

# A Framework for Service Provision in Intermittently Connected Mobile Ad hoc Networks

Nicolas Le Sommer

Valoria Laboratory, University of South Brittany  
E-mail: Nicolas.Le-Sommer@univ-ubs.fr

## Abstract

*Mobile ad hoc networks have been recently receiving much attention due to the important prospects they offer in service terms. Indeed, beyond the application services envisioned so far in emergency and military scenarios, civilian mobile ad hoc services usable in everyday life are considered from now. Service provision is a critical issue to the success of this new computing paradigm. In this paper, we present a novel approach for service provision in intermittently connected mobile ad hoc networks, as well as a framework implementing this approach.*

## 1. Introduction

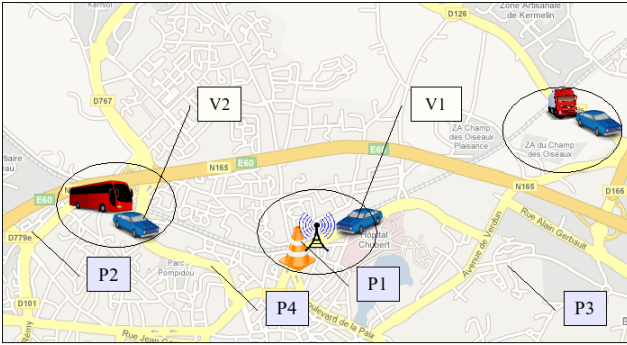
Recent technological advances in hardware computing and wireless technologies are opening up exciting possibilities for the future of mobile and pervasive computing. Indeed, beyond the application services considered so far for mobile ad hoc networks (i.e. services for emergency and/or military domains), new kinds of services relying on wireless proximate communications, and do not requiring any infrastructure-based networks, can be envisioned from now for civilians.

Most of works investigating service provision in Mobile Ad hoc NETWORKS (MANETs) consider that communication between two devices in a network is possible only if these devices are both simultaneously active, and if a transmission route can be established between them whenever needed. Making such an assumption is actually equivalent to assuming that communication can only be made using a point-to-point synchronous communication paradigm, and that it can only occur in a fully connected network. A large number of works thus aim to define and to implement routing algorithms for MANETs in order to reuse the service provision mechanisms designed for wired-networks (e.g. Jini, UPnP, SLP). Yet, when considering the large-scale peer-to-peer mobile ad hoc networks that could appear in a near future, one can observe that the topology

of these networks is very dynamic due to the mobility and the volatility of devices, and that they are frequently fragmented into small disconnected islands. Therefore, the above-mentioned assumption is somewhat contradicted by the nature of these networks.

A number of recent research papers address specifically the problem of communication in disconnected, partially connected, or intermittently connected mobile ad hoc networks. Some of these papers focus on point-to-point routing in "delay-tolerant networks" [6, 9], while others rather consider the problems of data flooding and dissemination in such networks [3, 7]. Moreover, while some of the papers assume that mobility patterns can be identified –and sometimes controlled– in ad hoc networks [11, 6], other papers make no such assumption and propose to rely on redundancy in order to improve the reliability of delay-tolerant transmission. In the latter category, it is usually proposed to rely on epidemic [10, 2, 7] or probabilistic [8] propagation schemes. Nevertheless, none of these works address specifically the service provision issues in such networks. Existing middleware solutions, such as DEAPspace [5] and Konark [4], have already investigated the service provision issues in ad hoc networks, but not specifically in intermittently connected mobile ad hoc networks. DEAPspace provides a support for the discovery and the delivery of services in single-hop wireless ad hoc networks. Konark in contrast considers multi-hop wireless ad hoc networks, but it makes some assumptions regarding the network by considering that a transmission route can be established between a client and a service provider whenever needed. The lack of solutions for service discovery and delivery thus motivated us to propose a new service provision approach for these networks.

The remainder of this paper is organised as follows. Section 2 identifies the functionalities suited for service provision in disconnected, partially connected, or intermittently connected peer-to-peer mobile ad hoc networks. Section 3 presents the framework we have designed to support service provision in such networks. Section 4 summarises our contribution and concludes this paper.



