Proactive Services in a Distributed Traffic Telematics Application

Nils Gura[1], Albert Held[2] , Jörg Kaiser[3]

1) Sun Microsystems Laboratories, Nils.Gura@sun.com
2) DaimlerChrysler Ulm, Albert.Held@daimlerchrysler.com
3) University of Ulm, Kaiser@informatik.uni-ulm.de

***Abstract**: The paper describes a mobile application for traffic telematics, based on the Jini middleware. The application is realized as a set of user-level services, characterized by following properties: The services should be proactive, i.e. they act on behalf of the user based on the user's preferences. Secondly, context information derived from floating car data is used to improve the quality of the proactive service and thirdly, mechanisms are introduced to cope with the inherent unreliability of the wireless communication channel. The paper also briefly presents some basic measurements on packet transfer rates over a wireless IEEE 802.11 communication link.*

***Keywords:** proactive services, traffic telematics, mobile computing, wireless networks*

1. INTRODUCTION

Wireless technology providing mobile access to information and communication services is playing an increasingly important role. Accompanying this trend, a transition of end user devices from a small number of PCs and laptop computers to numerous small, yet powerful devices such as mobile phones, PDAs and embedded devices is taking place. In the automotive sector, traffic telematics systems built into vehicles will supply passengers not only with up to date traffic information, but also offer a variety of convenience and safety services. In this paper, we describe a component based software architecture supporting proactive applications and building a link between lower-level sources of information and the end user in a mobile environment.

Up to now, information and services are provided by centralized mechanisms through mobile telephone links. In the future, however, information will be generated, distributed and shared between vehicles using short-range wireless communication and ad hoc links in multi-hop networks. Services will be offered in a completely decentralized way in a network of service consumers and service providers which dynamically may change their roles. Moreover, higher level user-centric services will recursively use all kind of lower level services eventually exploiting useful information derived from deeply embedded sensors [1]. Already available in vehicles today, a large number of different sensors generate a steady stream of data giving information on rain, traction, braking, speed, position, acceleration etc. These sensors can serve as distributed sources of context information [2], [3] to be shared with other vehicles. Exploiting this floating car data as context information for proactive services is one of the key issues in our work.

Current communication and information equipment is often demanding significant attention from the user [4]. In order not to overwhelm the driver of a car who wants to benefit from these services, the user interface becomes a major concern. There are two main aspects of the user interface. One simply refers to the input and output devices used by the driver to interact with the system. The other aspect which is more in the focus of this paper is that a personalized proactive system assists the driver in using these services. The system uses a driver's profile to autonomously filter information, evaluating options and making decisions, thus reducing the continuous stream of information to a perceivable amount and presenting it to the driver in an adequate way adapted to his preferences. Assisting the driver, without sacrificing traffic safety by drawing his attention to the gadgets instead to the road, is the goal of the system. Proactivity of the user-level services exploiting context information is one of the keys to achieve this.

The paper is organized as follows: In the next section we briefly sketch the requirements of services in a car and motivate the proactivity of these services. Chapter 3 introduces our experimental application which is based on the Jini platform. The impact of mobility and the wireless communication is discussed in chapter 4 and a summary and outlook on future work concludes the paper.

2. PROACTIVE SERVICES

The main contributions to proactivity in this paper is the definition of a user agent which closely interacts with a service discovery and lookup facility. Because Jini [5] was the middleware selected for the project, we were able to benefit from the respective Jini services. A Jini Lookup service is available in each vehicle offering a set of both, user-level services and lower level services collecting, enhancing and distributing information extracted from all kinds of embedded sensors. The low level services form the basis for context creation. To exploit these services in a proactive way, a user agent is designed to perform the following tasks:

1. User Agents keep track of available services on other vehicles. They are parameterized by user profiles and selectively choose according to the user's settings.
2. User Agents evaluate the lower level floating car data to create the context according to which decisions about services can be based.
3. User Agents provide a user-defined appearance of user-level services. Thus a single look-and-feel interface masks the various interfaces of different services as much as possible.

