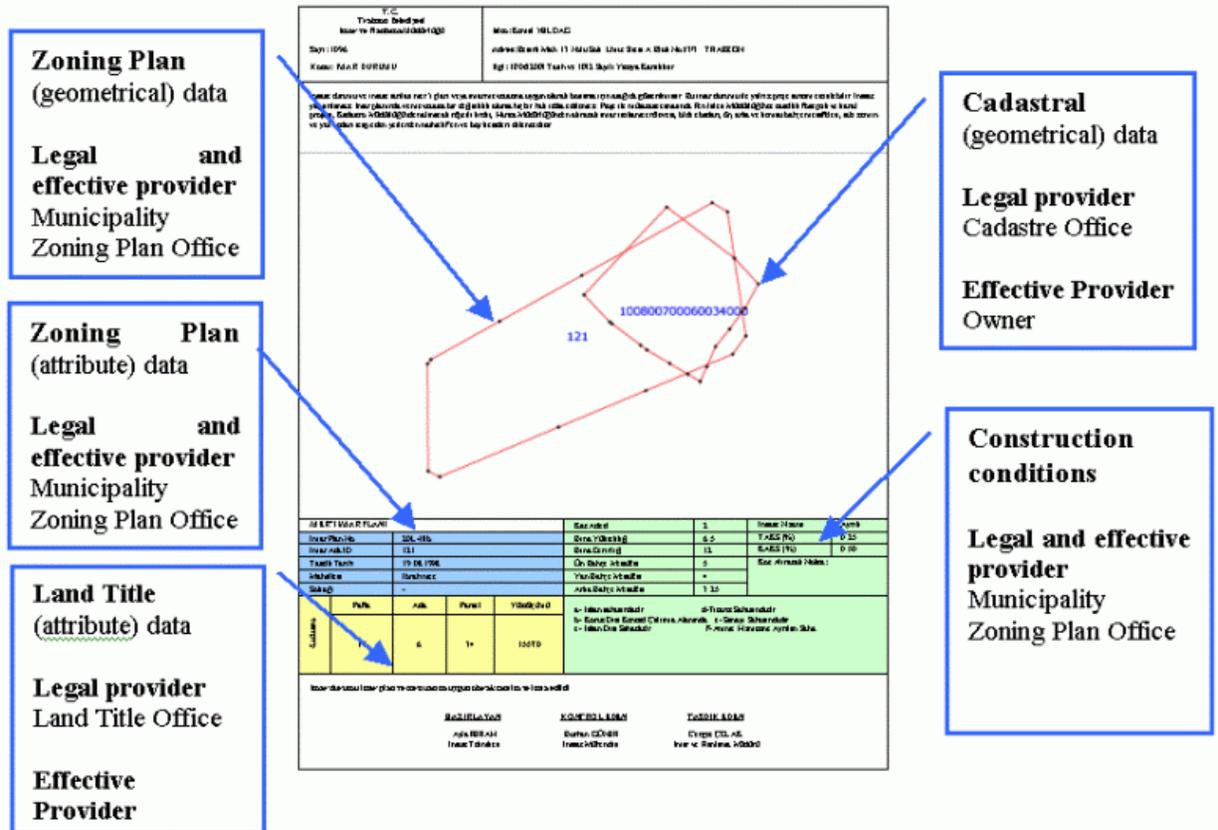


Figure 2. Zoning Plan Form, involved data and their legal and effective providers.

ZPFs are given by the Zoning Plan Offices (ZPO) of municipalities upon the request of the "interested" which might be citizens, government agencies, private sector, or municipality departments. For the sake of clarity, it is assumed here that it is the citizen. The steps a citizen has to follow to get a ZPF are illustrated in Figure 3. The steps of the citizen's workload are as the following:

1. He goes to the ZPO of the municipality and asks for a ZPF for the parcel of interest. ZPO requests Cadastral and Land Title data from him.
2. He travels in the city and to go to the Cadastre office for the Cadastral data. For the citizen, this means lengthy legal procedures and may involve frustrations due to the rather slow pace of the activities. For the officer it involves finding the related map sheet and performing a manual clip on the part containing the parcel. If the citizen is lucky, he can get the job done on the same day.
3. He goes to the Land Title Office for the Land Title data. If Cadastre and Land Title Offices are not in the same building as in some cities, the citizen has to have another travel for this. For the citizen, once again, this requires lengthy legal procedures may involve frustrations due to the rather slow pace of the activities. For the officer it involves locating the related record via usually a computerized database interface. Again if he is lucky he can get the job do on the same day.

