An Agent-based User Selection Method for Business-aware Service Provisioning

Darko Grubisic, Kristina Kljaic, Igor Ljubi, Ana Petric, Gordan Jezic
University of Zagreb, Faculty of Electrical Engineering and Computing, Department of Telecommunications, Unska 3, HR-10000 Zagreb, Croatia
E-mail: {darko.grubisic, kristina.kljaic, igor.ljubi, ana.petric, gordan.jezic}@fer.hr

Abstract: **Third generation network operators are facing the issue of having to introduce new, personalized services to their users, while at the same time they have to increase revenue and cut down the operating costs to survive the increasing competition on the telecom market. Intelligent software agents can act on behalf of the network operator to allow the operator much more flexibility and a quicker adaptation to volatile conditions that dominate in a competitive telecom market. In this paper authors propose a new scenario for automated business driven service provisioning process. It utilizes the agent technology and the operator's business support system to provide the list of potential users during the planning phase of the service provisioning process.**

1. INTRODUCTION

Service provisioning in the second generation (2G) mobile telecommunication systems is developed and controlled directly by the network operator. Due to the complexity and costs involved in deploying a new service into the network (components that enable the service often had to be implemented directly into the network's core), network operators choose to deploy only selected services for which their business forecasts predicted that will be used by the majority of their subscribers [1].

The introduction of the 3G network, and the convergence towards an all-IP network, will force the operators to offer a higher number of services, as well as to introduce new services at much faster rate. Therefore, it is predicted that the traditional service provisioning process will be expanded with new participants. In addition to network operators and service users, new participants of the service provisioning process are Content Providers and independent Value-added Service Providers (VASPs) [2]. Figure 1 depicts all of the parties of service provisioning process as well as their relationships.

Network operators, Content Providers and VASPs establish Business-to-Business (B2B) relationships, through which new software and/or content for planned services is bought. This enables creation of personalized and context-aware services for users (established Business-to-Consumer (B2C) relationships). Network operators possess a large quantity of information about their subscribers (which includes their usual locations, services used and account balance). Since 3G advanced services have to be cost effective and rapidly deployed to users, the analysis of subscriber data could greatly help in achieving this goal.

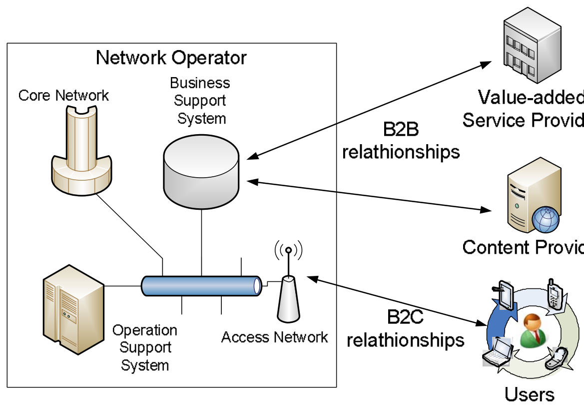

Figure 1 - Service provisioning process in 3G network

This paper focuses on the research aimed towards successful charging for the use of services. In order to make credit-worthy services, network operators have to adapt existent simple charging schemes to avoid potential revenue loss associated with either B2B or B2C relationship risks (e.g. content has been bought for a high price but small number of users have purchased it). While B2B risks can be managed at a higher level, those involving B2C highly depend on the fact whether the users actually use offered services, and therefore are creating revenue for the operator. Our approach utilizes mobile and intelligent agents to analyze and extract the information in order to determine the potential users for our planned service [3, 4].

This paper is organized as follows: Section 2 provides an overview of technologies used to create a proof-of-concept prototype for complete service provisioning process in a 3G network. Section 3 describes in detail the architecture of the proposed solution and explains the case study, while the Section 4 gives directions for future work and concludes the paper.

2. USED TECHNOLOGIES

2.1 Software Agents

A software agent is a piece of software that acts in behalf of its user or owner. Such "action on behalf of" implies the authority to decide when (and if) the action is appropriate. The idea is that agents are not strictly invoked for a task, but activate themselves when a certain event occurs in a system [5, 6, 7].

The proposed multi-agent system is implemented using the Java Agent DEvelopment (JADE) framework [9, 10]. JADE is a software development framework for developing multi-agent systems conforming to Foundation for Intelligent Physical Agents (FIPA) [11] standards for intelligent agents. Each running instance of the JADE runtime environment is called a Container as it can contain several agents. The set of active containers is called a Platform. A single special Main Container must always be active in a platform and all other containers register with it as soon as they start. The Main Container is an agent container where the Agent Management System (AMS) and the Directory Facilitator (DF) are running and where the Remote Method Invocation (RMI) registry, which is used internally by JADE, is created. The agent platform can be split on several hosts. Figure 2 illustrates the above described concepts through a sample scenario showing the split JADE platform on three different hosts.