Point 1 is crucial to guarantee a certain privacy of the user. Normally the users have to expose their preferences to the service providers to allow the services to be adapted to their specific needs. The user agent can choose services selectively without disclosing these details of the user settings. Instead, the agent locally filters information adapts it to whatever is specified by the user.
The collection, filtering and use of floating car data is an emerging field of research [6,7]. Information, originating from the embedded sensors of a car are disseminated via wireless links. Apart from problems concerning the propagation and constraints for this data [7], the user agent filters and evaluates this information according to the specific requirements of the respective application. In chapter 3 we will present some examples for this agent functionality.
Although Jini services are accessible through a standardized interface, the user-level appearance of the service can vary widely. Therefore, as pointed out in 3, the agent should provide the services according to the user's preferences and the specific multi-modal interface of the local system such as speech, vision, tactile devices or various display options.
Of course, the user agent can beneficially combine these functions, e.g. exploiting context information like the car's light sensor to decide on the way of presenting data e.g. visually or by voice output.

3. THE EXPERIMENTAL APPLICATION

To evaluate the feasibility of our architecture in a first experimental application, we implemented a mobile traffic Road-Look-Ahead (RLA) service as shown in fig. 1. It enables the driver of a car to observe the traffic conditions on the road ahead, e.g. when locked behind a truck in a blind corner. The driver's own limited view is complemented by a remote camera service provided by the truck showing the road ahead.

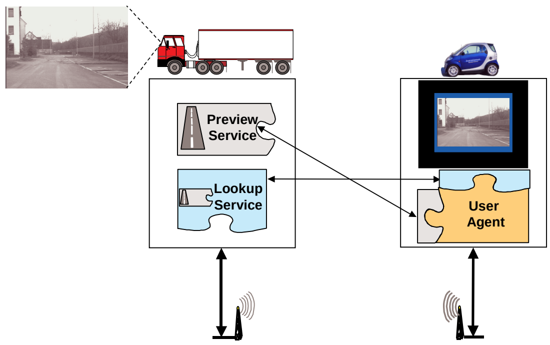

Figure 1: Road-Look-Ahead Service as an example of a Proactive Service

This RLA Service should comprise all relevant elements of a proactive service listed in the 3 statements above. It dynamically connects a car, e.g. a truck as shown in fig. 1 that advertises an image from the local telematics system. To select an image which may be offered by many cars in reach, the user agent has to perform a selection based on the context information like the speed of the car and the direction, thus filtering out cars which are far away or come from the opposite direction (although this may also be a function which a user may select). Fig 2 and 3 show the architecture of our system exploited for the RLA service where fig 2 sketches the server (producer) site and fig 3 the client (consumer)site respectively.
We implemented the system architecture on the basis of Java and Jini technology. The notion of a service is central to the Jini system. This gave us a number of advantages. Firstly, we were able to abstract and encapsulate low level context information like position,

direction and speed in a Jini location service which abstracts from all specific details of the navigation devices. Secondly, Jini provides a number of mechanisms which are related to mobility. Because wireless LANs connecting moving vehicles are inherently subject to varying bandwidth, transmission errors and loss of connection, services need to be fault tolerant with respect to delayed, missing or erroneous data packets. Jini mechanisms such as leasing, remote events, discovery and lookup support these requirements. An additional advantage of Jini is based on the property of Java, that the code of a service can be downloaded from a remote service provider and executed locally. This allows dynamic installation of services without involving the user.

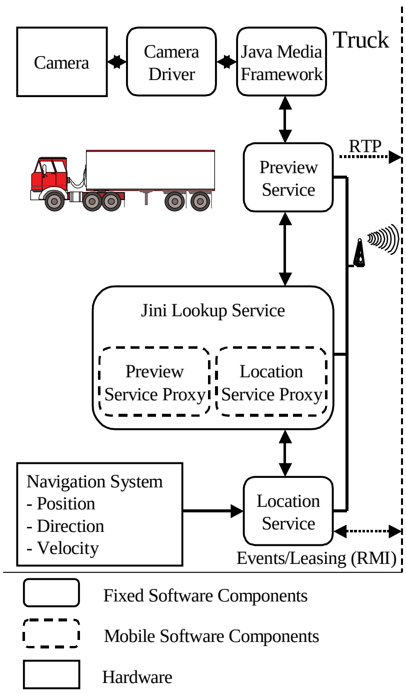

Figure 2: Components of the Look Ahead Service (producer side)

The system providing the service to the end user can be divided into hardware components (predominantly I/O devices), fixed software components and mobile software components (cf. fig 2). Hardware components are encapsulated in Jini services and can be accessed through a well defined and standardized interface. Exploiting the concept of Jini attributes, the services can describe their semantics and provide parameters for their execution. The fixed part of a service is bound to the host where it has been started. Upon startup, the service registers a Jini proxy in the local Lookup. Now, the service can be discovered and used by a remote client. To enable service usage by the remote client, the mobile code contained in the proxy can an be transmitted to the respective host.