4. He goes to the municipality by another travel and presents the data he had been collecting. He, in most of the cases, could not get the ZPF immediately. Because a ZPF is laid out by manual or semi-automated methods. And either of the methods takes time. So, he is advised to come back two to three days later and get the ZPF.

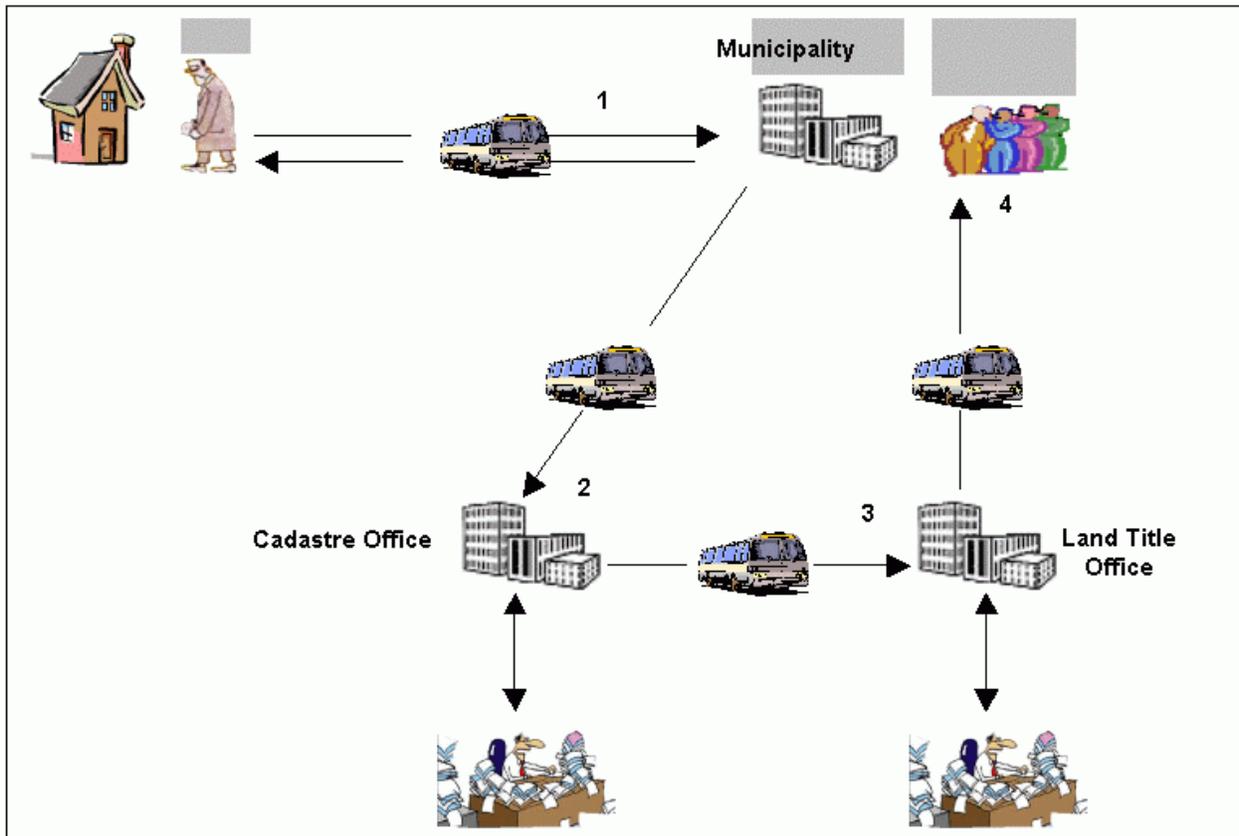


Figure 3. The workload of a citizen in getting a ZPF.

It can be concluded with an optimistic view that, getting a ZPF would take at least a whole working week. And this, in no ways, is an acceptable from the side of either the requestor or the provider of the service.

Some of the problems of the traditional system such as low pace, high cost, and the poor quality of services are clearly noticeable in this activity. Once again the main cause of these problems is the lack of an interoperability infrastructure among the data providers of the activity.

We may also spell out corruption due to the involvement of Land Title and Cadastre Offices. The basis of this statement is the Country wide statistics which show that Land Title and Cadastre Offices are one of the least counted agencies.