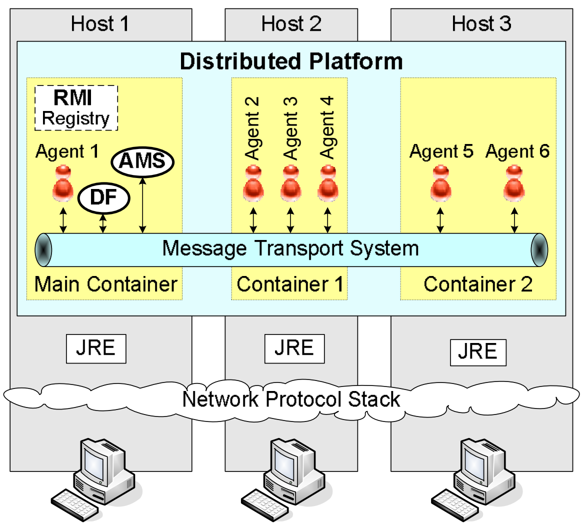

Figure 2 - JADE Agent Platform distributed over several containers

The AMS is an agent who represents the authority in the Platform. It administers supervisory control over access and use of the platform (for instance it is possible to create/kill agents on remote containers by sending a request to the AMS). It is also responsible for a naming service (i.e. each agent must register with the AMS in order to get a valid Agent IDentifier - AID). Therefore, the AMS provides life-cycle service and White Page service. Only one AMS exists in a single platform.

The DF is an agent who provides the default yellow page service in the platform (i.e. an agent can find other agents providing the service it requires in order to achieve his goals).

The Agent Communication Channel (ACC) is a Message Transport System that controls the exchange of all messages within the platform, including messages to/from remote platforms.

When a JADE platform is launched, the AMS and DF are immediately created and the ACC module is set to allow message communication. The Java Runtime Environment (JRE) enables end-users to run JADE platform.

Probably the most important feature that JADE agents provide is the ability to communicate. The communication is performed by exchanging asynchronous messages between the agents. Therefore each agent must have a sort of mailbox (message queue) where the JADE runtime posts messages sent by other agents. Messages exchanged by JADE agents have a format specified by the ACL language defined by the FIPA

2.2. Diameter Protocol

The Service Charging based on Diameter Protocol [12] is an optional feature part of the PrePaid System (PPS) [14] product. It is used for real-time cost and credit control of service and content charging between the PPS and Content Providers. The Diameter interconnection between Content Provider and PPS is shown in Figure 3.

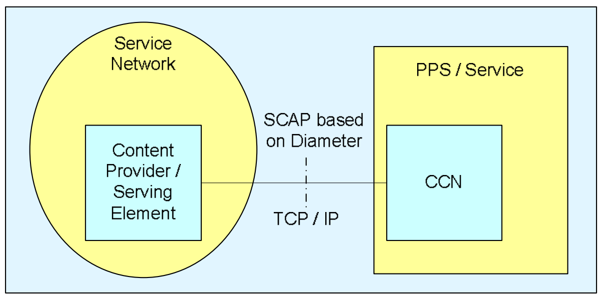

Figure 3 - Diameter interconnection between a Content Provider and the PPS

The service provided by the PPS uses the Service Charging Application Protocol (SCAP) based on the Diameter Protocol. The SCAP is a vendor specific protocol that adds Attribute Value Pairs ((AVPs) – The basic data primitives) to the Diameter Base Protocol's accounting commands (Accounting Request (ACR) / Accounting Answer (ACA)). The transport protocol used is Transport Control Protocol (TCP).

The Charging Control Node (CCN) is one of the nodes in the PPS. It is built on the Telecom Server Platform (TSP). TSP is a multi service platform that includes all necessary hardware, middleware as well as the required Operating System [8].

The nodes in the Service Network (Serving Elements) act as Diameter SCAP Clients interacting with the PPS which acts as a Diameter SCAP Server. The service implementation requires the Diameter Base Protocol to be used. Secure transfer can be obtained by adding a firewall and a security gateway function that supports, for example, Internet Protocol Security (IPSec). Authentication and authorization of the end-user can be obtained by implementing the mechanisms of Authentication, Authorization and Accounting (AAA).

