

# Agents Technology Extended With Mobile Devices

Fábio Calhau<sup>1</sup>, Lino Pereira<sup>1</sup>, Paulo Costa<sup>1</sup>, Luís Botelho<sup>1</sup>

<sup>1</sup>“We, the Body and the Mind” Research Lab”, ADETTI / ISCTE,  
Avenida das Forças Armadas 1600-082 Lisboa Portugal  
{fabio.calhau, lino.pereira, paulo.costa, luis.botelho}@iscte.pt  
<http://adetti.iscte.pt>

**Abstract.** This paper presents framework that can be used for improving working conditions. We have extended intelligent agents technology for the Internet with cell phone technology. This extension allows the mobile worker to have access to the full power of server based computation using cheaper, smaller mobile devices. The most demanding part of the computation is performed by agents running on Internet servers while cell phone computations are concerned only with the interface between end-users and server-based agents. Although we have focused only on Cell Phones, the same technology can be used with Pocket PCs, PDAs and other mobile devices.

## 1 Introduction

Mobility has become the principal aspect for the enterprises. Becoming mobile means being reachable anywhere any time, making the work more profitable. Combining mobility with agents technology is using the advantages of both technologies to create applications that were never thought to be available in a mobile scenario. This will bring an enormous advantage since it will contribute to reduce the necessary physical work space, to increase the geographic coverage of the enterprises (Gateway Group, 2003), and it will also contribute to decrease car traffic and consequently to improve environmental quality. With this technology, people can work while they travel, improving their comfort, their quality of life in general, and their productivity.

We have used a combination of mobility and agents technologies in a concrete multi-agent system that allows people to easily find restaurants of their preferences and to book tables for their meals. This application has been developed by ADETTI's *We, the Body and the Mind* research lab in a partnership with Vodafone Portugal.

The paper describes the generic multi-agent system architecture and technology (Section 2), and details the mentioned application (Section 3).. Section 4 compares our proposals with related work. Finally, Section 5 presents conclusions and future work.

## **2 Agent Technology Extended With Mobile Devices**

This section presents the reference model for multi agent systems using the proposed extensions for mobility and describes adopted technological solutions. In our framework, cell phone programs are not seen as autonomous agents but as interfaces to existing agents running on Internet based servers. Each cell phone is associated to an agent running on an Internet server.

### **2.1 The Mobile Solution for Agent Technology**

Mobility is a necessity in our days. Many of the things we used to see in non mobile environments are now becoming available in mobile environments. Agent technology is not an exception. This paper presents an approach for the integration of mobility with agent technology. Because agent platforms are heavy and mobile devices do not possess the same computational power as a server, instead of using mobile devices as agent platforms, we have used them only as interfaces associated with agents running on Internet servers. With this, most of the computational effort is done by the server instead of the mobile device. This is the best solution for the integration of the two technologies, given the computational limitations of the current generation of mobile devices.

In our proposal, the only agents aware of the existence of mobile devices are those associated with the mobile interface. The other agents completely ignore their existence. This way, all agents in the application communicate with each other using a single kind of communication technology.

