Service Provision in Disconnected Mobile Ad Hoc Networks
Nicolas Le Sommer

To cite this version:

HAL Id: hal-00498329
https://hal.archives-ouvertes.fr/hal-00498329
Submitted on 7 Jul 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Service Provision in Disconnected Mobile Ad hoc Networks

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

Abstract

With the proliferation of mobile devices equipped with communication interfaces such as IEEE 802.11 or IEEE 802.15, the mobile ad hoc networks have been recently arousing an increasing interest, and are considered from now as relevant supports for the ubiquitous and pervasive computing. The service provision in such networks arises as a critical issue for the success of these new computing paradigms. In this paper, we present both an approach and a middleware platform for the discovery and the invocation of services in disconnected, partially connected or intermittently connected mobile ad hoc networks. This approach relies on a content-based management of messages, on a flexible service invocation and on context-awareness.

1. Introduction

The generalisation of powerful portable devices capable of ad hoc communication are opening up new possibilities for service provision for nomadic people. 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 opportunistic communications should be deployed and put at the civilians’ disposal in the future.

Most of works investigating the discovery and the invocation of services in Mobile Ad hoc NETworks (MANETs) make strong assumptions regarding the communication between nodes by considering that this one is possible only if these nodes are both simultaneously active, and if a transmission route can be established between them whenever needed. This 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. By the way, a large number of works 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 real mobile ad hoc networks, one can easily observe that their topology is extremely dynamic and frequently fragmented into small disconnected islands. The above-mentioned assumptions are therefore somewhat contradicted by the nature of these networks.

A number of recent research papers address specifically the problem of message delivery in disconnected, partially connected, or intermittently connected mobile ad hoc networks by focusing either on point-to-point routing in “delay-tolerant networks” [7, 13, 9] or on data flooding in such networks using epidemic [14, 10] or probabilistic [12] propagation schemes. Nevertheless, none of these works address specifically the service provision issues in such networks. The lack of solution for service discovery and delivery thus motivated us to propose a new service provision approach for these networks.

The remainder of this paper is structured as follows. In Section 2 we identify the functionalities suited for service provision in disconnected, partially connected, or intermittently connected mobile ad hoc networks using a scenario, and we present our approach. In Section 3 we detail the mechanisms we have implemented in a service-oriented middleware platform in order to evaluate our proposition. In Section 4, we compare our approach with related works, and in Section 5 we summarise our contribution and we conclude this paper.

2. Service provision in disconnected mobile ad hoc networks

2.1. Motivating scenario

So as to point up problems inherent in the discovery and the invocation of services in disconnected, partially connected or intermittently connected mobile ad hoc networks, let us consider the scenario illustrated in Figure 1. In this scenario, students having portable devices (e.g. PDA, laptops) equipped with IEEE 802.11 network interfaces (a.k.a WiFi) are supposed to use application services while moving in a campus. These application services – that can deliver information such as course and exam timetable, campus maps, extra-scholar activities– are provided by specific
2.2. Proposition for service provision

In the remainder of this section, we describe our proposition to address the above-mentioned issues by specifying the notion of service and service provider, as well as by giving a specification of the messages exchanged in the service provision process.

A service is traditionally described by both its interface and its non-functional properties. A service $S_i$ implementing an interface $I_i$ and having a set $NP_i$ of $n$ non-functional properties can thus be defined as

$$S_i = \langle I_i, NP_i = \{np_j\} / j \in [1..n] \rangle$$

A same service, or a similar service, can be provided by several hosts in the network. A service $S_i$ can be considered as being similar to a service $S_j$ if and only if

$$I_i = I_j \text{ and } \forall x \in NP_i, x \in NP_j$$

So as to provide a flexible service invocation scheme, a context-based service selection, hosts must be characterised by a unique identifier, which can be for instance an IMEI number or a MAC address, their contextual properties (e.g. hardware characteristics, position), the services they provide and they have discovered, as well as with the groups of which they are members. The concept of group allows a host to send the same message to a set of remote service providers without invoking them successively. A host, whose the identifier is $\varphi$, can be described as

$$D_{\varphi} = \langle \varphi, LS_{\varphi} = \{S_i\} / i \in [1..p], RS_{\varphi} = \{S_j\} / j \in [1..q], CP_{\varphi} = \{cp_k\} / k \in [1..r], G_{\varphi} = \{g_l\} / l \in [1..t] \rangle$$

where $LS_{\varphi}$ is the set of services provided by the device, $RS_{\varphi}$ the set of remote services considered as available in the network by the device, $CP_{\varphi}$ the set of contextual properties exhibit by the device, $G_{\varphi}$ the groups of which the device is member.

