Improving Mobile Complex Service Composition with Graph Technology

Ljupco Antovski[1], Marjan Gusev[1] and Pece Mitrevski[2]

Abstract. **The advances in mobile and wireless technologies enable cost effective usage of e-government services. The case of e-government and m-government in Macedonia is outlined and discussed. The latest findings from the developed platform in the M-GOV projects are presented. The M-GOV platform is based on simple ideas that together provide architecture with high level of flexibility and low levels of technological requirements. The architecture is designed to scale, both from a technical and a financial perspective and to be applicable to integration scenarios from small agency scenarios through to large administration scenarios. The M-GOV architecture is formed of three major components: Citizens access devices, Service Discovery Directory and the Collection of public electronic services from various sources. The built-in M-GOV ontology is discussed. The selection and composition of services based on predefined graphs of belonging is presented. Some results concerning service discovery, composition and consumption of services are outlined.**

Keywords. **Mobility, ontology, service composition, fuzzy logic, graphs, pervasive, electronic public services, m-government**

I. Introduction

E-government efforts aim to benefit from the use of most innovative forms of information technologies, particularly web-based Internet applications, in improving governments' fundamental functions. These functions are now spreading the use of mobile and wireless technologies and creating a new direction: mobile government (m-government). M-government is defined as the strategy and its implementation involving the utilization of all kinds of wireless and mobile technology, services, applications and devices for improving benefits to the parties involved in e-government including citizens, businesses and all government units [Kushchu, 2003] .The main point in favour of m-government is the fact that mobile phone's penetration is reaching an 83% rate surpassing fixed telephony subscriptions in Europe (Netsize 2005, IDABC 2006).

[1]Ljupco Antovski and Marjan Gusev are with the Institute of Informatics, Faculty of Science, "Ss. Cyril and Methodius" University, Arhimedova 5, P.O Box 162, 1000 Skopje, Macedonia, Email: anto@ii.edu.mk, marjan@on.net.mk

[2]Pece Mitrevski is with the Faculty of Technical Sciences, University "Sv. Kliment Ohridski", I. L. Ribar bb, 7000 Bitola, Macedonia, E-mail: pece.mitrevski@uklo.edu.mk

Innovative and radical services applied through mobile technology may significantly improve the operations and communication efficiency of governments. Such innovative service is to extend public administrative transactions in ways that actively involve citizens allowing them to communicate and collaborate with public administration systems through their mobile devices in a transparent and trusted environment.

Currently the main focus in the service composition is discovery of appropriate services according to the user's needs. If no service is available, then a complex service should be composed, consisting of several services fulfilling the overall goal of the user. In the M-GOV project we propose fuzzy logic ontology that enables soft selection of services. It enables better flexibility in the selection of partial services.

The main hypothesis of the research is the idea to cache the ontology properties of every service during the design and registration phase and to use these results in selection of services during the composition phase. This approach will enable narrowing the search time in great manner.

II. M-GOV Project

The M-GOV (Mobile Services for Government) project is a research and innovation project at the Institute of Informatics, designed to encourage the access to new mobile and wireless public electronic services. The project is based on close cooperation between Public Authorities, SMEs and Universities. The main goal of M-GOV is to contribute to a development of a new cost-effective open public service platform for mobile citizens. The new platform will support usability, openness, interoperability and scalability. It will introduce the business model of close cooperation among the service providers, public authorities and citizens. The main M-GOV innovation is the Service Discovery Directory. It will enable discovery and instant consumption of new and available M-Government services in the current location of the mobile citizen wherever in Europe.

The M-GOV project aims to meet the following objectives: New models specifying how mobile services for multiple ambiances will be coupled, integrated, assembled and offered by various service providers; New communication platform that can support highly personalized mobile services, while adapting to different networks and protocols; Plug and Play environment for new mobile services; Implementation of optimized interfaces among the service providers, public authorities and citizens; and Innovative Service Discovery Directory.

The Service-oriented architecture is believed to become the future e-government technology solution that promises the agility and flexibility the business users have been looking for by leveraging the integration process through composition of the services spanning multiple agencies (Channabasavaiah 2004). SOA is an approach to loosely coupled, protocol independent, standards-based distributed computing where software resources available on the network are considered as Services (Arsanjani 2005).