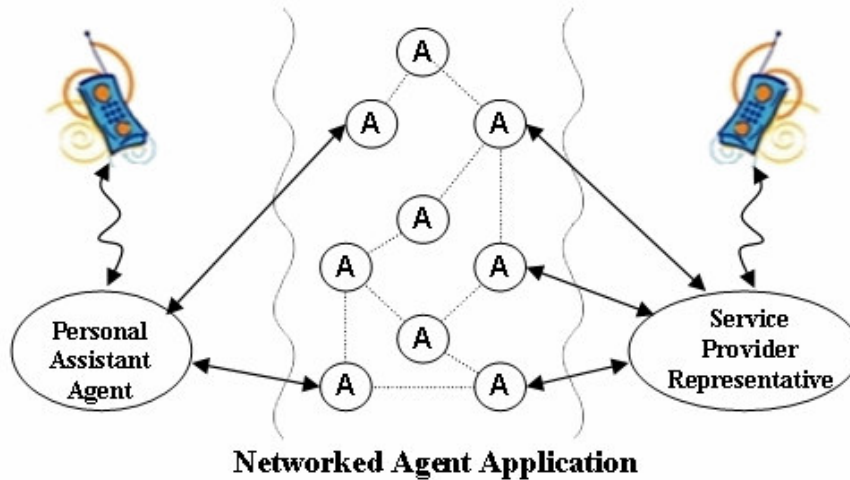
Using the power of agent computation associated with mobility several new types of application can be developed. The architecture presented in this paper is an example of the power and flexibility of these types of applications since it represents a generic framework for developing several types of applications based on service discovery and execution. Agent technology provides flexibility and value added service composition to the user. Mobility allows the user to access this powerful structure in any place, while traveling or while going to work, improving reachability and productivity.

Since one of the greatest advantages of agent technology is its modularity. New functionalities can be implemented very easily by just adding new agents providing new services. This modularity enables users to extend the functionalities available on their mobile devices just by upgrading or adding new interfaces to existing agents without ever worrying about the complexity of the agent network necessary to fulfill their demands.

### **2.2 System Architecture**

The system backbone in the proposed architecture is a service providing agent network with personal assistance agents and basic service representatives. These are the end points of the agent network. Personal assistance agents provide personalized user access to the networked agent application. Service representative agents provide the

agent interface that allows the integration of component services in the networked agent application. Finally the agent network between personal assistance agents and service representative agents creates all sorts of value added services for people from the basic component services. These may range from simple information agents such as yellow pages agents, to service brokers, negotiators, reputation agents, and service composition planning agents.



**Fig. 1.** Reference Architecture

The fundamental idea is the association of mobile devices such as cell phones with Internet agents. Since cell phones and other mobile equipment are used by the end-user, the Internet agents to whom they are associated are personal assistance agents and service representative agents (**Fig. 1**).

The tight association of terminal mobile devices with personal assistance and service representative agents allow users (consumers and producers) to have access to the full computational power of networked agent applications using small less powerful devices. Most of the computation is provided by the agents to whom they are associated and by the other agents in the network. Equally important, this type of organization allows other agents in the network to forget about the existence of mobile equipment with specific requirements. Mobile equipment programs are hidden by the agents to whom they are associated from the other agents in the network. Therefore, agent interaction works exactly as if no mobile equipment were integrated; only a single interface is used. The interaction with the mobile equipment is completely handled by the specific agents associated with that kind of interface device.

### 2.3 Adopted Technological Solutions

This section presents the technological solutions we have adopted to keep the mobile device programs as simple and as small as possible, to signal mobile device users that they should activate the interface programs on their mobile equipment, and for the communication between mobile devices and associated Internet agents.

**Bootstrapping and configuration.** In order to keep the mobile equipment computational requirements to a minimum, the programs running on the mobile equipment are as simple and as small as possible. At bootstrap, they request the necessary information from the Internet agents to whom they are associated in order to proceed. The requested information consists mostly of configuration data. We have used the Java Micro Edition to write the programs in the mobile devices. The communication between mobile devices and associated Internet agents uses serialized Java objects over the HTTP protocol.

Since more than one mobile device associated agent may be running on the same computer, we use a database containing the port of the internet agent to whom the mobile device is associated. All that is required is for the mobile device interface program to use its unique identifier to ask the agent port from the database in the Internet based computer. This is a very flexible process coping with possible changes of the ports of the Internet agents. Even if the Internet agent is temporarily disconnected, when it resumes its operation and its port is settled, it registers the port in the database. This way, mobile devices will always have access to the actual ports of their associated Internet agents.

Once the port is acquired, the mobile device program sets the communication channel with its agent so that it can ask for and send information to the agent.