Web Services for e-Government

A number of Web services for the investigated activities of the Trabzon Municipality have been designed and partially implemented. These services were developed and deployed using Cape Clear Software (Cape Clear, 2003). A municipality officer can develop his applications using

these web services in a very short development time. It took 15 minutes to build a *getParcelZoningPlan* application that produces ZPF in a fully automated fashion. We currently use static binding of the services for service composition (Systinet, 2003). The *getParcelZoningPlan* is also a web service that calls other web services to do its job. The *getParcelZoningPlan* will be a service of the Zoning Plan Office (ZPO) of the municipality. The user or the officer in ZPO will initiate this service from his Internet browser. A brief explanation of what happens then is given below:

1. A user sends a parcel's RealEstateID using a HTML Form. *getAParcelsZoningPlanStatus* Web service invokes *getParcelGeometry* Web service in LCD server using RealEstateID. *getParcelGeometry* returns an array that includes parcel's attributes, vertice coordinates, and bounding box coordinates.

2. *getAParcelsZoningPlanStatus* Web service invokes the *getZoningPlanID* Web service in ZPD server using parcel's bounding box coordinates. Thus, we find the zoning plan's ID(s) that cadastral parcel is located.

3. *getAParcelsZoningPlanStatus* Web service invokes the *getZoningPlanBlock* Web service using parcel's vertice coordinates. This service takes vertice coordinates and finds zoning plan block ID(s) using java.awt.Polygon class. Then, it creates an array that includes block ID(s) and coordinates.

4. *getAParcelsZoningPlanStatus* Web service invokes the *overlayBlocksAndParcel* Web service. It takes two arrays that includes cadastral parcel, and zoning plan block(s) as an input parameters, and creates a GML file. Then, transforms result map from GML to SVG using a XSLT stylesheet.

5. The *getParcelInfo* Web service is called. It returns an array that includes parcel's attributes.

6. *getZoningPlanBlockInfo* Web Service is called. It returns an array that includes zoning plan block's attributes.

Finally, ZPF is prepared using all returned data by web services, and sent back to the user in the form HTML (Figure. 4).

How a citizen gets a ZPF in the web services scenario is as given below:

1. The citizen goes to the ZPO officer as for a ZPF in a short time. He does not have to actually go to the municipality; He can make this request over the Internet.

2. ZPO officer initiates *getParcelZoningPlan* Web service from his internet browser

3. The *getParcelZoningPlan* calls other Web services to get the ZPF done. These services get the needed data on-line from the remote databases that are Cadastre, Land title, ZPO databases. The services do also the necessary processing.

4. The officer takes ZPF output from his printer and submits it to the citizen.

If there is no problem with the communication lines, getting a ZPF would take seconds or minutes. This is a great improvement compared with the traditional way. Therefore, we can say that Web services provide solutions to the aforementioned problems of the traditional system.