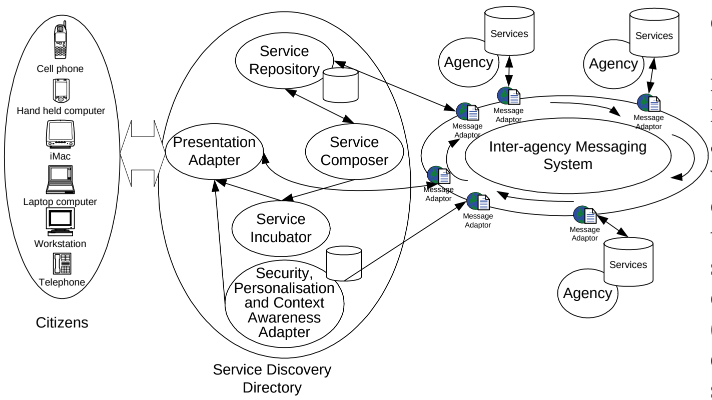

Fig. 1 M-GOV Generic Architecture

The M-GOV architecture as shown on Figure 2 is based on the above architecture and is mainly formed of three major components: Citizens access devices, Service Discovery Directory and the Collection of public electronic services from various sources.

The citizens can access the electronic public services from anywhere, anytime from any Internet-connected device. The Service discovery directory consists of presentation adapter responsible for transferring the data in citizen device's acceptable form, Security Adapter with PKI infrastructure responsible for the security and privacy mechanisms in the framework, Service Repository that minds all the required information for available service consumption, Service Composer responsible for composition of complex services, and Service Incubator, a component responsible for state preservation, personalization, quality of service and adaptation of the current service in use.

The M-GOV platform communicates with the available public electronic services using an innovative circular messaging system. The agencies and the Service Discovery Directory use message adaptors to send and receive messages in predefined format. The goal of the message adaptors is to enable communication among systems working on different platforms.

III. Inter-agency Messaging System

The basic idea for the Inter-agency Messaging System (IAMS) is to use a flexible XML format called Mobile Government Extensible Language (MGML) in binary form (optimized for mobile environment). The flexible format will enable implementation of new services with different content. The format of the new service will be reported in the salutation phase with the Service Discovery Directory and stored in the Service Repository.

The IAMS utilizes the concept of message adapters to wrap services from the technology used by other services. The IAMS provides a variety of message adapters that give services the illusion that all other services use the same messaging transport technology they do. This approach provides a high-level of technical interoperability between agencies that have heterogeneous technologies. Of course, for true interoperability agreement amongst the business owners over the structure and semantics of data is also required.

On the M-GOV IAMS, the synchronous abstraction is also made available to the services without compromising on robustness or reliability by means of a message adapter. This adapter gives the consuming service a synchronous invocation view of any M-GOV exposed service. This message queue is created, maintained and managed by the IAMS on behalf of the second service. It is up to the business owners of second service to pull their messages from the IAMS managed outbox. This has a number of loose coupling advantages (Channabasavaiah 2004). The core of the design is a simple data message routing, auditing and security. In this manner, services can be started, stopped, upgraded without bringing the M-GOV system. This is obviously important in M-Government architectures where very high availability is a required.

IV. Ontology Graph

It is expected that the SOA systems in near future will have to handle the very large number of available Web services which expectably will be several millions.

Most of the implementations use logical reasoning and keep all relevant knowledge in the working memory, which limits the number of resources tremendously.

The most interesting approach for overcoming this is to reduce size of the search space and the number of Web services that need to be taken into consideration for semantic matchmaking between a requested and provided functionality. We find that the most suitable realization is to group available Web services in a way that exhibits the properties of an efficient search graph. One can organize Web services with respect to the provided functionalities that allow achieving a higher scalability of service detection mechanisms. Most adequately, this can be achieved by organizing available Web services in an efficient search graph.

(Stollberg 2007) defines semantic matching as four degrees exact, plug-in, subsume, intersect as different situations wherein the basic matching condition is satisfied, and disjoint denotes that this is not given. We feel that using a more humanoid approach in expressing degree of maching is better. In this manner we propose to use soft fuzzy values for maching condition values. One service can mach with an ontology not only with crisp values (0 or 1) but in the whole spectrum from 0 to 1.

Without any optimization, the computational costs for service detection are in linear time O(n) with n denoting the search space as number of available Web services. The efficiency of the system can be radically improved if the information used in the service design (registration in SDD) phase is structured and latter used in the discovery phase. In this manner we narrow the search space only for the services that in some degree match a specified goal ($O(n_u)$, where n_u is the number of services that meet the required goal) The design phase includes the registration of the services in the repository with complete ontology specification and fuzzy values for matching degrees. The information is structured in graph consisting of inner nodes (ontology values) and leaf nodes (web services). M-GOV in the discovery phase requests the necessary information from the graph composed in the design phase of the system. Then M-GOV uses a heuristic strategy that allows it to constrain the queries sent to the repository. This heuristic relies on the fuzzy evaluation of the specified preferences.