**Short Message Service.** Using currently available technology for cell phones, it is impossible to trigger the interface program on the side of the mobile device when information is available for being processed. In order to overcome this difficulty, we use SMS messages to signal mobile device users that they should activate the interface program on their devices so that the available information is processed. For instance, when there is information in the personal assistance agent that should be displayed in the user cell phone, the personal assistance agent sends an SMS message to the cell phone signaling the user that the interface program should be executed.

**Cell Phone-Agent communication language.** We have developed a domain-independent communication language for the interaction between the mobile device interface program and the associated Internet agent. This language includes a set of communicative acts including requests, queries and information messages.

### 3 Application Description

This section describes the application we have developed in which the proposed framework has been tested. In the implemented application, users find restaurants of their preference using their cell phone interface with the networked agent application running on the Internet. Once the restaurant is found, users may book a table for their meals. Table booking is done through the communication between the restaurant representative agent and the restaurant cell phone.

The implemented system is not a complex and sophisticated system but it is fully implemented whose main goal is to be used as a proof of concept for the propose approach.

### 3.1 System Architecture and Functionality

In the restaurant table booking application, there are personal assistance agents, an intermediate agents network, and restaurant representative agents. Each user has its own personal assistance agent, and each restaurant has its own restaurant representative agent. The intermediate agent network is formed by a single information agent containing summarized information about all existing restaurants and the identifications of their representatives. The personal assistance agent accepts restaurant preferences and table booking requests from the user, through his or her cell phone interface, and interacts with the intermediate agents' network to discover a restaurant satisfying the user preferences. Once the desired restaurant and its representative agent have been found, the personal assistance agent directly sends the table booking request to the restaurant representative agent. This uses the cell phone interface to the responsible person in restaurant to negotiate the table booking. The responsible person in the restaurant may accept the required booking or suggest changes such as the hour of the meal. The acceptance or the suggested changes are sent to the restaurant representative agent, which sends them to the personal assistance agent. The acceptance notification or the suggested changes are then presented to the user through the cell phone interface. The user just has to accept or decline the possible changes. If the user does not accept the possibly suggested changes, he or she may initiate another restaurant discovery process.

If desired, the interaction between users or restaurants and the agent application can be done through a web page instead of the cell phone interface. This possibility is not further explained in this paper since its main focus is the use of mobile device interfaces.

Besides the described restaurant discovery and table booking operations, the implemented application allows other operations. Each time a restaurant creates a representative agent, the restaurant data has to be filled in the restaurant representative. A restaurant having a representative agent may modify its data. Finally, representative agents must update the summarized information regarding its restaurant in the intermediate restaurant information agent. **Fig. 2** represents the table booking protocol described above.



**Fig. 2.** Table Booking Interaction Protocol

When the personal assistant agent (PAA) sends a table request to the restaurant representative agent (RRA), this may accept the table request specified conditions, it may reject the request altogether, and it may also suggest changes to the specified conditions. For instance, it may suggest a different time for the meal, two tables instead of just one, or one table as specified but only in the smoking area.

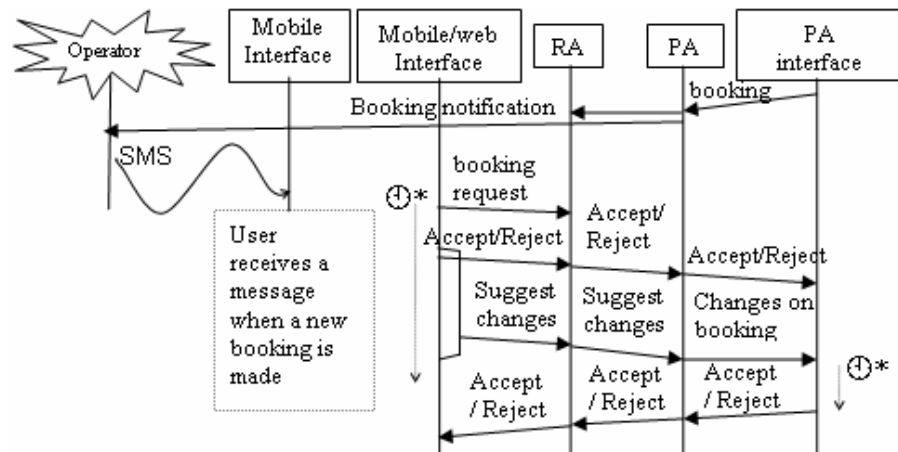
If the RRA accepts the specified table request, the client is automatically bond to the request. If the RRA suggests modifications to the original request the client cannot suggest counter-modifications. The client may only accept or refuses the suggested changes.

