# A Context-Aware Solution for Personalized En-route Information through a P2P Agent-Based Architecture

José Santa, Andrés Muñoz, and Antonio F. G. Skarmeta

Department of Information and Communications Engineering
Computer Science Faculty
University of Murcia
Campus de Espinardo, 30071 Murcia, Spain
josesanta@dif.um.es|amunoz@um.es|skarmeta@dif.um.es

**Abstract.** Communication technologies and the management of context aware information are two key researching lines in the intelligent transportation domain. A suited network infrastructure which allows Vehicle to Vehicle (V2V) and Vehicle to Infrastructure (V2I) communications, on one hand, and the provision of location-based information adapted to the driver, on the other hand, comprise two important components in vehicle telematics field. Following this idea, our work includes a communication infrastructure based on Peer to Peer (P2P) networks which deals with V2V and V2I requirements. Reasoning agents have been integrated in the designed network, so the vehicle and the infrastructure's entities act as agents which adapt the information about the traffic area according to a driver profile. The paper describes also a prototype which implements both the vehicle edge, using an adapted vehicle, and the road edge, developing the network infrastructure and the knowledge management system. A performance study of the inference of contextual information has been added as well.

## 1 Introduction

Communications are essential in the current information society. Practically everyone can connect to Internet at work or at home, and even in mobility environments, using different kinds of portable devices. Cars are other places where the user stays during long periods too, so connectivity here is a new valuable field. However, apart from the direct usage of the Internet, some on-board services need specific communication primitives. The provision of context aware information to the driver is a key item at this point, where the synergy between the vehicular networks, the satellite based location technologies, and software agents that manage knowledge systems enable a new researching field.

Considering the connectivity requirements of services implemented on board vehicles, two types of communication necessities could be summarized: communications between the vehicle and the infrastructure, and communications among vehicles. In the first case, on-board services require a local connection with the infrastructure located at the road side. This kind of connectivity is usually named as Vehicle to Infrastructure (V2I) communications. Some electronic fee collection systems use, for example, DSRC gantries to keep the path followed by a vehicle in order to calculate the associated cost. Here, the main technologies used are DSRC, infrared, and wireless LAN. The second group of such services are the vehicle to vehicle applications. Here, the most extended term is Vehicle to Vehicle (V2V) communications. Solutions for collision avoidance, and warning mechanism in general have been developed. For this type of services, the most extended technology is the ad-hoc networks applied to the vehicle field (VANET or *Vehicular Ad-hoc NETworks*). As can be read in the rest of the paper, the solution presented here proposes a communication architecture which deals with the these connectivity requirements. Hence, services with V2I and V2V necessities are considered in our proposal. Cellular networks are used, jointly with peer to peer (P2P) agent networks, to design a communication system based on coverage areas to reach such as architecture.

Using a suitable communication technology, vehicles could be connected to an information system which provides interesting notifications about the current status of a traffic zone. This kind of applications are common in mobile phones, where the current cell in the network indicates the position of the user, and therefore commercial advertisements about interesting places around can be sent to the mobile device. This model can be applied to the vehicle field. Basically, using a GPS device, the vehicle could be connected to a server which provides it with location based information about interesting places, for example. This approximation, however, does not take into account the user's preferences. Here is where the user's profile is important. If the system is able to adapt the information which is sent to the vehicle according with her profile, the driver would receive only the information she is interested in. In order to design a system with such features, not only a suitable knowledge representation model is necessary, but also an efficient inference technique is mandatory. The inference task must be carried out quickly, because now we are dealing with vehicles, that obviously move faster than a mobile phone carried by a person. As a result, the context information is updated more frequently.

Our work integrates a knowledge management model based on ontologies, using OWL (*Ontology Web Language*) modeling, to represent the information about interesting places in the traffic zones and the driver's profile. The ontology model allows the inference of interesting information about the zone according with the driver's profile, thanks to the formal logic language used in the description of interesting places or services of each zone and the driver's profiles. All the reasoning and data storage about interesting places and profiles are integrated in our communication architecture by means of the agents. Thus, as it is exposed along the paper, the data processing and data storage are distributed across the system.