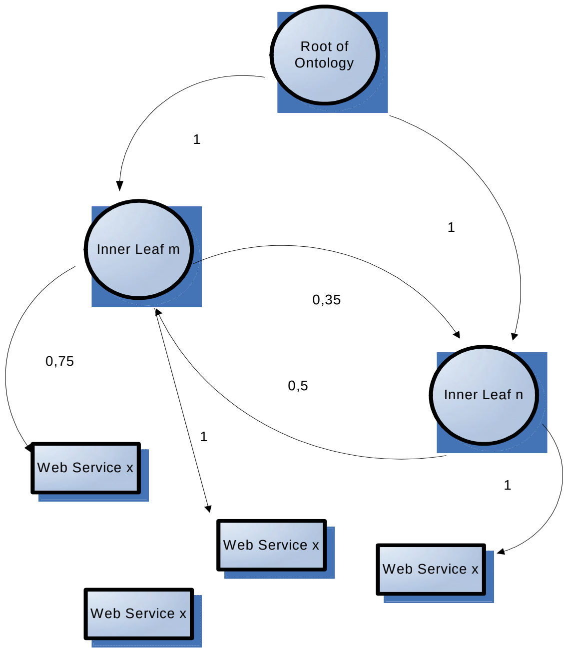

Fig. 2 Ontology Graph

The graph is constructed (Figure 2) with every inner node being a goal template that is described by defined ontology value. Every leaf node is a Web service. The links are mediators that connect nodes and define the matchmaking degree between them, such that the inner nodes are connected by mediators with similarity expressed with fuzzy value from 0-1 and each leaf node or Web service is connected by mediator to the inner node to which it belongs with the greater mediator value also expressed with fuzzy values from 0-1. When a service is registered in the repository, then it is added in the graph. Then all the links-mediators to the specified ontology values are added. If some matching information in the service specification is available for different ontology values, than that information is added also to the graph as inner links..

One service may have matching degrees with several ontology values (complex goal). That web service has links – mediators to all the matching ontology values (goal leafs). This approach improves the search efficiency in the discovery phase, as one should only check if links from all the ontology values in the complex goal lead to a specific service. If no service can be found than a complex services is composed solely on selection of services that match the single specified ontology values (single goals) in the complex goal. In all the cases, the very first web service that has the greatest mediation degree is selected.

For instance, there is ontology for museums in the future European capital. There is an inner leaf 1 in the graph Austrian, with web service 1 for information about a museum only for the 19th century. The mediator is 0,6 There is an inner node 2 Hungarian with no services in the end nodes. The correlation mediator between inner node 1 and 2 is 0,4 bidirectional. If a mobile user searches for a museum for Hungary, there is no mach, but considering he will be offered the web service 1 with belonging value to the ontology of 0,6*0,4=0,24. In the language of fuzzy logic if we have uniform triangular distribution with values as Non Appropriate, Little Appropriate, Appropriate, Very Appropriate and Complete Match, it will be that the service that is found is Very Little Appropriate.

VI. Discovery, Composition and Consumption of Services

The Service Discovery Directory (SDD) contains components for Service Discovery, Service Composition, Service Selection and Service Execution, all of which are involved in the creation of the complex service offerings. Service discovery and composition in M-GOV is not a standalone function. It relies heavily for its innovative qualities within a Security, Personalization and Context Awareness component. Service discovery uses personalization within service selection, to select the service offerings for a particular service type that are most suited for a specific user profile. Personalization uses context parameters.

Service discovery is the mechanism by which services conforming to a certain set of criteria are found in the Service Repository. It will return all services that support any of the discovery mechanisms as used by the M-GOV that conform to the criteria given. If the appropriate service is not found than the Service repository starts the internal service called Search Adapter. It uses a filter engine that evaluates a query against the discovered information from the comprehensive search of services in the M-GOV Platform. The service discovery component is only one step in the overall service composition function. It is usually called several times during the construction of a composite service. This component is responsible for the discovery, registration and deregistration of all services within the M-GOV Platform. When the list of services for one set of criteria has been found, they are then transferred to the Service Composer. This module then interacts with Security, Personalization and Context Awareness Adapter.

In a typical scenario, the user with mobile device (MU) sends a composition request message to the SDD. This is done using the standard request protocol. This message must contain two components, the initial state and the goal to achieve. After receiving the request, the SDD cannot immediately perform the composition and before that, it must contact the repository to retrieve the descriptions of relevant services.

When the repository, its message must include a sequence of OWL complete service descriptions. After receiving the set of services that will be used for composition, the SDD performs the second step, which refers to internal processing and the planning process itself. Once the planning completed successfully, the SDD converts the plan to a composite service description and informs the MU about the service.