Finally, if the table booking request is actually done, the RRA informs the PAA of the assigned reservation number.

The protocol depicted in **Fig. 3** governs the interaction between the RRA (and possibly the associated cell phone) and the PAA (and possibly the associated cell phone).

Since it was not possible to automatically activate the interface program in the restaurant cell phone when a message is received, the restaurant representative agent sends an SMS message to the restaurant cell phone informing the restaurant responsible person that a reservation request is ready for being processed and thus the cell phone interface application should be launched.

When booking a table, there is mandatory and optional information accompanying the table booking request message. Mandatory information includes preferred meal hour, number of people, and the name of the person responsible for the reservation. Optional Information includes smoking / non-smoking preferences, table location preferences, and special features specifications (e.g., facilities for the disabled such as wheel chair facility).



\* - The response to the booking request and the response for the suggested changes are time limited. After that time if there is no response the system assumes that it is negative

**Fig. 3.** UML Interaction Protocol

The SMS message sent to the restaurant cell phone has five states, which can be visualized on the cell phone screen.

1. “Not answered”, means the SMS message is not answered or visualized by the restaurant responsible person
2. “Visualized”, means that the restaurant responsible person has visualized the message but has not sent the respective reply (rejected, accepted, or suggestion of alterations)
3. “Pending”, means that the confirmation message of the booking or the Suggest changes is hanging, waiting to be accepted or rejected by the client
4. “Rejected”, means the client has rejected the received booking confirmation or suggestion for alteration of the original booking from the restaurant.
5. “Accepted”, indicates that the client has confirmed the booking or accepted the suggestions for alteration of the original booking indicated from the restaurant

### **3.2 Agent-Mobile Devices Interaction**

As referred above, we have developed a domain independent communication language for the communication between mobile devices and associated Internet agents. This communication is based on serialized java objects that are sent through http requests. This is used in all operations that require sending information from the mobile device to the associated agent or the other way around. For this to happen both the mobile device and the associated agent must be aware of the types of objects that can be sent.

The use of serialized java objects is preferred because less information is sent, and although it may seem that the communication will be dependent of using JAVA it is not quite so. Since there is no serialization in JAVA Micro Edition, we had to develop our own serialization classes and, by doing that, we ensured that the serialization process is language independent. Each JAVA object is simplified to its most simple parts, which are serialized and then aggregated to form the final result of the serialization. Since only the simple object parts are transformed into bytes, and since this simple data objects are common in every object oriented language, it is possible to deserialize the object using a different programming language.

When talking about mobile devices the amount of information sent is very important because the communication costs are high, less information means less cost. Another important subject when referring to mobile devices is bandwidth availability. Since the device has limited bandwidth, our application only sends information to the associated agent when it is necessary (i.e. when changing the restaurant data the application only sends the new data when the user finishes all the changes he wants to make). By doing this the number of connections is significantly reduced.

### **3.3 Agent-Agent Interaction**

Unlike the connection between the mobile devices and the associated agents, the connection between the agents is fixed, allowing a different kind of communication. Inter-agent communication uses ACL and SL communication and content languages.