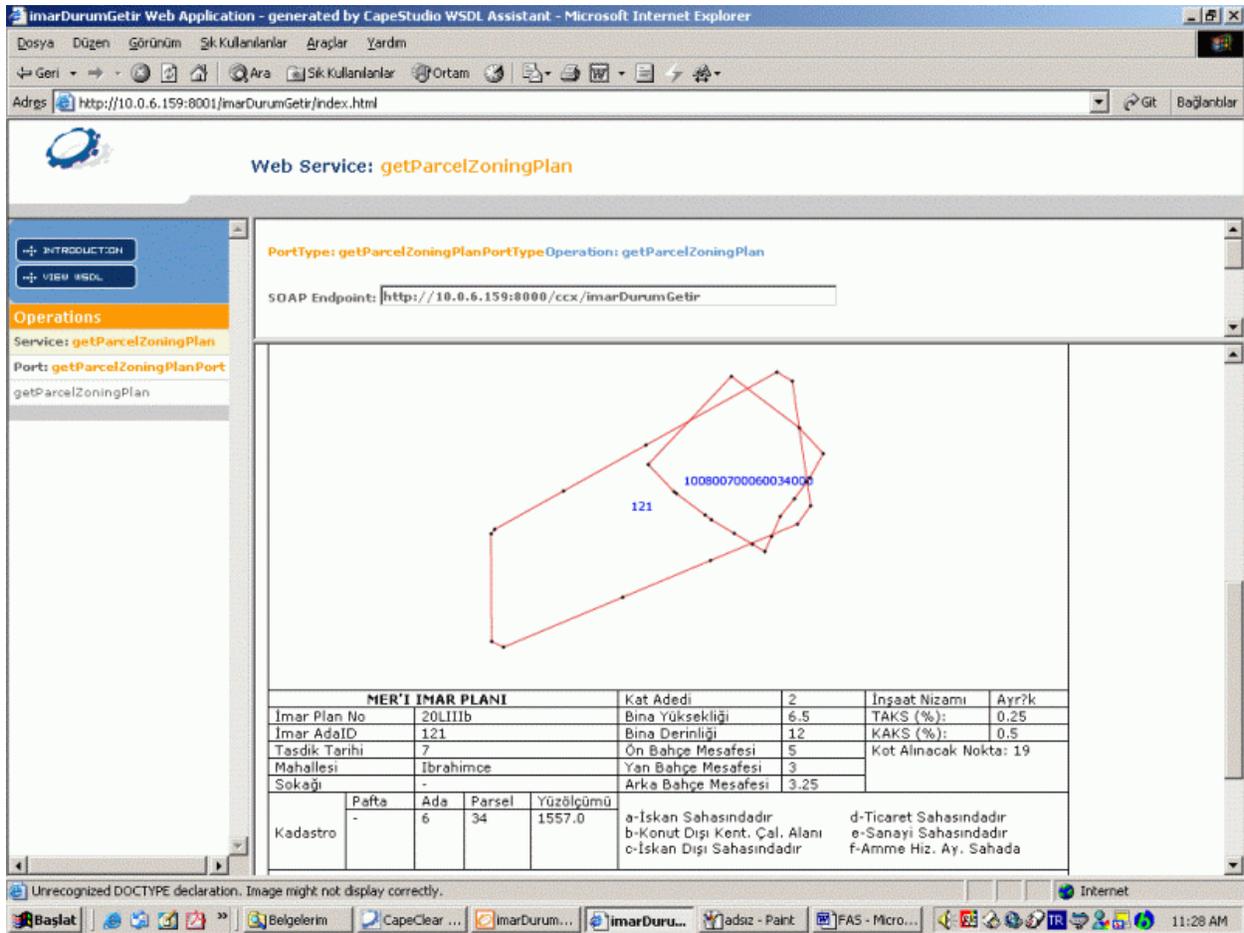


Figure 4. Response of *getParcelZoningPlan* Web service

Potentials and issues

In this part, the solutions that Web services promise for the problems of the traditional system are identified. Also the issues a web services based e-government may confront are touched upon. Web services will enable "on-line" or "e-services". G2B, G2C, and G2G transactions will be performed over Internet. Without actually going to the municipality and spending plenty of time in frustration, citizens and the other parties will be able to request information, pay their taxes, get building permits, get zoning plan forms and so on over the Internet. Besides improving the service quality, this will make great contributions to the economy. For instance, there have been 70-90 % outages in real estate taxes in Turkey. And a considerable part of this loss is due to the tedious procedures involved. Web services would eliminate this effect. Authors of this paper have already proposed and implemented a system called REIS (Real Estate Tax Information System) which would be a very important component of the e-municipality (Cömert, Akinci, 2002). A web services implementation of REIS is available now. REIS will collect and calculate real estate

taxes using web services. The only thing the citizen or the interested has to do is to pay his taxes. And he could do this either by conventional means or on-line through the Internet.

Web services will enable "quick" services. Citizens and other parties can get answers to their requests within seconds. Efficiency of services will make valuable contributions to the economy. Consider how the time to get a building permit will affect the construction business and related sectors, for instance. The high pace of services would also mean high quality of services.

High quality services will make positive sociological impacts on the people of a country. High quality services will make people feel honored by their government and they will feel the joy of being served well. This will make them happy and greatly improve every single person's productivity. This continues to be a dream concerning the reality in Turkey.

Web services will enable new services. Since managing the traditional tasks would be easier with Web services, municipality departments will have more time to spend on specializing their services. For instance, the Real Estate Tax Office (RETO) might offer web services for the real estate private sector. Private sector may get connected to RETO and ask for certain information such as the owner, location on the city map, zoning plan conditions etc. about real estates.

Web services will cut down application development and service provision costs. Because once a number of main services are defined, new services can be built on the existing ones in a short time with a minimum development cost. As already mentioned, developing the *getParcelZoningPlan* service has taken only 15 minutes. Also, data transfer costs will decline due to both the data transfer medium and reduced transfer requirements of a Web services environment.