On the telematics system of the truck, the RLA Service provides a video stream acquired through a digital camera. Bitmap images from the camera driver are sampled at a resolution of 176x144 pixels and at a frame rate of approximately 10 frames/s. As an appropriate middleware to support the necessary conversion of individual frames to a video stream, the

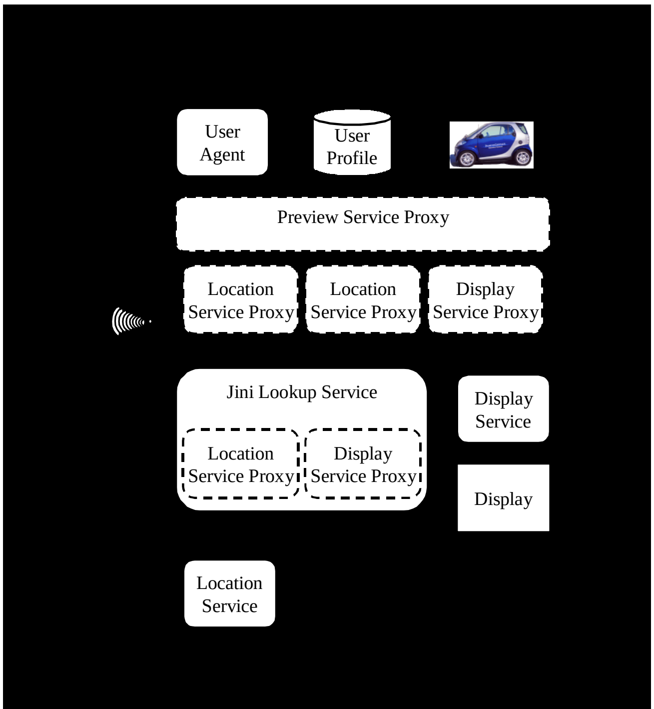

Figure 3: Components of the Look Ahead Service (client side)

encoding and decoding of the stream, the transmission over a wireless link and the adaptation to various input and output devices, we adopted the Java Media Framework (JMF) [8]. In our example, the H263 encoding scheme [10] was used to convert the sequence of frames to a compressed video data stream suitable for transmission over a wireless network. Further, we used the Real-time Transfer Protocol (RTP) which supports the transmission of continuous media data streams [9]. The RTP provides timestamps and sequence numbers to order frames but does not gives any temporal end-to-end guarantees. To provide these guarantees, mechanisms

have to be deployed on the MAC layer [11,12] as well as incorporating predictability measures in the operating system [13] which is beyond the scope of this work.
The Java Media Framework (JMF) as a middleware layer supports both, H263 and RTP encoding/decoding,. Besides capturing, encoding and decoding video frames the JMF was also used for presenting the image on the user's display. Using the mechanisms provided by RTP, the receiver can discard lost, erroneous or not timely received image data.
To be accessible by User Agents, the RLA Service registers its Jini proxy with the Lookup Service local to the truck. Data from each vehicles' navigation system is abstracted through a Location Service likewise locally registering itself with the Lookup Service. To be automatically updated with location data, user-level applications sign up for receipt of Location Event Messages which are distributed through Jini's remote event mechanism. The leasing concept of Jini, requiring the remote application to periodically confirm the service registration, is employed to remove unreachable or erroneous clients from the list of recipients on the provider side.
Encapsulating the car's display facility, a Display Service publishes its proxy through the local Lookup Service. User-level services can use this proxy to display their specific information independent of device specific features like resolution or colour representation

A User Agent runs on every user's local telematic system. It performs the tasks of 1. discover remote Lookup Services e.g. on another car and 2. to check for a RLA Service provided by the respective remote car's telematics system (fig. 3). Once discovered, downloaded and invoked by the User Agent, the RLA Service Proxy connects to the local and remote Lookup Service checking for Location Services to acquire and evaluate context information from the navigation systems. The application-specific context is composed of the relative position and proximity of the car (in front of vs. behind/on the side), its direction (same direction vs. opposite direction / coming from the side) and speed (coming closer vs. gaining distance). After confirming appropriate positioning between truck and car and establishing connection to the truck's counterpart of the RLA Service, the local Display Service is activated to display the camera image. The RLA Service process automatically exits after passing the truck or getting out of its reach. Note that all application specific code needed by the user's telematics system is provided by the truck, no software needs to be installed or maintained by the user. All context information necessary for the application is evaluated proactively. The user is not required to interact with the system and can focus on the traffic.