The ACL and SL have been developed by FIPA and are increasingly used in agent communications (FIPA 2002a),

When an agent associated with a mobile device receives a request from the device it has two options. It can either solve the problem by itself or it can send an ACL/SL request message to another agent to solve the problem, receive the response from that agent and then send the response to the mobile device. The interaction between agents involves more communication when making a booking request, which requires all the application agents to participate. When making the booking request several ACL/SL messages are sent, asking for the list of restaurants satisfying the user search criteria, or consulting detailed information about the selected restaurant, and booking the table. The booking request is initiated by the personal agent, which sends an ACL/SL message to the representative agent associated with the selected restaurant .

### **3.3 Implementation Technology**

We have used the JADE agent platform (Bellifemine 1999), which is a FIPA compliant agent platform (FIPA 2002a). We have used FIPA ACL (FIPA 2002b) and Extended FIPA SL (FIPA 2002c)(Botelho et al 2002) as the agent communication and content languages over the HTTP protocol. We have used serialized JAVA objects over the HTTP protocol for the communication between cell phones and Internet agents. Finally, we have used JAVA for the Internet agents' implementation and JAVA Micro Edition for the cell phone interface programs implementation.

## **4 Related Works**

Mobility is a very important issue as described in (Chlamtac and Redi, 1998). Using agents extended with mobile devices, we associate the interisteng computational power, flexibility and autonomy of agents with the ubiquity of mobile devices.

In the MAP Agent platform (Tomarchio et al, 2000), mobile users are supported through the migration of agents from host to host of the network, being possible to sleep and awake agents.

In order to overcome the comparatively little computational power of mobile devices, the in the Ruse system (Landay and Kaufmann, 1993), the mobile device acts only as a keyboard and display, being the computation done in remote applications. This is exactly the same approach we have taken in the described system. . On the other hand, Ruse system doesn't take advantage of the agents architecture, because it just doesn't use it.

In Ruse, popup boxes and scrollbars where used to fit the information in the small screen space. We have used scrollbars, optimized menus, and especially, we have tried to reduce the amount of displayed information to the indispensable.

Another reason it is important to reduce the amount of information is the limited bandwidth available toPDAS and cell phones (Wooldridge and Ciancarini, 1999).

In the MOVE project (Carrega et al, 1999) a scenario is presented where a website is accessed by the mobile client to search for a hotel. In this application a self-interface cell phone is used to avoid problems with page presentation and to minimize



the user interactions (e.g. scrolling, box selection, etc), since walking through the information in a cell phone is a lot more laborious than in a laptop or in a desktop.

In the MOVE project, as in our system, user requests are sent to the agent through HTTP.

There are some search engines for cell phones as Vodafone Portugal service “*perto de mim* [close to me]” (Vodafone, 2003) where a place (like a bar or restaurant) close to you is searched, or as WithAir Mobile Search Engine (Kawai et al, 2002), where keyword predictive navigation and regional information search is used. With our model, we go further than simple searching. It is possible to make reservations, interact in the process with the restaurant through a defined message protocol, register our own restaurant, and authenticate the restaurant account through the insertion of login/password.

## 5 Conclusions and Future Work

In this paper we present a proposal to extend traditional agents technology with mobile devices technology. The main contribution of the proposal consists of avoiding overloading mobile devices with heavy computational processes. This goal was achieved by the tight association between personal assistance agents or service representative agents and their mobile device applications. In this framework, mobile devices play the role of mere interfaces to the Internet based agents. Most of the computations are performed by the agents to whom they are associated and by the other agents in the network with which they interact.

The technology concept was proved an implemented concrete application in which the proposed framework was successfully used.

In the near future we intend to use the developed technology for different scenarios such as virtual auctions and virtual enterprises applications. The exploration of other types of mobile devices such as pocket PCs and PDAs will also be considered.

Maybe more importantly, we will improve the complexity and sophistication of the agent layer between the service representative agents and the user personal assistance agents. This layer will be extended with powerful brokering capabilities (Botelho et al. 2003) developed in our group during the Agentcities project (Willmott et al 2001) and dynamic context-dependent service composition planning capabilities to be created in the forthcoming European IST project CASCOM.