The rest of the paper is organized as follows. In Section 2, the presented system is placed in the current researching world about vehicle communications and knowledge management. Section 3 explains the designed communication architecture. Section 4 describes the integrated system for providing location aware information to drivers. The Section 5 serves to show the developed prototype. Finally, Section 6 contains the conclusions of our work.

## 2 Related Work

Regarding to vehicular communications, as it has been stated previously, the researching lines are centered in the usage of ad-hoc networks in the VANET field. The interest of this work lies in proving how cellular networks can be suitable not only for a direct Internet access, but also for enabling V2V and V2I communications [1]. In [2], a communication technique between the vehicle an the road infrastructure based on the DSRC technology is presented. This illustrates an example of a V2I communication method. The main drawback here is the cost of deployment at the road edge, which requires the installation of multiple DSRC devices, in contrast to our approach, based on cellular networks. In this line, [3] presents a handover system for wireless LAN networks in vehicles. Although the idea could be valid for urban or fixed railway zones, the deployment is limited by the cost. A work which captures a P2P approach through JXTA to notify events to the road side is shown in [4]. Our system use the JXTA technology as well in the vehicle field, however it also includes an architecture valid for V2V communications.

Delivering contextual information to the driver is the other main issue of the paper. The novelty of the system presented here is the inference of contextual information regarding to the driver's profile. Several approaches have been presented in other environments. Chou et al [5] develop a context-aware museum tour guide, in order to adjust its recommendations about the museum's items to the visitor's individual context. Similarly to us, they build the tour guide based on Semantic Web technologies such as OWL and web rule inference engines, which reduce the maintenance costs and the addition of new information. The rules are expressed in ROWL, a extension of OWL, whereas we use the Jena format for expressing our rules. A similar idea, but centered in touristic city tours, can be found in [6]. In this work, the mobile devices gather contextual information (location, preferences. . . ) about the tourist and then it is sent to specific servers that process that information. These servers are Profile Management Server, for dealing with the tourist's preferences, and Location Server, that determines interesting touristic places on the tourist's location. The mobile devices use WiFi or GPRS as communication mechanism, whereas the information is represented using a own language based on XML. We advocate for the use of OWL as it is a W3C standard. Another work in this line is [7], changing the scenario to buildings and indoor spaces. In the vehicle field, [8] comprises a multi-purpose management system for contextual information. The system is exposed as valid for the vehicle domain and allows queries about the traffic conditions and route guidance, however it only presents an application for the urban environment, and does not adapt the retrieved information to the user preferences.

# 3 V2V and V2I Communications

## 3.1 A Multi-purpose Communication Infrastructure

It has been stated that cellular networks are the appropriate technology in order to use an unified communication mechanism. That is, cellular networks represent the communications capabilities which allow the implementation of any on-board service with connectivity requirements. However, an important issue in the service development arises when the amount of applications in the vehicle compartment grows. [10] works on this problem, and proposes a platform for the development of vehicle services based on a general purpose on board unit (OBU). A multi-platform system based on OSGi (Open Service Gateway initiative) is used for the deployment of services implemented as software. In the vehicle telematics field, a large number of services need communication capabilities. This is the reason why the work presented in [10] is extended with a middleware which allows the use of a high level communication interface. Using this mechanism, each new service does not require the implementation of its protocol starting from the TCP/IP basis. This common interface allows the message exchange in both V2V and V2I environments.

Using a P2P approach over cellular networks, the vehicle can receive and send contextual information about its current environment. Fig. 1 shows a general diagram of the high level communication architecture proposed, with the Group Server and Environment Server agents. The traffic zones are organized in coverage areas, each one of them using a different P2P communication group. Vehicles are able to move from one P2P group to another through a roaming process between coverage areas. This roaming is based on the location of the vehicle, provided by the GPS sensor. Information about areas is received from the Group Server using a TCP/IP link over GPRS/UMTS. The Group server stores the geometry of every service area. A local element called Environment Server manages special events inside the area. Event notifications are sent and received by service edges, located either at the vehicle or at the road side (Environment Servers). All messages emitted are encapsulated in P2P packages. Two different techniques of emission have been developed, so a P2P message can be broadcast in the area or sent to a specific vehicle.