4. COMMUNICATION OVER THE WIRELESS LINK

As the experimental system, two SMART cars were available equipped with laptop computers running Windows2000 and WLAN 802.11 (System Aironet WLAN PC4800) with a maximum data rate of 11Mbits/sec. To provide the neccessary context information, access to a navigation system was possible to extract position, direction and speed. Additional sensors were accessible via the SMART's internal communication bus (CAN-Bus) but have not been used in the example application. The SMART gives us the possibility to test the wireless communication link in under realistic concerning e.g. bandwidth and disturbance by the movement of the cars. We measured a single sender and receiver, i.e. we had no effects because of network arbitration or contention. Certainly not specifically designed for this class of applications the wireless network performed fairly well at moderate speeds. Figure 4 and 5 show bandwidth measurements for both the UDP and the TCP protocol respectively. The measurements were taken from a vehicle passing a stationary receiver at

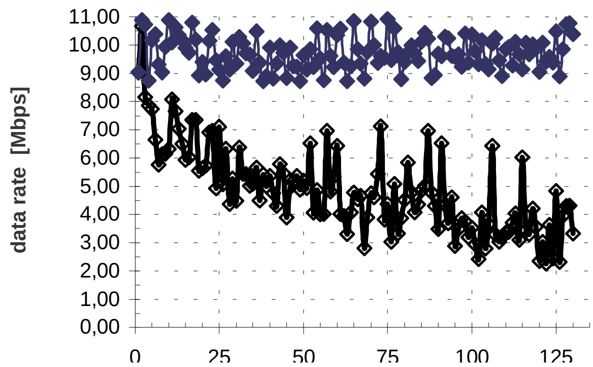

Figure 4: UDP transmission and reception data rate at 50 km/h

50 km/h. The distance to the receiver was always in the range provided by the Aironet WLAN PC 4800 equipment. Detailed measurements for two moving vehicles have not been made, but should confirm the results of this setup. Each measurement summarizes the bandwidth for the transmission of fifty 1kB packets. In the case of UDP, an unidirectional link was measured whereas roundtrips were considered for TCP. Transmission errors on the physical layer apparently lead to rejected packets and a significantly reduced reception