2.3. PPS Diametar Emulator

The PrePaid System Diameter Emulator uses the Charging SDK (Software Developer's Kit) API (Application Programming Interface) [13] to emulate the behavior of the Diameter implementation in the PPS. It enables operators to perform the first basic tests of their applications without the need for real PPS. The emulator contains an account database, a currency database, and a tariff database just as the actual system does.

There are three configuration windows: currency, account and tariff configuration window. In the currency configuration window (Figure 4.) the user can define currencies the emulator needs to know about and convert between. Each currency has three fields: currency id, currency name and exchange rate. It is recommended that the currency is defined in the emulator before the account and tariff are defined since they use information from the currency configuration.

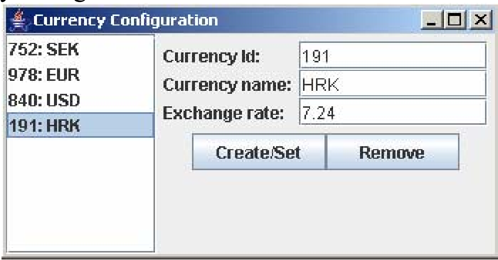

Figure 4 - The currency configuration window

In the account configuration window, displayed on Figure 5, the user can define the accounts. Each incoming request refers to a certain user so all users must have an account if the Diameter emulation processing should be successful.

Each account is identified by the Mobile Station Integrated Services Digital Network (MSISDN). This is an actual phone number, and it must be unique. The International Mobile Subscriber Identity (IMSI) is also an identifier like MSISDN. IMSI should be unique, although this will not be verified in the SDK. The account balance shows how much money in the specified currency is available on the user's account. Reserved is a field which displays how much money is currently reserved for processing the incoming request in the active session on the account.

In the tariff configuration window tariffs are defined. If the price for the incoming request is not pre-rated, the emulator must find the price of the service. The emulator finds the price for the requested service by searching for a match of the incoming parameters in the tariff list.

The tariff name must be unique. The name as such is not used while selecting the tariff. The tariff condition is entirely responsible for that. The tariff rate is the cost for each requested unit. If the rate is 5 and the requested units are 4, the total cost will be 20.

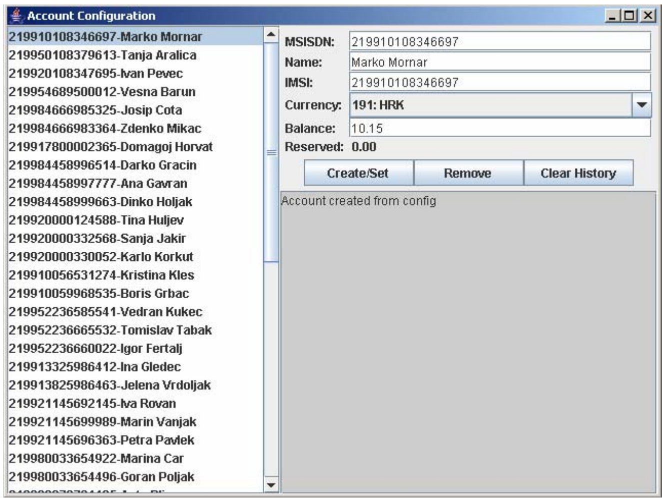

Figure 5 - The account configuration window

All parameters in the condition are optional, and a value will only be used if it is valid for the data type specified. Empty value fields means that the parameter is not used. The request has to match all the entered fields of the tariff in order to be valid. The first tariff that matches the parameters sent into the emulator is used, and will be used for the whole session [4].

3. BUSINESS-DRIVEN SERVICE PROVISIONING

With the rising complexity of 3G mobile services and the increased number of actors involved in the service provisioning process, service providers have to find a plausible and cost-effective way to ensure that the service will be delivered exactly to the users who are most likely to use it. Type of mobile equipment, user's behavior, location or account balance has to be taken into consideration during the service provisioning process. As a part of a larger solution, this paper proposes an agent-based model in which the operator can efficiently select users that will most likely desire to use a newly developed service. Figure 6 shows the simplified architecture of the system and the communication between the agents. The communication is based on FIPA Inform, QueryRef, and Request protocols.

After the Content Providers and Software Vendors release a new version of the content/software component for a particular type of a mobile terminal, they will send a notification to the *Business Manager Agent* (BMA). The BMA utilizes Java Expert System Shell (JESS) [15] and its rule engine to check whether all of the preconditions for a particular initial service deployment have been fulfilled. When all of the preconditions have been met, the BMA will initiate the Service Provisioning Process as described later in this section.