Apart from the designed architecture, in Fig. 1 the three most representative scenarios are exposed from left to right. In the first one, a vehicle is passing from one area to another. The Group Server provides the P2P parameters to maintain the communication in all the active services. Because the Group Server sends the area geometry to the vehicle, the latter has to communicate with the central entity only when it is necessary, saving communication resources. In the second scenery, a safety service located in the vehicle notifies a repair event. This event is broadcast, so all vehicles located in the area receive the warning. Because the Environment Server also receives all the broadcast messages, it can process any event which requires a special treatment, such as a collision, for example. Messages from special services (repair, collision...) are forwarded by the Environment Server to the adjacent areas, in order to improve the warning

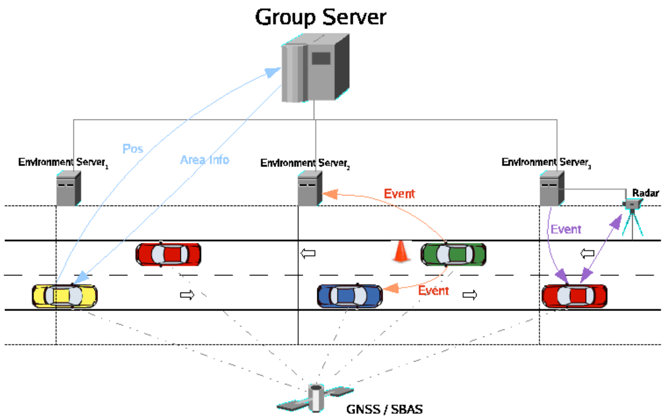


**Fig. 1.** High level communication architecture

mechanism. The last scenery in the diagram shows how the Environment Server can be connected to the rest of the road side infrastructure, which may be composed of speed radars, identification sensors, video cameras, etc. Thanks to this, local events can be notified to the vehicles in the area, and useful reports can be sent to the central entity.

### 3.2 Protocol Details

It is clear that the use of several protocols among the entities described in the previous section is necessary in order to perform all the processes that have been explained. To understand the operation of the whole system, the messages exchanged are divided into the three existing protocols detailed in the next lines.

Table 1 includes the messages exchanged between the vehicle (V) and the Group Server (GS). The communication is carried out by a message exchange over TCP, using the cellular network interface directly. The first process between these two entities is the management of the roaming. An initial *Area_Update_Query* message is used by the vehicle for registering itself against GS. The *Disconnect* message serves for ending the communication. The roaming process uses the *Area_Update_Query* and *Area_Update_Response* messages. The first one is used when a vehicle detects the end of its current coverage area. In that moment, it sends a new *Area_Update_Query* message with its new location. GS replies with an *Area_Update_Response*. This message contains the available services in the zone and the P2P parameters to allow subscribing to them. When the user wants to use a defined service, she sends an *Event_Subscription* with the services she is interested in. It is important to note that every service has a different P2P group in every area. This allows for distinction among coverage zones.

Regarding to the communication between the Environment Server (ES) and GS, Table 2 summarizes all the interchanged messages. The *ES_Registration*

**Table 1.** Vehicle-Group Server communication

| Purpose | Messages | Sender |
|---|---|---|
| Communication | Area_Update_Query | V |
| Management | Disconnect | V |
| Roaming | Area_Update_Query | V |
| Process | Area_Update_Response | GS |
| Event | Event_Subscription | V |
| Subscription | | |

