Cascading Web Services to Bridge Wired and Wireless Networks for Data Transactions

Miranda Kajtazi  
Master Student in Computer Science at Växjö University  
Stallvägen 17.311  
352 52, Växjö, SWEDEN  
+46 73 788 72 70  
mirandakajtazi@gmail.com

Bahtijar Vogel  
Master Student in Computer Science at Växjö University  
Stallvägen 17.311  
352 52 Växjö, SWEDEN  
+46 73 788 72 70  
bahtijarvogel@gmail.com

ABSTRACT
This paper describes and introduces ‘Cascading Web Services’ representing a collection of services offered in a system of multiple interacting and platform independent networks. Cascading Web Services are demonstrated as a technical design solution to the communication of devices using request-respond commands by cascading these commands between different Web Services. These services are self-contained and independent on their context or state. The system signifies a typical Service Oriented Architecture based on a distributed system.

Keywords
Cascading Web Services, Ubiquitous Computing (UbiComp), Service Oriented Architecture (SOA), Distributed System, Mobile ad-hoc Networks, N-Tier, Two-Tier.

1. INTRODUCTION
Information and Communication Technologies have been developed very rapidly in recent years. Multiple transport platforms of networks correlated with Web Services support intermediate processes, delivery of information, easier and rapid access to database systems.

Cascading Web Services intend to examine intelligence in a distributed system where devices are present and ubiquitous in the environment. Ubiquity integrates computation into environment enabling people to move and interact with the system at the same time [6].

User interaction with the system will be unconscious to trigger information from one Web Service to the other. This is performed by wrapper Web Service which creates an automatic transmission of information, designing the system to interact with users without manual intervention [6].

Ubiquitous computing revolutionized the way of interaction and data transfer between users, by putting information into environment and integrating computation into our everyday life, everywhere and anywhere [1].

Mobile technology that is encompassed by Ubicomp is a new area of modern technology. It supports and handles web applications with Web Services, resembled to web-based applications that run on powerful and traditional desktop computers [5].

Communication processes between mobile world can become more interesting in contrast with traditional static and wired communication, possessing innovative features such as providing the possibility to sequentially use request-respond commands through Web Services over interoperable multiple networks. Designing such communication enables data transmission to be manipulated from wireless to wired networks [3].

Existing services and traditional static and wired network systems are distinctive, such as: Bluetooth, Wi-Fi, GPRS, Wired Networks, which de facto do not cooperate with each other at the same time within one system for involving the user to interact with it [3]. While with cascading Web Services we represent a new technological innovation, creating a huge environment that involves all network platforms to be interconnected through communication channels, transmitting data and essentially bridging the wireless with wired networks.

These heterogeneous networks can always access WWW and many other services available via devices connected through WAP, GPRS, UMTS, Bluetooth, etc. [1].

Transmission of information and access to services is ubiquitous, which impacts full user mobility by using wireless devices connected with wired devices. This offers the capability to communicate through distributed systems and also through mobile ad-hoc networks [7].

2. CASCADING WEB SERVICES IN A DISTRIBUTED SYSTEM
The system design of Figure 1 uses multiple Web Services that perform data transactions between diversity of devices that are connected through Bluetooth, Wi-Fi, GPRS and wired networks. This system potentially creates a technical solution that can retrieve data content from any device and through different types of communication platforms (Bluetooth, GPRS, Wi-Fi, TCP/IP).

---

1 Service Oriented Architecture (SOA) is a software architecture that defines a relationship between two or more services.

2 Users can access data while they are in motion.
The process does collecting and transmitting data content through Cascading Web Services.

Independent communication environments have personal web servers that allow their slaves to manipulate with data as Two-Tier architecture.

Interconnected transport platforms present a distributed system divided in some networks such as location based (Bluetooth) so the user can access the database and Web Services locally from the first tier which in fact the retrieval process is much faster considering that this is the first level for accessing one Web Service. Otherwise, a user can cascade through Web Services for finding the proper response.