Figure 1. Example of a disconnected mobile ad hoc network.
All the messages exchanged asynchronously by devices involved in a service provision process must 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 destination, 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 make the content-based management possible. A message can be a service discovery request, a service advertisement, a service request or a service response. These messages can be specified as follows:

\[ M_i = \langle i, O_i, D_i, C_i, CD_i, T_i, L_i, H_i, CP_i, > \]

The message \( M_i \) is identified by \( i \). The origin, the destination and the content are specified by \( O_i, D_i \) and \( C_i \) respectively. The properties describing the content \( C_i \) is specified by the field \( CD_i \). A message returned in response to another message also includes the identifier of this one. \( T_i \) is the time when the message is sent. \( L_i \) is the lifetime of the message. \( H_i \) is the spatial property expressed in number of hops. \( CP_i \) is the contextual properties exhibited by the sender \( O_i \) at the time \( T_i \). A UML representation of messages is given in Figure 3.

Using the contextual, spatial and temporal properties specified in messages, each host is able to maintain its own view of the services available in the network, to estimate the probability to invoke a remote service provider successfully, to choose the better provider among several providers, as well as to select the most relevant invocation mode to obtain a response (invocation of a particular provider vs invocation of a group of providers).

3. A middleware platform for service provision in mobile ad hoc

This section presents the middleware platform we have designed in order to evaluate the proposition specified in the previous section. This platform, whose architecture is presented in Figure 2, is based on an OSGi gateway[1]. Application-level services deployed on this middleware platform are expected to use the two middleware-level services we have designed in order to discover remote services and 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.

3.1. Service Management

The core of our framework is based on the OSGi framework. 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 issues pointed up in the previous section. These extensions, whose a partial UML representation is given in Figure 3 (only most relevant 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.

**Service discovery** The proactive service discovery is performed using objects implementing the interfaces `ServiceRegister`, `ServicePublisher` and `ServiceTracker` respectively. The 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 creates 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 sends 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 register of another node, this one checks 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 service tracker of the service requestor. 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 sent periodically or sporadically by a service publisher. As shown in Figure 3, 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 of a service advertisement is not valid, then the service register removes 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. Like the fields source and destination, the contextual properties exhibit by hosts are specified in messages as header fields. (see Figure 3).

**Service invocation**

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 in 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 in parameters a timeout –which should be equals to the lifetime specified in the request– and a number specifying how many responses must be handled. This handler uses the publish-and-subscribe functionalities provided by the first middleware-level service in order to receive responses from the network. When the timeout it received in parameter is triggered, the handler is expected to unregister itself from this first middleware-level service.

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

**3.2. The communication service**

With the service performing an epidemic dissemination of messages, any message sent in the network is maintained as long as possible in a local cache by as many devices as possible, so it can remain available for devices that could not receive it at the time it was sent originally. The underlying idea is that the dissemination of multiple copies of the same message may contribute to overcome the volatility of devices, while the mobility of these devices can itself help transport information between islands in a fragmented network or in distinct networks. Besides providing a caching system where messages can be maintained in mobile devices, our service also provides facilities for message advertisement, message discovery, and message transport between neighbouring devices. For example, a device can sporadically or periodically notify its neighbours about all or part of the messages stored in its cache. It can also look for specific messages in its neighbourhood, and either push messages toward or pull messages from its neighbours. In
the current implementation, message dissemination is performed by broadcasting UDP datagrams. We thus assume, for the time being, that each message is small enough to fit in a single datagram. In the future we plan to improve this communication service so as to address issues pertaining on message segmentation and reassembly.

Structure of messages  As shown in the previous section, all messages used in the service provision process are reified as standard Java objects. These messages can be sent in the network as XML-formatted documents. A message is structured in two parts: the content, and a set of headers providing information useful in message handling. Figure 4 shows a typical XML-formatted message sent by a service provider in return of a service invocation.

4. Related work

Existing middleware solutions, such as DEAPspace [6], Konark [5], ReMMoC [4] and INDISS [2], have investigated the service provision issues in ad hoc networks. DEAPspace provides a support for the discovery and the delivery of services in wireless single-hop ad hoc networks. The middleware Konark targets environment and goal similar to DEAPspace, but considers multi-hop wireless ad hoc networks. Like in DEAPspace, each device in Konark maintains in a directory a vision of services available in the network, and acts both as client and provider of services. Konark makes some assumptions regarding the network by considering that a transmission route can be established between a client and a service provider whenever needed. Thus it does not support the service provision in intermittently connected mobile ad hoc networks. ReMMoC and INDISS have investigated the dynamic discovery and interaction in pervasive environments, particularly focusing on configuring and using the appropriate discovery protocol to find services in the current environment. These systems have so far only considered infrastructure-based wireless networks in which standardised discovery protocols such as Jini, Service Location Protocol, and Universal Plug and Play are used.

There is now emerging a suite of discovery protocols for mobile ad-hoc networks (MANETs), which have been developed to deal with the limitations of nodes in such environments (i.e. unpredictable network topology, variable number of participating nodes, different levels of dynam-
proach aims at reducing the service discovery traffic. SSD assumes IP-level connectivity using the underlying routing protocol (e.g. OLSR, AODV, ZRP).

5. 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.

Acknowledgements

This work is done in the project SARAH. This project is supported by the French ANR (Agence Nationale de la Recherche) in the framework of the 2005 ARA SSIA program.

http://www-valoria.univ-ubs.fr/SARAH

References