**Table 2.** Environment Server-Group Server communication

| Purpose | Messages | Sender |
|---|---|---|
| Communication | ES_Registration | ES |
| Management | ES_Registration_OK | GS |
| | ES_Registration_ERROR | GS |
| | ES_Registration_Update | ES |
| | ES_Registration_Update_Response | GS |
| Message | Neighbour_Groups_Query | ES |
| Propagation | Neighbour_Groups_Response | GS |

message is used by the ES to connect with GS. This packet contains the location of ES, so GS is able to search for the P2P parameters of all services available in the zone through the *ES_Registration_OK* message. If any error occurs, GS sends a *ES_Registration_ERROR*. This packet includes the reason of the error. Because the Environment Servers are static entities, which may be connected during long time periods, they need to update its configuration periodically. The *ES_Registration_Update* message performs this task. It only asks for the P2P parameters of the coverage zone. GS answers with this information using the *ES_Registration_Update_Response*. The second group of messages are used in the message propagation mechanism. Sometimes it is necessary to ensure that all vehicles near the problem receive critical events. The location of such problem may be close to the border of the coverage area. Consequently, it is necessary to propagate the associated message to the neighbouring areas. In order to perform this task, the Environment Server, which listens to all events, uses the *Neighbour_Groups_Query* message to ask for the neighbouring P2P groups for the service in question. GS replies with this information through the *Neighbour_Groups_Response*. Now, ES can forward the message to the neighbouring areas.

Finally, Table 3 includes the three messages used by the services to send messages. The first message we can see is *Vehicle_Event*. It contains a message sent from a vehicle service edge. All vehicles located in the area, and the Environment Server, receive these messages. The *Environment_Event* messages are sent by the Environment Servers. This is used, up to now, by the message propagation method described previously. The last packet, *Specific_Environment_Event*, allows for sending a message to a specific vehicle.

**Table 3.** Vehicle-Vehicle/Environment Server communication

| Purpose | Messages | Sender |
|---|---|---|
| Message Passing | Vehicle_Event | V |
| | Environment_Event | ES |
| | Specific_Environment_Event | ES |

# 4 Managing Context-Aware Information for Road Services

## 4.1 Information Management Through Ontology Modeling

The importance of context is receiving considerable attention in combination with mobile devices as these devices are particularly affected by environmental changes. A general motivation is that context-awareness can serve to compensate for the abstraction that is required in the first place to make systems accessible in changing environments and situations. Several advantages are acquired by modeling these kind of domains with ontological languages such as OWL. First, the information represented by the model can be easily and broadly exchanged among heterogeneous applications that understand this language. Another fundamental capability that is provided by adding semantics is the inference process. This process allows reasoning about the model, giving as a result the extraction of new knowledge that was implicit in the domain. This inference process is achieved by triggering some predefined set of rules about class hierarchy, types of properties, etc. Taking this idea further, we can define our own rule set that expresses our preferences about the behaviour of a semantically modeled application in each moment. Thus, this application will work according to the user's contextual situation.

Therefore, to deal with this type of scenario that involves context, location and preferences, we are going to make use of ontologies in modeling the user's profile and her context, the tracking location system explained in the previous section, and a set of special rules called *context rules*. Context rules model the user's preferences and they will have influence about the behaviour of an application depending on the user's context, for example the user's location, time of the day, relations with other users, etc. In this way, the same application may derive in different activities according to the context rules that are triggered when executing the application. These rules are formed by ontological elements, as it is explained in [11].

## 4.2 Ontology Modeling Adapted to the Generation of Context Aware Information in a Large Scale Environment

We have augmented semantically our P2P architecture by means of modeling the context aware information and the driver's profile using different ontologies that have been developed in order to satisfy the requirements of the system commented along the paper. One of these requirements is its distributed composition. Consequently, the ontologies are spread among the different components of the architecture. Before explaining how this distribution has been designed, it is important to distinguish between the scheme and the instances of an ontology. Whereas there is only a scheme that represents the structure of the domain modeled, there could be various instances of that scheme for the concrete information captured. For example, the Profile ontology contains a scheme with the different preferences of a driver about restaurants, hotels, etc., and there is (at least) one instance of this scheme per each driver.