**Figure 1. Example of an intermittently connected mobile ad hoc network.**

## 2. Service provision issues in intermittently connected mobile ad hoc networks

This section presents a scenario of service provision in a peer-to-peer mobile ad hoc network, and underlines the major issues that must be addressed.

In the scenario of Figure 1, we consider provision of road traffic information services. These services –which can be delivered by devices deployed temporarily by teams carrying out road works and/or by rescue squads during accidents, and by devices installed permanently in difficult traffic areas– are expected to be used by devices embedded in vehicles (e.g. GPS) in order to notify drivers about the eventual traffic difficulties. We also consider in this scenario that the devices embedded in vehicles and those deployed in the environment are equipped with short range radio interfaces (e.g. IEEE 802.11). The topology of the network considered in this scenario thus suffers from frequent and unpredictable changes and is often partitioned into small disconnected islands.

For the sake of illustration, let us consider that a team carrying out road works in  $P_1$  (see Figure 1) has temporarily installed an equipment offering an access to a service of type of those above-considered, and that at the time  $T$ , only one vehicle, the vehicle  $V_1$ , is in the vicinity of this equipment. At this time only  $V_1$ , which is going to  $P_2$ , is able to discover and to invoke this service. Yet, it is, for instance, suitable to allow the vehicle  $V_2$ , which is going to  $P_3$ , to discover and to invoke this service as soon as possible even if no route can be established between it and the provider of this service. Therefore, at the time  $T + \Delta t_1$ , when the vehicles  $V_1$  and  $V_2$  will be both at  $P_4$ ,  $V_1$  should spontaneously advertise  $V_2$  of the availability of the considered service, or should reply to a discovery request sent by  $V_2$  for such a service. If at the time  $T + \Delta t_2$ ,  $V_2$  invokes the service thus discovered with a request similar to that addressed by  $V_2$  at the time  $T$ , and if  $V_1$  is always in the vicinity of  $V_2$ ,

$V_1$  should send to  $V_2$  the response it obtained itself from the service at the time  $T$ , if this response is still valid obviously. Such a service discovery and invocation process should occur in a recurrent manner according to the mobility of the vehicles.

Besides issues inherent in communication in mobile ad hoc networks (e.g. the minimisation the traffic generated by mobile nodes), provision of such services requires

- To provide a flexible addressing scheme so as to enable a node to send a message to a particular host, to a group of remote hosts or to all reachable hosts, thus minimising the number of messages disseminated in the network.
- To support asynchronous communications relying on the store-and-forward communication principle in order to face up to the fragmentation of the network, as well as to the mobility and volatility of devices.
- To support a content-based management of messages in order to reduce the service provision latency, and to help each device to decide what messages it should accept to store.
- To take into account spatial, temporal and contextual properties in order to improve service provision and to reduce the propagation of messages in the network.

## 3. A framework for service provision in mobile ad hoc networks

### 3.1. Overview of the framework

The core of our framework is based on the OSGi framework [1]. In this framework, services are characterised by both the Java interface they implement and their non-functional properties. Since, this framework only supports the registration, the discovery and the invocation of local services, we have defined a set of extensions in order to support a service provision process addressing the above-mentioned issues. These extensions, whose a partial UML representation is given in Figure 3 (only most pertinent elements and methods are specified), make it possible to describe local and remote services, to characterise service providers, to define messages exchanged in the service provision process, to perform a proactive and/or reactive service discovery and an asynchronous service invocation. In this framework, each mobile host is characterised by its contextual properties (e.g. hardware characteristics, position), the services it provides and it has discovered and the groups of which it is a member (see Figure 3). The concept of group aims to allow a host to send the same message to a set of remote hosts without invoking them successively.

