Expeerience: a Jxta middleware for mobile ad-hoc networks

Mario Bisignano, Andrea Calvagna, Giuseppe Di Modica, Orazio Tomarchio
Dipartimento di Ingegneria Informatica e delle Telecomunicazioni
Università di Catania - Viale A. Doria 6, 95125 Catania - Italy
Email: {*Mario.Bisignano,Andrea.Calvagna,Giuseppe.DiModica,Orazio.Tomarchio*}*@diit.unict.it*

1 Introduction

The combination of personal computing devices and wireless ad-hoc networks allows the concept of mobile ad-hoc information system, consisting of a highly dynamic, decentralized and self-organizing network of autonomous and mobile devices that interact as peers[2, 3, 4]. Application developers have to deal with a new set of problems peculiar of these systems, due to user and terminal mobility, low bandwidth, transient loss of connectivity and lack of centralized infrastructure. It is evident that, for these classes of systems, applications cannot be designed according to the conventional architectural paradigms, such as for example the client-server one. On the contrary *"peer-to-peer"* architectures are needed, where no single host is permanently seen as a server, but where each single host is able to play the role of server and client according the user's needs. This means that an appropriate "middleware" is needed, and must be able to provide appropriate architectural abstractions to upper applicative layers [1]. This work goes towards this direction, trying to define a middleware layer, named *Expeerience*, that can answer to these issues: while doing this, we have decided to adopt as a basic software framework, an emerging P2P open technology such as JXTA [5]. The second issue we address in this work is the integration of some code mobility support in the developed middleware to allow the distribution and execution of services on peers that originally do not own the service code. This has been introduced in our platform by adding a *mobile code* service, which enables to download and install new services dynamically only if necessary.

2 Architectural design issues

Compared to middleware for wired networks, middleware design and development for ad-hoc networks will be strongly influenced by considering intermittent network connectivity as a normal lifecycle and not as an exception. Our main purpose in the design of Expeerience was to use the services provided by a P2P environment, in order to quickly build a working prototype on top of which the various modules needed for MANET environments could be developed and tested. JXTA technology may be seen as a basic framework on top of which developers can concentrate on functionality of their own applications, without worrying about low-level details that are instead provided by the underlying runtime system. Our idea is therefore that of creating another software layer on JXTA, in order to meet all the requirements that characterize MANET networks [6], and that cannot be met by JXTA. On one hand, some changes have been made to the original setup of JXTA core. On the other hand, new JXTA services have been introduced, which make new features available to the developers of services for MANET networks. The new middleware introduces these new features: management of the intermittent connections and of multiple physical interfaces, increase in the potential of resource discovery, and code mobility service. Figure 1 shows the basic ideas of our middleware framework: in particular, the changes to the core layer and the new added services have been highlighted.

Multiple interfaces and intermittent connectivity: The EndPoint structure of JXTA has been extended, in order to be able to dynamically deal with multiple interfaces (in its current version JXTA does not support neither the use of multiple network interfaces for each peer, nor the assignment of more than one address to the same interface). At a lower level the TCPTransport Service has been modified to cope with the high dinamicity of ad-hoc networks, where peers can appear and disappear very frequently from the network, without any explicit log-on or log-off procedure.

Resource discovery: Starting from the existing JXTA Discovery Service, new features have been added in Expeerience, in order to cope with the requirements of mobile ad hoc networks. The user applications will be able to configure the advertisement lifetime and to get benefit of faster advertisement storage procedures, relying on the central memory rather than on the disk cache.

Code Mobility: In Expeerience, the concept of *Mobile Service* has been introduced. The peer is given the capability of uploading/downloading a service to/from another peer. This way the service code, composed of files and classes, is migrated towards the requesting peer and is locally executed. Furthermore, the service code maintains its execution state during the migration, allowing for more complex and flexible applications on top of this architecture. The service for the code mobility is described by a specific interface *MobileCodeService*, which defines the methods for the management of mobility. These are the methods for requesting and/or sending a class and accepting a class that has not been requested. This way, the middleware we have created provides the programmer with a set of APIs. For instance, these methods enable to implement one of the pure concepts of code mobility in the scenario of MANETs with a few code lines: a mobile agent system. In fact, a mobile agent can be created, which keeps its state and resumes the interrupted execution by migrating from a peer to another and using the services provided by the MobileCodeService and the ones of the PipeService (this is another standard service of Jxta).