Going back to the distribution issue, the Profile ontology (scheme and instances) is stored in the GS. The possibility of that each car manages its own profile was studied, however due to performance results and the heterogeneous capabilities of the on-board units, we finally decided to move all profiles to the Group Server. In any case, we will keep on evaluating these two options and they will be present in our future work. By indicating the vehicle identifier, the GS returns the driver's profile, if it exists. On the other hand, the Service ontology (Fig. 2) is stored among the Environment Servers. This ontology models the context information of each area, i.e., the elements that could be of interest to the users. Each ES maintains the same schema, but the instances are specific for each zone controlled by them.

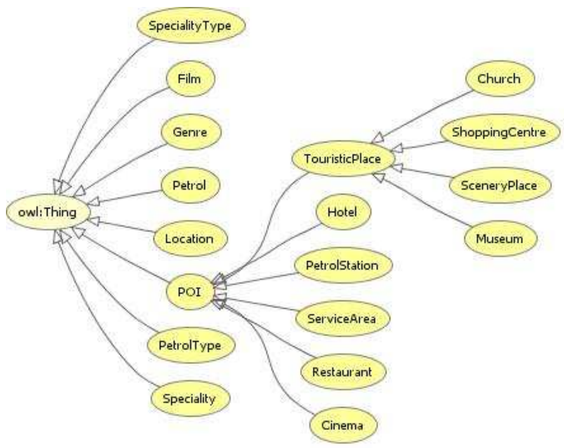


**Fig. 2.** Service Ontology in a hierarchical diagram

By combining the Profile ontology with the specific Server ontology of each ES, we obtain enough information to perform the inference process. Now, in order to incorporate the driver's preferences to the model, we make use of context rules. In Fig. 3 an example of these rules is shown. Concretely, this rule infers the restaurants which menu price is between a maximum *(le(?mp,?maxmp))* and minimum *(ge(?mp,?minmp))* value stated by the driver. If this condition is hold by any restaurant, it is added to the list of driver's profile matches, specifying which condition is satisfied. Finally, the reasoning process is performed in the ES, as it will be explained later. One advantage of using context rules is that they can be used with any driver's profile or any specific instance of Service ontology. As a result, the same rules can be applied in any ES. Moreover, since the rules are defined in a external file to the source code, recompiling is avoided in case that we need to add new behaviours or change them.

```
# Check menuPrice
[menuPrice:
                (?profile rdf:type prf:Profile)
                (?profile prf:restaurantProfile ?rp)
                (?rp prf:maxMenuPrice ?maxmp)
                (?rp prf:minMenuPrice ?minmp)

                (?res rdf:type srv:Restaurant)
                (?res srv:menuPrice ?mp)

                le(?mp,?maxmp)
                ge(?mp,?minmp)
                ->
                (?profile prf:matches ?res)
                (?res srv:matchesWith 'menuPrice')
]
```

**Fig. 3.** A context rule for matching the driver's preferred menu price

In Fig. 4 a sequence diagram of the proposed system is shown. As it has been stated, the method by which the Environment Server detects the location of the vehicle is independent from the inferring and the knowledge model. In this approach, a RFID (*Radio Frequency IDentification*) system is used to detect a vehicle in certain locations. All the messages involved in the process of contextual information inferring are shown in the sequence diagram as well. First, the vehicle is detected by the RFID subsystem at a specified location. When the RFID reader detects the vehicle, it sends a *Reader_Notification* message. In this message, the RFID subsystem sends the vehicle identifier, so the Environment Server can use this information to search for the driver's profile. The message *User_Profile_Query* is addressed to do this task. The Group Server looks for the driver's profile in its data base and, if the profile does not exist, it replies with a *User_Profile_Not_Found*. In other case, if the profile is found, the Group Server replies with a *User_Profile_Response*. This message contains the driver's OWL profile. Now, the Environment Server processes the local information about the area, together with the driver's profile, to infer the interesting notifications in the zone. When this information is computed, a *Specific_Environment_Event* message is sent to the vehicle in question. The service edge at the vehicle is in charge of showing this information in an appropriate way.