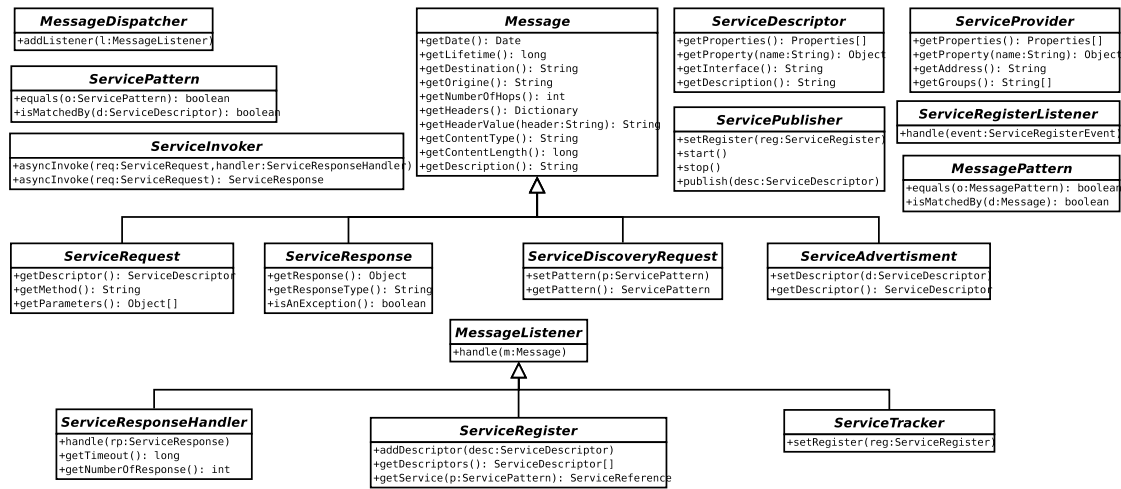


Figure 2. A partial UML representation of the framework.

All the messages exchanged asynchronously by devices include a unique identifier, temporal properties so as to decide if a message can be still considered as being valid, the identifiers of the sender and of the recipient, both spatial and contextual information to control the propagation of messages in the network, the content of the message, as well as a set of properties describing this content in order to help in the content-based management. The spacial, the temporal, and in a less measure the contextual properties, associated with the messages are used by routing protocols in order to improve message delivery in mobile ad hoc networks. These properties are rarely accessible at application levels. Yet in mobile ad hoc networks, these properties are relevant to the elements that perform the service discovery and delivery. Indeed, such properties can help them to select the best service providers, to estimate the probability to access to a given remote service, to choose the best service invocation mode (invocation of a group of providers vs invocation of a specific host), to enable client and service providers to define the lifetime and the range of their request and response, etc. Moreover unlike routing protocols, these elements are able to correlate these properties with the content of messages. A UML representation of messages is given in Figure 3.

### 3.2. Service discovery

Our framework defines functionalities to support a proactive and/or reactive service discovery. The proactive service discovery can be performed using objects implementing the interfaces *ServiceRegister*, *ServicePublisher* and *ServiceTracker* respectively. A service register is responsible for maintaining locally a list of information about

services and service providers. The service register is expected to be invoked by local services in order to discover remote services and to obtain references on service providers. When a service requested by a local service was not discovered yet, the service register should create itself a service discovery request (i.e. an object implementing the interface *ServiceDiscoveryRequest*) using the information specified by the local service (i.e. using the object implementing the interface *ServicePattern*), and should send it in the network. An object implementing the interface *ServicePattern* makes it possible to select services according to their description. When a service discovery request is received from the network by the service request handler of another node, this one is expected to check if this request is matched by a descriptor of a local service, and if it is true, it asks to the service publisher to send in the network a service advertisement for this service. Such an advertisement should be received by the local service tracker. This one is responsible for handling service advertisement from the network and to register them with the service register.

The reactive service discovery relies on the reception of unsolicited service advertisements and on their management by the service tracker. These advertisements can be send periodically or sporadically by a service publisher. As show in Figure 2, service advertisements include objects implementing the interface *ServiceDescriptor*. Service descriptors provide information about services (e.g. the name of the interface, the properties of the service).

When the lifetime associated with a service advertisement become obsolete, then the service register remove this descriptor and notifies the local services using the considered service that this one is not reachable from now. Thus local services can adapt dynamically their behaviour in order to take these changes into account.