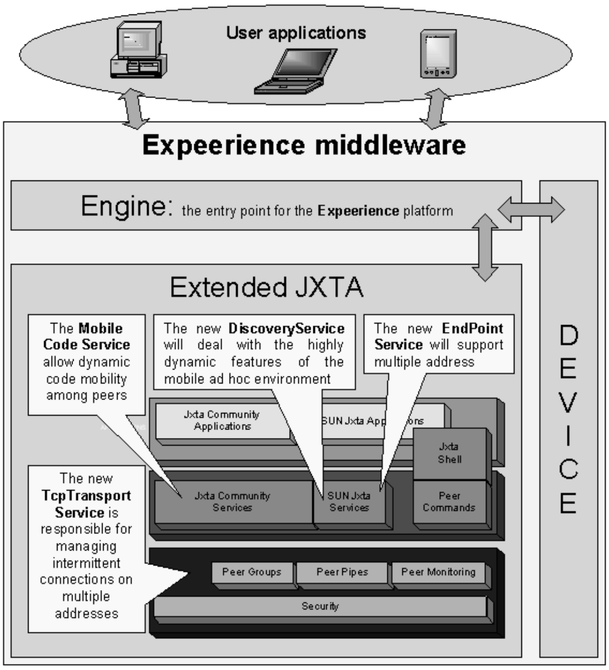
User applications
Expeerience middleware
Engine: the entry point for the Expeerience platform
Extended JXTA
The Mobile Code Service allow dynamic code mobility among peers
The new DiscoveryService will deal with the highly dynamic features of the mobile ad hoc environment
The new EndPoint Service will support multiple address
Jxta Community Applications
SUN Jxta Applications
Jxta Shell
Jxta Community Services
SUN Jxta Services
Peer Commands
The new TcpTransport Service is responsible for managing intermittent connections on multiple addresses
Peer Groups
Peer Pipes
Peer Monitoring
Security
DEVICE

Figure 1. Expeerience architecture

3 Event Organizer: an example Application

Instantaneous application availability is one of the features that can be implemented by making use of the Code Mobility Service. Whichever application can be shared to other peers in the same way as files are shared. Let us assume, as an example scenario, that some users are working on a common project and need to organize a meeting. They are all equipped with their own mobile device, but unfortunately they are in a place without any fixed network infrastructure. One of the user owns a specific application that allows to automatically synchronize appointments. By using the Expeerience middleware, this application can be shared among peers and effectively used by all involved people. The application relies on the code mobility service also for setting the meeting date: a mobile agent migrates onto the devices of the involved peers, trying to match the meeting's date and hour with the local agenda of the user. At the end of the process all of the users will have their own agenda updated with the new meeting information.

4 Conclusion and Future work

The main goal of our work was to define a middleware providing high-level support for MANET application developers exploiting P2P technology over mobile ad hoc networks. The prototype of the runtime environment, named Expeerience, has been developed in Java and is based on JXTA, which significantly simplified the development work and is a well designed P2P system with a rich set of concepts. The middleware designed fully satisfies MANET requirements: it manages the discovery service, multiple interfaces, and intermittent connectivity, and supports the code mobility for writing applications that use the mobile agent programming paradigm. Apart from the implementation of other more complex test applications to prove the benefits and the added features of our system, our future research will focus on the following areas: security, new and optimized resource discovery algorithms, and PDAs support.

References

[1] L. Capra, W. Emmerich, and C. Mascolo. Middleware for mobile computing. In *Tutorial Proc. of the International Conf. on Networking 2002*, Pisa, Italy, May 2001. LNCS 2497, Springer Verlag.

[2] M. Hannicainen, T.D. Hamalainen, M. Niemi, and J. Saarinen. Trends in personal wireless communications. *Computer Communication*, 25(1):84–99, January 2002.

[3] L. Kleinrock. Nomadic Computing and Smart Spaces. *IEEE Internet Computing*, 4(1):52–53, Jan-Feb 2000.

[4] ACM International Symposium on Mobile Ad Hoc Networking and Computing. http://www.sigmobile.org/mobihoc/.

[5] The JXTA Project. http://www.jxta.org.

[6] R. Schollmeier, I. Gruber, and M. Finkenzeller. Routing in mobile ad hoc and peer-to-peer networks: a comparison. In *Int.Workshop on Peer-to-Peer Computing. In Networking 2002*, Pisa (Italy), May 2002.