## 5 Prototype and Tests

The system described in the previous sections has been implemented in a real platform. All technological details about the prototype are explained in this

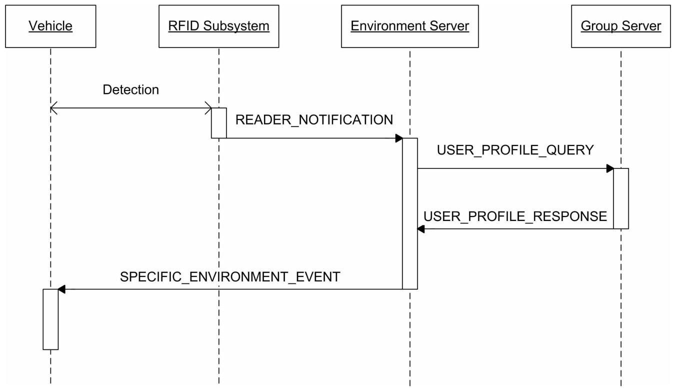


**Fig. 4.** High level communication architecture

section. The explanation is divided into the vehicle edge, the road side equipment used, and the software implementation at the road side and, mainly, at the vehicle edge.

### 5.1 Test car and Road Side Equipment

The vehicle used for the system deployment is a vehicle widely sensorized at the University of Murcia. In [12] it is explained in detail all the hardware installed in the vehicle. The GPS sensor is utilized to know the position of the car and, therefore, it is essential for the roaming process among the coverage areas. The car in question is shown in Fig. 5. The on-board unit is a SBC (*Single Board Computer*) of VIA. The interface between the user and the OBU is carried out by a LCD panel installed on the dashboard. The operating system used in the OBU is a Linux Fedora Core 4, and a Java Virtual Machine 1.5 enables the computer to support all the multi-platform software implemented. In [10] it can be found more details about the software platform used in the OBU. Regarding to the communications, a cellular network PCMCIA transceiver has been used. The model is a Novatel Wireless Merlin U530, which allows GPRS and UMTS data connections.

At the road side, the tests have been carried out using a set of areas defined over the campus of the University of Murcia, as well as several kilometers of a close motorway. Eight areas has been used to probe the communications architecture. The Environment Servers have been implemented as software entities, running in a server located in one of our laboratories. The Group Server runs over another computer. All of them are Linux based systems, with a common Intel processor. A RFID system has been used to test the provision of contextual information. Using a tag located on the windscreen of the car, the reader can detect the presence of a vehicle. The tests in this case have been carried out in an open environment located at the campus of the University of Murcia.

**Fig. 5.** Prototype vehicle used in the development

### 5.2 On-Board Software Implementation

A new service has been implemented at the vehicle side to probe the feasibility of the communication architecture presented. JXTA (JuXTApose) [13] has been the P2P technology used. JXTA presents a group based communication system with the features of any other P2P protocol. It forms an underlying network which abstract the user from the internal details. Fig. 6 shows the Message Console service. Using the P2P middleware, an event based mechanism has been created to notify traffic incidences. Each "event" has been modeled like a service in the communications system. Eight types of events are available in the right box, so the user can subscribe to any of them. The most important service is named *On road information*. If any of the services is active, the corresponding events can be received or thrown. The events regarding to the on road information service cannot be thrown by a user, because it is managed by the road side infrastructure. All the received events are shown in the central part of the window.

In the example provided in Fig. 6, the driver has activated all the available services, but only three of them are available in the current area. An event regarding to the "On road" information service has been received. As can be noted, the driver's profile has been used in order to adapt some tourist information available in the current coverage area. This information has been printed in the window, showing the matches with the user's preferences. It is noticeable how *Guggenheim* matches with one hundred percent of the user likes, but the three other museums only match at a fifty percent.