These independent communication platforms (i.e. Bluetooth only, or GPRS only etc.), are interconnected together that shape an N-Tier architecture. This allows communication between devices through all transport services. Interconnection is based on Service Oriented Architecture implemented through Cascading Web Services.

In general, Web Services offer an infrastructure to interconnect different platforms without boundaries [4]. Operation of Web Services is platform independent based on XML as the main feature for communication in distributed systems [4]. Combining and implementing Web Services within a diversity of communication platforms (Bluetooth, GPRS, Wi-Fi, TCP/IP) determines the trigger behavior of collecting and storing data content.

3. GENERIC ARCHITECTURE

3.1 Cascading Web Services Architecture

The relationship and principles of the system architecture play an important role for data manipulation. Therefore we proposed a system architecture design (illustrated in Figure 1) that shows all communication services for access control, used for controlling transmission of data specifically with mobile devices.

This architecture design explains the communication of devices using request-respond commands to the web server. It follows these commands cascading from one Web Service to the other Web Service.

Our system design is based on an architecture level explaining the capacities of wireless network that can be bridged with wired network through internet. Request and respond data transmission can be done in three ways according to distributed systems architecture as presented in Figure 1:

1. Unicasting.
2. Multicasting.

Data transmission works by collecting and storing all the necessary information in databases. Data transactions are done directly by users that store and get data from the databases such as text, images, audios and videos.

Communication process, concerning the collaboration of networks for data transmission which is presented in this section is also shown in Figure 1. This is an architecture which represents transport and session layer involved for proceeding data transactions.

![Figure 1. SOA using Cascading Web Services.](image)

According to the figure above, the intention of exploring and designing such architecture is to show a group or collection of services that communicate with each other that are self-contained and independent on their context or state.

This figure represents a typical Service Oriented Architecture that works within distributed systems [2]. This kind of communication between different networks and devices facilitates and opens new opportunities for data transaction. Cascading Web Services are introduced by involving multiple transport networks including for instance, Bluetooth Technology, GPRS, Wi-Fi, and Wired Networks.

Bluetooth connection corresponds to a typical “location based” network allowing the creation of PAN without cables or wires.

GPRS as a mobile data service available to users of GSM, transmits - receives data in packets.

While TCP/IP is a very suitable communication protocol used here to connect hosts in internet through wire or Wi-Fi.

All these network technologies create interoperability that use exchanged information from one network to the other via Web Services residing in Web Servers having the data placed in databases. Indeed, this is bridging the gap of non-proper communication and data transmission between such networks.

3.2 Internal Architecture – Data Flow

Application logic, distributed data processing and data flow, primarily rely on a connection-based, Two-Tier architecture.

Integration of multiple Web Services that cascade for the data flow is N-Tier architecture of the distributed system; data interaction is platform independent; transactions are scalable and support different data content.

---

3 Pan-Area Network.
Inter-process communication of this distributed system handles all processes proceeded by the user interacting with the system. The data flow network as it is described in Figure 2, executes processes sent by the user. Transactions are performed among Web Services that automatically retrieve data and return it to the user. The retrieval process of Web Services creates a data flow between databases (Figure 2), which interact with each other. This operation creates an environment of multiple independent platform networks that work together as a distributed system where the data flow plays an important role for the overall communication.

4. CONCLUSION
This project is still in progress. It is an innovation technology that uses existing transport platforms to develop a creative, intelligent and interactive application for the user. It will work in a huge system of different transport platforms. We will continue to investigate and test the system by providing a real application in hand. Its design and implementation will show the interaction of the user within a distributed system. The intelligent process of Cascading Web Services will be provided by wrapper which purpose is to combine and relate these services with each other in order to show the automatic interaction.

Third-party will be involved to experiment and test the robust, flexible and the adaptable application.

5. ACKNOWLEDGMENT
We thank “CeLeKT” Research Center at Växjö University for helping and improving our idea for the project.

6. REFERENCES