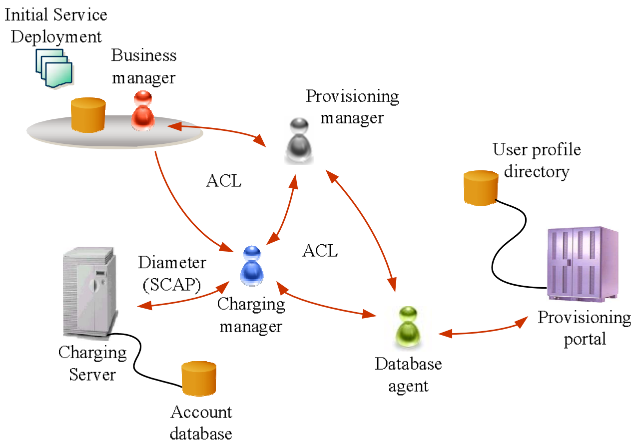

Figure 6 - Communication of the Charging Manager Agent

After that, the BMA first sends an ACL message to the *Charging Manager Agent* (CMA) with the appropriate business related information using the FIPA Inform interaction protocol. Each message represents one initial service deployment and contains deployment parameters: ID, content type, location, time, QoS and the user type. This is important since the CMA checks accounts of target users, according to the constraints specified in the notification sent by the BMA. The amount of messages being sent determines the maximum number of initial service deployments.

As soon as the CMA receives the ACL Inform message, the agent knows the specified type of users, but what it really has to find are actual mobile users (identified by their IMSI numbers). That is why it contacts the *Database Agent* (DBA). The DBA has the access to the database with user profiles and serves as an interface to the network operator's database that contains user information. This communication is supported by FIPA QueryRef interaction protocol. The DBA checks the database and returns the IMSI numbers of users that belong to a specified user type to the CMA.

While the BMA forwards parameters to the CMA, it also initiates the optimization process using the genetic algorithm to achieve the minimum operational costs [5]. Finally the optimal execution order is sent to the PMA using FIPA Request protocol. The calculated conformation represents the optimal order of initial service deployments. These messages contain the same parameters as the ones sent to the CMA by the BMA. The parameters are needed to find potential target users (for example the users who really want the particular mobile service and are within the location area denoted by the initial service deployment). The PMA has to select those target users that will receive notification about an upgrade of software components.

The CMA checks the received user accounts to see which users have enough money for a particular mobile service. The parameters that it received from the BMA (content type, location, time, QoS, user type) are needed to specify the tariff of the mobile service. The CMA connects to the Charging Server (CS) using the Diameter SCAP.

After receiving the conformation with optimal execution order from the BMA, the PMA checks the database with user profiles in order to find users suitable to a particular initial service deployment (potential target users) and finds out their IMSI numbers. With target users' IMSI numbers, the PMA sends a request to the CMA in order to check user accounts and find out which users have enough money for a specified mobile service.

Now, the CMA can finally make an intersection of both users' lists according to the previously specified initial service deployment ID and find out for the specified deployment which users want the particular mobile service and also have enough money on their account. The CMA sends the result to the PMA. Then, the PMA initiates and coordinates the deployment of software/content from the software/content providers to targeted users.

3.1. Potential communication problems

What will happen if, for example, the CMA can't access the CS (it may be unreachable for some reason, maybe the network is disconnected) or when the DBA can't access the database with user profiles for the same or some other reason? The CMA also might try to check a non existing user account, or simply, one of the agents might not have appropriate information when it is needed, etc. What will happen when an error occurs? Will the whole agent system fall down?

Luckily, this will not happen, at least it should not in most cases. These are the reasons why the communication between the agents should have security mechanisms.

The main security mechanism for the agent communication is the agent's protocol. It allows the agents to send a Refuse and a Failure message when something does not go according to the plan.

If the CMA receives the Refuse message from the DBA, it indicates that the DBA can't access the database with user profiles for some reason. If the CMA receives a Failure message, it knows that an error occurred, and can handle it in a special procedure. If this happens, the CMA will not even try to access the CS (because it has not received the IMSI numbers needed to check the balance) and later when it receives the Request message from the PMA, the CMA will answer with a Refuse message.

The Diameter SCAP also has security mechanisms. For example, preventing the use of a service (direct debit) when the user's credit has expired or is spent; or the credit control server detects fraudulent attempts in case a fraud detection system has been incorporated with the credit control server. In the case above, the credit control server will reject the request by returning a specific result code that instructs the service node to reject the service request.