### 5.3 Inference Performance

We are mainly concerned about how much time is employed at inferring and delivering interesting notifications to the car, since it could drive very fast and the notifications could arrive when leaving the corresponding zone. In order

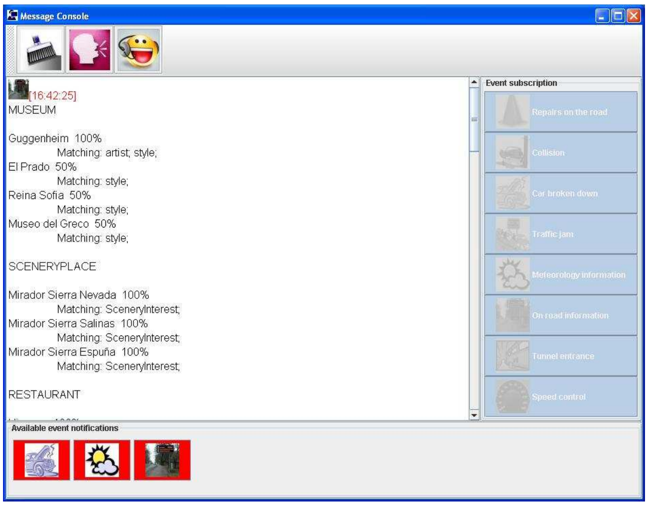


**Fig. 6.** An screenshot of the vehicle application

to evaluate the system performance, we have developed a model that contains enough information to deal with real situations and with all possible profiles. This model is composed of eleven and eighteen ontology concepts in the Profile and Service ontology, respectively. A Environment Server has been set up to simulate the zone where the car is driving along. This server contains one hundred and ten instances of different Service concepts, such as restaurants, cinemas, etc. Moreover, thirty-two context rules has been defined to match the driver's profiles with these concepts or services. Four different profiles have been proposed to measure the system responses. Each profile is centered on different types of services. Thus, Profile 1 is more keen on restaurants, Profile 2 on cinemas, Profile 3 on hotels, and Profile 4 is a mixture of these three. We have divided our performance test in three parts: First, the loading of the knowledge base (Service and Profile ontologies with the service instances and one driver's profile, together with the context rules) in the ES. Second, we measure the inference process, i.e., the application of the context rules over the knowledge base. Thirdly, the collecting matches task and the sending of information to the user is evaluated. The results are presented in Table 4. The ES employed is an Intel Pentium D 3Ghz with 1GB of RAM running under Linux, with the Jena [14] inference engine configured for the reasoning process.

The most expensive operation is the loading of the knowledge base, normally 2 or 3 seconds. However, since this process is performed once during the ES setting up, it does not affect to the normal operation of the system. More relevant are the results of the inference process and the collecting matches task, critical processes when a car is in the ES zone. As the profile is more complex, the inference process takes longer, but it still stays inside of acceptable levels. However, the collecting matches task is unpredictable and variable, depending on

**Table 4.** Evaluation results of the system performance

| Profile | Load Knowledge Base | Inference Process | Collect Matches |
|---|---|---|---|
| Profile1 | 2484ms | 250ms | 62ms |
| Profile2 | 2625ms | 281ms | 63ms |
| Profile3 | 2890ms | 313ms | 47ms |
| Profile4 | 3410ms | 381ms | 105ms |

the number of matches found. In principle, this is the faster operation, therefore the characteristics mentioned above do not represent a serious problem.

## 6 Conclusions

The paper exposes the work carried out in the synergy between the vehicular communications and knowledge management fields. A communication technique, based on cellular networks and the P2P protocol provided by JXTA, has been designed to offer the programmer a high level mechanism to use V2V and V2I communications in her on-board services. In addition, an integrated context-aware system to provide location-based information according to the driver's profile has been proposed. Semantic Web technologies as OWL have been selected to represent the information of the model, and an inferring process is followed to adapt information about the traffic zone according to the driver's profile. Finally, the designed prototype exposes the feasibility of he whole system.

In the future, several tests are planned to be made with the aim of testing the performance of the communication architecture in contrast to ad-hoc network solutions. Even though, this proposal is considered more general than ad-hoc ones, because it has a V2V message passing scheme based in P2P networks which also includes V2I capabilities. Another future line of work will be the evaluation of moving the driver's profile from the GS to the vehicle's OBU, in order to study the effects produced in the system's performance and reliability. This would offer the driver the possibility of changing the profile information dynamically.