### 3.3. Service invocation

In our framework, the asynchronous service invocation is achieved using notably the objects of type *ServiceRequest*, *ServiceResponse*, *ServiceInvoker* and *ServiceResponseHandler*. To invoke asynchronously a remote service (or a set of remote services), a local client is expected to use the methods *asyncInvoke()* defined by a *ServiceInvoker* object. The first method *asyncInvoke()* takes as parameters a *ServiceRequest* object and a *ServiceResponseHandler* object. The handler object is used by the client to handle the responses it receives following an event-based programming approach. This handler takes as parameters a timeout –which should be equal to the lifetime specified in the request– and a number specifying how many responses must be handled. This handler is registered with the *ServiceResponseListener* by the *ServiceInvoker* in order to receive responses from the network. When the timeout it received as parameter is triggered, the handler is expected to unregister itself from the *ServiceResponseListener*. A *ServiceResponseListener* is used to listen service response from the network and to dispatch them to objects of type *ServiceInvoker* and/or *ServiceResponseHandler*. The second method *asyncInvoke()* only takes as parameter a *ServiceRequest*. This method is designed to be blocked until a response is received from the network. When the timeout specified in the request is triggered, the method is expected to return a null value.

Since temporal, spatial and contextual properties are intrinsically service-dependent, client services and service providers are responsible to specify themselves these properties in the *ServiceRequest* and *ServiceResponse* objects respectively.

### 4. Conclusion and future work

In this paper, we have presented both an approach and a framework for the service provision in partially connected, intermittently connected, or disconnected mobile ad hoc networks. This approach relies on a flexible addressing scheme, on a content-based management of messages and on asynchronous communications relying on the store-and-forward principle.

The middleware platform implemented using this framework will be evaluated in the future through simulation under different network conditions, varying the network size, node mobility level, the number of services and requests. Moreover, we plan to enable bridging MANETs with infrastructure-based networks since Intranet and/or Internet remains the primary source of service provisioning.

### References

- [1] OSGi Alliance. Osgi service platform, release 3, March 2003. <http://www.osgi.org/>.
- [2] Patrick Eugster, Rachid Guerraoui, Anne-Marie Kermarrec, and Laurent Massoulié. From Epidemics to Distributed Computing. *IEEE Computer*, 37(5):60–67, May 2004.
- [3] Khaled A. Harras, Kevin C. Almeroth, and Elisabeth M. Belding-Royer. Delay Tolerant Mobile Networks (DTMNs): Controlled Flooding in Sparse Mobile Networks. In *IFIP Networking Conference, Waterloo, Ontario, CANADA*, May 2005.
- [4] S. Helal, N. Desai, V. Verma, and C. Lee. Konark : Service Discovery and Delivery Protocol for Ad-hoc Networks. In *Third IEEE Conference on Wireless Communication Networks (WCNC)*, New Orleans, USA, March 2003.
- [5] R. Hermann, D. Husemann, M. Moser, M. Nidd, C. Rohner, and A. Schade. DEAPspace - Transient ad hoc networking of pervasive devices. *Computer Networks*, 35(4):411–428, March 2001.
- [6] Qun Li and Daniela Rus. Sending Messages to Mobile Users in Disconnected Ad-hoc Wireless Networks. In *Proceedings of the Sixth Annual International Conference on Mobile Computing and Networking*, pages 44–55, Boston, August 2000. ACM Press.
- [7] Mirco Musolesi, Cecilia Mascolo, and Stephen Hailes. EMMA: Epidemic Messaging Middleware for Ad hoc networks. *Personal and Ubiquitous Computing Journal. Springer*. 10(1), pages 28–36, February 2006.
- [8] Yoav Sasson, David Cavin, and André Schiper. Probabilistic Broadcast for Flooding in Mobile Ad Hoc Networks. Technical Report IC/2002/54, Swiss Federal Institute of Technology (EPFL), 2002.
- [9] Ritesh Shah and Norman C. Hutchinson. Delivering Messages in Disconnected Mobile Ad-Hoc Networks. In *Proceedings of ADHOC-NOW 2003*, Montreal, October 2003.
- [10] Amin Vahdat and David Becker. Epidemic Routing for Partially Connected Ad Hoc Networks. Technical report, Duke University, April 2000.
- [11] Wenrui Zhao, Mostafa Ammar, and Ellen Zegura. A Message Ferrying Approach for Data Delivery in Sparse Mobile Ad Hoc Networks. In *Proceedings of ACM Mobihoc 2004*, Tokyo Japan, May 2004.