Finally, we will work on the improvement of the personal assistance agent with the aim of making it more autonomous. One of the obvious directions is the possibility to learn user preferences through user feedback and also through collaborative filtering.

Although the implemented system is a simple testbed, it clearly proves the concept that, using current mobile computing technology, using mobile devices as mere interface Internet agents is an adequate approach to enable mobile device users to have access to agent technology and as a way of using homogeneous communication technology through out the system for agent interaction.

## 6 References

1. Bellifemine, F; Poggi, A.; and Rimassa, G. "JADE - A FIPA-compliant agent framework". CSELT internal technical report. Partially published in Proceedings of PAAM'99, 1999; pp.97-108.
2. Botelho, L.M.; Antunes, N.; Ebrahim, M.; and Ramos, P. "Greeks and Trojans Together". Proceedings of the Workshop "Ontologies in Agent Systems" of the Autonomous Agents and Multi Agent Systems Conference (AAMAS 2002), 2002.
3. Botelho, L.M.; Mendes, H.; Figueiredo, P.; and Marinheiro, R. "Send Fredo off to do this, send Fredo off to do that". Proc. of the International Workshop on Cooperative Information Agents (CIA2003), p152-159, 2003
4. Carrega, D.; Fournier, E.; Hope, S.; Muyal, H. "Delivering Integrated Voice and Data Services for Mobile Customers with the MOVE Middleware Architecture", 1999
5. Chlamtac, Imrich; Redi, Jason; "Mobile Computing: Challenges and Potential"; The University of Texas at Dallas and Boston University. Encyclopedia of Computer Science, 4th Edition, International Thomson Publishing, 1998.
6. Foundation for Intelligent Physical Agents. "FIPA Agent Management Specification". Document SC00023J, 2002. <http://www.fipa.org/specs/fipa00023/SC00023J.html>
7. Foundation for Intelligent Physical Agents. "FIPA Communicative Act Library Specification". Document SC00037J, 2002. <http://www.fipa.org/specs/fipa00037/SC00037J.html>
8. Foundation for Intelligent Physical Agents. "FIPA SL Content Language Specification". Document SC00008I, 2002. <http://www.fipa.org/specs/fipa00008/SC00008I.html>  
Gateway Group, Offshore Development. 2003. <http://www.gatewaytechnolabs.com/en/offshore/advgateway.asp?lid=222>
9. Kawai, H.; Akamine, S.; Kida, K.; Matsuda, K.; Fukushima, T. "Development and Evaluation of the WithAir Mobile Search Engine", 2002. <http://www2002.org/CDROM/poster/102.pdf>
10. Landay, James A.; Kaufmann, Todd R.; "User Interface Issues in Mobile Computing"; School of Computer Science - Carnegie Mellon University. Appeared in the Proceedings of the Fourth Workshop on Workstation Operating Systems, Napa, CA, October 1993.
11. Tomarchio, Orazio; Vita, Lorenzo; Puliafito, Antonio; "Normadic users' support in the MAP agent platform" Dipartimento di Ingegneria Informatica e Telecomuazioni, Università di Catania and Dipartimento di matematica, Università di Messina, 2000.
12. Vodafone, "perto de mim". A portuguese mobile phone service, 2003. <http://www.vodafone.pt/main/funandinfo/localizar/Pertodemim/default.htm>
13. Willmott, S.; Dale, J.; Burg, B.; Charlton, P; and O'Brien, P. 2001. "Agentcities: a world-wide open agent network". Agentlink News, 8:13-15
14. Wooldridge, Michael; Ciancarini Paolo; "AGENT-ORIENTED SOFTWARE ENGINEERING"; Dept of Computer Science, University of Liverpool and Dipartimento di Scienze dell'Informazione, University of Bologna. Handbook of Software Engineering and Knowledge Engineering Vol. 0, No. 0 (1999)