**Acknowledgments.** The authors would like to thank the Spanish Ministerio de Educación y Ciencia and Fundación Séneca for sponsoring the research activities under the grants AP2005-1437 and 04832/BPS/06, in frames of the FPU and 2003-2006 PCTRM programs, respectively. Thanks as well to the financial support given by the European Spatial Agency (ESA) under the GIROADS 332599 project, and to the Spanish Ministerio de Fomento for its continuous support in vehicular researching.

## References


1. Luisa Andreone, Michele Provera: Inter-vehicle communication and cooperative systems: local dynamic safety information distributed among the infrastructure and the vehicle as "virtual sensors" to enhance road safety, *ITS Hannover 2005*, Hannover, January 2005.
2. Gen Hattori, Chihiro Ono, Satoshi Nishiyama, Hiroki Horiuchi: Implementation and Evaluation of Message Delegation Middleware for ITS Application, *2004 Symposium on Applications and the Internet-Workshops (SAINT 2004 Workshops)*, Tokyo, January 2004.
3. Toshiya Okabe, Takayuki Shizuno, Tsutomu Kitamura: Wireless LAN Access Network System for Moving Vehicles, *10th IEEE Symposium on Computers and Communications (ISCC05)*, La Manga del Mar Menor, June 2005.
4. Luciano Baresi, Carlo Ghezzi, Antonio Miele, Matteo Miraz, Andrea Naggi, Filippo Pacifici: Hibrid service-oriented architectures: a case-study in the automotive domain, *5th International Workshop on Software Engineering and Middleware (SEM'05)*, Lisbon, September 2005.
5. Chou, S., Hsieh, W., Gandon, F. L., and Sadeh, N. M.: Semantic Web Technologies for Context-Aware Museum Tour Guide Applications, *19th international Conference on Advanced information Networking and Applications*, Washington, DC, March 2005.
6. Juan Ignacio Vázquez, Ana Belen Lago, Iker Larizgoitia, Ignacio Uria and Rafael Bañales: WALLIP: Una plataforma de servicios con percepción contextual basada en localización, *Ubiquitous Computing and Ambient Intelligence, CEDI 2005*, Granada, September 2005.
7. José M. Falcó, Roberto Casas, Álvaro Marco, Jorge Falcó: Using GUIA to "Wayfindig" in unkown builidings, *Ubiquitous Computing and Ambient Intelligence, CEDI 200*, Granada, September 2005.
8. René Meier, Anthony Harrington, Vinny Cahill: Towards Delivering Context-Aware Transportation User Services, *2006 IEEE Intelligent Transportation Systems Conference (IEEE ITSC 2006)*, Toronto, September 2006.
9. Tomas Nolte, Hans Hansson, Lucia Lo Bello: Wireless Automotive Communications, *Euromicro Conference on Real-Time Systems (ECTRS 05)*, Palma de Mayorca, July 2005.
10. José Santa, Benito Úbeda, Antonio F. G. Skarmeta: A Multiplatform OSGi Based Architecture for Developing Road Vehicle Services, *Consumer Communications & Networking Conference 2007 (CCNC 2007)*, Las Vegas, January 2007.
11. Andrés Muñoz, Antonia Vera, Juan A. Botia, Antonio F. Gomez Skarmeta: Defining basic behaviours in ambient intelligence environments by means of rule-based programming with visual tools, *1st Workshop of Artificial Intelligence Techniques for Ambient Intelligence*, Riva de Garda, August 2006.
12. Skarmeta A.G., Martínez H., Zamora M.A., Úbeda B., Gómez F.C, Tomás L.M.: MIMICS: Exploiting Satellite Technology for an Autonomous Convoy, *IEEE Intelligent Systems*, N IV. V. vol. 17, pp. 85-89, July 2002.
13. JXTA web site. *http://www.jxta.org*
14. Jena web site. *http://jena.sourceforge.net*