Despite all its merits, Web services are not free of problems. Some major issues are related to security and privacy. Medjahed et al. (2003) makes a clear distinction between security and privacy and points out that privacy is more difficult to deal with than the security. Web services are also criticized for what they had originally been proposed for; interoperability. However, we do not deal with these issues in this work whose purpose was to seek the feasibility of using Web services for an e-municipality or e-government infrastructure.

The Web services paradigm has been envisioned for a notion of environment where everybody provides some services for some others. Then anybody does its job by just calling or combining the suitable services. Therefore, a Web services based interoperability infrastructure for e-government will rely on the web services of many others involved. For instance, Land Title and Cadastre units must offer services needed by the municipality as well as other parties. Furthermore, the activities of Land Title and Cadastre and municipalities are not limited to each other. There would be other parties involved; various government agencies, private sector, universities, citizens and many others. Therefore, what is needed actually is the NSDI for e-municipality and e-government to work. And building an NSDI is not an easy job. It requires time, hard work, devotion, and cooperation. The time is a particularly critical here. Because NSDI projects take long time to be completed. But e-government projects do not have to wait for NSDI. Rather, e-government projects should be undertaken in line with the NSDI projects. In any case, priorities have to be determined well. Related to NSDI also is the name, "NSDI" which may

sound obsolete. Given the implications of the Web services it is not the "data infrastructure" anymore. However, that is another discussion we might face in the coming years.

Conclusions

The need for interoperability, the ability by which the different applications can talk and cooperate with each other, is more pressing than ever. Interoperability infrastructures are needed in such sectors as business-to-business (B2B), government-to-citizens (G2C), government-to-business enterprises (G2B), and government-to-government (G2G) to increase revenues and to enable high quality, low cost services. In this work, we have proposed Web services for e-government interoperability infrastructure. We have compared traditional and Web services scenarios for a traditional service of the Municipality of Trabzon, Turkey. As a result we have found that Web services enable quick and high quality services. In addition web services would enable cost savings in software development and service provision. Web services do have problems like security. We have not tackled these problems in this study. We have only examined the usability of a new technology, Web services, for the e-municipality infrastructure. Since e-government and e-municipality share similar structures and problems, the results of this work is valid within the e-government context as well.

References

1. Berners-Lee, et al. (2001) Tim Berners-Lee, James Hendler and Ora Lassila, The Semantic Web A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities, Scientific American, May 2001.
2. Cerami (2002). Cerami, E. Web Services Essentials, O'Reilly&Associates Inc., Sebastopol, 2002
3. CGDI (2001). Canadian Geospatial Data Infrastructure (CGDI), version 1, CGDI Architecture Working Group, December 2001.
4. Cömert, Ç., Akinci, H. 2002. Application Development in an interoperable GIS environment: A new system for real estate taxation in Turkey, Proceedings, Vol. I, p.200-205, 3rd International Symposium on Remote Sensing of Urban Areas, June 11-13, Istanbul, Turkey.
5. Kreger (2001). Kreger, H, Web Services Conceptual Architecture (WSCA 1.0), IBM software group.
6. Medjahed et al (2003). Medjahed, B, Rezgui, A., Bouguettaya, A, and Ouzzani, M., Infrastructure for E-Government Web Services. IEEE Internet Computing, IEEE Computer Society, Volume 7, Number 1, January/February.
7. Nagappan, et al, 2003. Nagappan, R., Skoczylas, R. and Sriganesh, R.P., Developing Java Web Services, Wiley Publishing Inc., Indiana.
8. Nokia, (2003) Web Services Interfaces - engine of innovation, <http://Searchwebservices.techtarget.com>.
9. OeE (2003). E-Government Interoperability Framework (EGIF), Office of the e-Envoy, Part1: Framework, Version 5.0.
10. Reed (2002). Reed C., A framework for web-based Geospatial Services, The OGC View on Web Services, GIM International, 30-33, July 2002.
11. Şahin (2000). Şahin, N. Designing Web services for e-municipality, MSc. Thesis, KTU Geodesy and Photogrammetry Department, August 2003, Trabzon, Turkey (in Turkish).
12. Thomas (2003) Thomas, M.A. Web Services: A Manager's Guide, Addison-Wesley.
13. World Bank, 2002. E-government handbook for developing countries, Center for Democracy and Technology, World Bank.

14. W3C (2002) Web Services Architecture, W3C Working Draft 14 November 2002, World Wide Web Consortium web sitesi, [http://www.w3.org/TR/2002/WD-Web servisleri-arch-20021114/](http://www.w3.org/TR/2002/WD-Web%20servisleri-arch-20021114/)