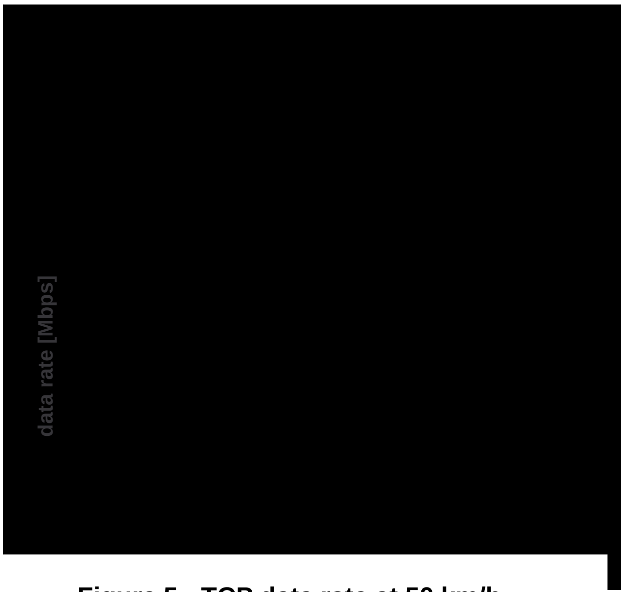

Figure 5: TCP data rate at 50 km/h

data rata. Approximately only half of the packets can be received correctly using UDP. For TCP, the same effects lead to retransmissions and an overall reduction of the transmission rate. The usable bandwidth is comparable for both protocols. It should be noted that the figures 4 and 5 present a series of subsequent measurements. Therefore, the collapse of bandwidth in fig. 5 is due to a local disturbance. As a result of the measurements, we can expect an average bandwidth which is well over 1 Mbit/sec which is above the necessary data rate to transfer the image of the look-ahead service. However, we observed a quality of the transferred images which have been below the expectations gained from the transfer rates. Partly this was attributed to the fact that because of lost packets and retransmissions subsequent packets are queued and therefore transferred with a large latency. Secondly, the operating system (Windows2000) does not give any temporal predictability when a packet is delivered to the application. This results in further latencies depending on the load of the system. Additionally, due to delays in receiving location events (which are used to confirm the service) and the expiration of leases, which could not be confirmed in time, the video transfer was interrupted. In summary a discontinuous presentation of the transmitted image was observed despite an average data rate well above the image data rate of less than 1 Mb/s.
Considering a safety-critical application like the RLA Service, the services provided by a wireless TCP/UDP/IP network are insufficient. However, we expect that at this end technological advances driven by car manufacturers and service providers will substantially improve the quality and reliability of the physical layer.

5. CONCLUSION AND FUTURE RESEARCH

The paper presents a traffic telematics application for a wireless network. It describes the service architecture and an example of a proactive system designed for mobile applications. Proactivity is a crucial property of a service to be usable in a car. Instead of the driver, the system has to decide on which service it should present to the driver in what form. To do this, it uses two sources of information: the preferences of the driver and the actual context information. We presents an architecture for such proactive services and introduce a user agent which collects, filters and presents services using the available preferences and context information. The context information is derived from the embedded sensors available in the car of today, which is disseminated to other mobile entities via wireless links. To cope with mobility and unreliable communication links, clients have to confirm their interest in a service periodically. However, at the moment, the system has just a best effort strategy. As is sketched in the paper, mechanisms have to be devised which better exploit the available bandwidth. Well known Quality of Service QoS mechanisms ca be used in a first step. These include mechanisms to drop outdated packets and reserving and synchronizing operating system resources.

Future research will be conducted in the area of improving the predictability of the communication and co-operation between mobile entities. To provide predictability in systems like the one described in this paper, concepts have to go well beyond the research in conventional QoS. Presictability has to incorporate not only resource reservation but also to a large extent resilience against failures and adaptation to a continuously changing environment. The research will partly be carried out in the European project CORTEX (Co-Operating Real-Time sEntient objects: architecture and experimental evaluation, IST-2000-26031).

6. REFERENCES

[1] D. Tennenhouse, “Proactive Computing”, *Communications of the ACM*, May 2000, p. 43-50

[2] B. N. Schilit, N. Adams and R. Want, “Context-Aware Computing Applications”, *IEEE Workshop on Mobile Computing Systems and Applications*, December 1994

[3] A. Schmidt, M. Beigel and H.-W. Gellersen, “There is more to Context than Location”, Computers & Graphics Journal, December 1999, p. 893-902

[4] M. Weiser and J. S. Brown, “The Coming Age of Calm Technology”, tech. report, Xerox PARC, October 1996

[5] SUN MICROSYSTEMS INC. “Jini Specification v1.0.1, November 1999, http//:www.sun.com/jini/

[6] Fleetnet: http://www.ccrle.nec.de/adhoc.html

[7] W. J. Franz “Reduzierung der Anzahl der Übertragungen in „Floating Car Data“-Diensten,

Proc. Kommunikation In Verteilten Systemen (KIVS) 2001, Hamburg, March 2001
[8] SUN MICOSYSTEMS INC. "Java Media Framework API Guide, 2000, http://java.sun.com/products/java-media/jmf/
[9] H. Schulzrinne, S. Casner, R. Frederick and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", Internet Engineering Task Force (IETF), Januar 1996
[10] H263 Video Coding Page: "http://www-mobile.ecs.soton.ac.uk/peter/h263/h263.html"
[11] J. Kaiser, M. A. Livani, W. Jia "Predictability of Message Transfer in CSMA Networks", Proc. 4th International Conference on Architectures for Parallel Processing – ICA3PP, Hong Kong, December 2000
[12] M. Mock, E. Nett, S. Schemmer "Efficient Reliable Real-Time Group Communication for Wireless Local Area Networks", Proc. Dependable computing - EDDC-3 / Hlavicka, Jan[Hrsg.] (Lecture notes in computer science), (1999)
[13] J. Kaiser, E. Nett „Echtzeitverhalten in dynamischen, verteilten Systemen", GI-Spektrum, Themenheft: Beherrschbarkeit in verteilten Systemen,12/.98, 1998