A potential problem in this work was the fact that the agents' communication is asynchronous. It means that there is no way to know which message will be sent first, second, and so on. Also, browsing databases and checking user accounts are actions that require some time (for example, user profile database of this project contains fifty users, but the real user profile database might have a million or even more users). It is possible or even probable that the CMA will receive several messages from the PMA while it is still processing the first message.

It is also possible that the CMA receives a Request message from the PMA before it has prepared an answer for the PMA (the process of checking user accounts is still in progress). In this case the CMA will send an Agree message to the PMA and later when the process of checking user accounts is finished, it will send an answer to the PMA.

3.2. Configuration of the Charging Server

We configured user accounts and service tariffs in the database of the CS that is represented by the PPS Diameter emulator. In order to test this work user accounts were configured with various amounts of balance.

The user accounts that are checked are determined by the process of selecting target users performed by the PMA. The result of this process (target users' IMSI numbers) is the input for checking the account information.

For testing purposes we defined mobile services, for which we created and configured service tariffs. There are twelve defined mobile services: Business news, Wireless video weather, Post-it, Presence&Location, Sightseeing, Urban music, Celebrities, Sport call, Inside the movies, Comedy, Paying bills and Doom. Each mobile service contains several parameters and according to them tariffs for each mobile service were defined.

We identified seven important parameters for each tariff: traffic case, service provider, location, content type, time type, user group and quality of service (QoS). Differentiation of the tariffs was made according to location, QoS, season and type of the user group. One of the limitations of the SDK we used is that all parameters have to be numerical.

5. CONCLUSION AND FUTURE WORK

This paper presented efficient business-driven user selection method for service provisioning process in 3G mobile networks. The usage of mobile intelligent software agents resulted in the automation of the selection process based on the information about user's equipment, behavior and account balance. As a result, the network operator is able to more efficiently and with lower cost select to which users a new service will be offered.

Future work will involve the research on how to perform actual charging of advanced multimedia services in 3G networks.

ACKNOWLEDGMENTS

Work presented in this paper is the result of the FER/ETK Summer Camp 2006, supported by Ericsson Nikola Tesla, Croatia.

REFERENCES

[1] Houssos, N., Alonistioti, A., Merakos, L., Mohyeldin, E. Dillinger, M., Fahrmair, M., Schoenmakers, M., "Advanced adaptability and profile management framework for the support of flexible mobile service provision", IEEE Wireless Communications Magazine, Special Issue on (R)Evolution towards 4G Mobile Communication Systems, Vol. 10, No.4, 2003.
[2] Geng, X., Whinston, A. B., Profiting from Value-Added Wireless Services, IEEE Computer, August 2001 (Vol. 34, No. 8), pp. 87-89.
[3] Vrancic, A., Jurasovic, K., Kusek, M., Jezic, G., Trzec, K. Service Provisioning in Telecommunication Networks using Agents and Rule-based Approach, Proceedings of the 30th International Convention MIPRO 2007, Rijeka, 2007. pp. 159-164.
[4] Dumic, G., Podobnik, V., Jezic, G., Trzec, K., Petric, A. An agent-based optimization of service fullfilment in next-generation telecommunication systems, Proceedings of the 9th International Conference on Telecommunications ConTEL2007, Zagreb, pp. 57-64.
[5] J. M. Bradshow, Software agents, Cambridge (USA), MIT Press, 1997.
[6] D. N. Chorafas, *Agent technology handbook*. New York, McGraw Hill, 1998.
[7] W. T. Cockayne, M. Zyda, *Mobile agents*. Greenwich (USA), Manning Publications, 1998.
[8] Ericsson "Programmer's Guide - Service Charging Based on Diameter, PPS 3.6", http://www.ericsson.com/mobilityworld/sub/open/technologies/charging_solutions/tools/diameter_charging_sdk .
[9] JADE Administrator's guide, http://jade.tilab.com/.
[10] JADE Programmer's Guide, http://jade.tilab.com/.
[11] FIPA web site, http://www.fipa.org/.
[12] Diameter Base Protocol, http://tools.ietf.org/html/rfc3588
[13] Diameter Charging SDK, http://www.ericsson.com/mobilityworld/sub/open/technologies/charging_solutions/tools/diameter_charging_sdk.
[14] Prepaid Postpaid Convergent Charging, White paper, http://www.ericsson.com/technology/whitepapers/3027_pre_post_conv_charg_A.pdf.
[15] Jess Web site, http://herzberg.ca.sandia.gov/jess/.