Selecting the set of possible services to form the plan is not an easy task. Several heuristics can be applied. In this section we propose a more informed method for filtering services that makes use of classification of services. We developed a framework for service-based filtering, and then instantiated it to different filters on the basis of context information obtained from the role ontology and the service category derived from the directory structure.

The service composition planning problem can be conceived as follows: Let P = {p1, p2… pn} be the set of all possible composite services for a given service request and S = {s1, s2… sm} the set of input services for proper service composition. The goal of a filter F is to select a given number X of services from S, such that the search is reduced, but the best plan of P is still found. However, it is obvious that this ideal case is not realistic since the problem would be already solved. In principle, one can build up matrices for every possible query.

Once the service has been composed, it is the Service Incubator that looks after the composed service instantiating it as a complete service and monitoring it during its lifetime. The Service Incubator is also responsible for service adaptation. Service adaptation must not be confused with content adaptation. Service Adaptation encompasses content adaptation and more. Service adaptation can be carried out both during service composition and while the service is running. The adaptation of the service can be triggered by changes in context of the user. The adaptation occurs as the system constantly attempts to match the users' needs and the available resources and capabilities of the service, by constantly monitoring changes.

The Service Incubator closely cooperates with the Presentation Adapter. The Presentation Adapter actively responds to the information received from the Service Incubator and adopts the presentation of data to the appropriate quality, quantity and demand of the presentation device used at the given moment. For instance, a citizen reads the flash news from a governmental agency service on his/her PDA while coming to work. As he/her enters her office and logs in to his/her computer, the Presentation Adapter actively responds with enabling a full screen enriched data presentation. But not only the presentation is changed, but the quantity also, because the bigger screen enables presentation of more detailed information. You are requested to strictly follow the author's guidelines.

VII. Final Remarks

The M-GOV platform is based on simple ideas that together provide architecture with high level of flexibility and low levels of technological requirements. The architecture is designed to scale, both from a technical and a financial perspective and to be applicable to integration scenarios from small agency scenarios through to large administration scenarios. The core technology concepts employed all proven technologies that are well suited to use in building a long lived M-Government infrastructure.

The system is implemented in laboratory scale. Future work includes tests on the functionality implemented in comprehensive real environment, and validation of the platform against similar systems, from the perspective of personalised service composition and service adaptation. It is also hoped to extend the adaptation process within the platform to optimize service performance in response to changing context conditions.

We feel that the impact of mobile technologies on government administration is huge as this will have implications for the success of widely adopted M-Government applications. In order to consume the benefits a sound standardized solution as the M-GOV attempted is to be defined.

The problem of composition of complex services has many proposed solutions. All of them are mainly based on QoS parameters selection or on transforming the ontology description to First Order Logic. M-GOV eases the constraints in the design of the services and enables human description of ontology belonging using Fuzzy Logic and Graph technology. We feel that this approach will enable better and easier composition of appropriate services in the m-government area.

References

[1] Ali Arsanjani, How to identify, specify, and realize services for your SOA, IBM Corporation, February 2005

[2] Antovski, Lj. And Gusev, M. (2005), M-Government Framework, Proceedings of the First European Conference on Mobile Government, (editors Ibrahim Kushchu and M. Halid Kuscu), 10-12 July, University Sussex, Brighton UK"

[3] Heeks, R. and Lallana, E. (2004), M-government benefits and challenges,http://www.e-evexchange.org/eGov/mgovprocon.htm

[4] IDABC (2006), eGovernment in the EU member states, eGovernment Observatory, IDABC Report, April 2006, 4th edition

[5] Jason Bloomberg, The SOA Implementation Framework, http://www.zapthink.com, April 2004

[6] Jeff Hanson, Coarse-grained Interfaces Enable Service Composition in SOA, JavaOne, August 2003

[7] Kishore Channabasavaiah and Kerrie Holly, IBM White Paper, Migrating to a Service-Oriented Architecture, April 2004

[8] KIT(2005), National Strategy for Information Society Development, http://www.kit.gov.mk , April 2005

[9] Kushchu, I. and Kuscu, H (2003), "From E-Government to M-Government: Facing the Inevitable", in the proceedings of European Conference on E-Government (ECEG 2003), Trinity College, Dublin

[10] Meta Group White Paper, Practical Approaches to Service-Oriented Architecture, November 2003

[11] Michel Stollberg. Semantic Goal Caching. Technical Report, DERI, 2007.

[12] Ministry of Justice of Macedonia (2006), E-Election List, http://www.izbirackispisok.gov.mk, May 2006

[13] Netsize (2005) 'The Netsize Guide', web: http://www.netsize.com, April 2005