A Platform and Applications for Mobile Peer-to-Peer Communications

Takeshi Kato Norihiro Ishikawa Hiromitsu Sumino

NTT DoCoMo Inc.
3-5, Hikarino-oka, Yokosuka
Kanagawa JAPAN

Johan Hjelm Ye Yu Shingo Murakami

Ericsson Research
Torshamnsgatan 23, Kista SE-16480
Stockholm Sweden

Ericsson Research Japan
3-4, Hikarino-oka, Yokosuka
Kanagawa JAPAN

***Abstract*-The peer-to-peer service has entered the public limelight over the last few years. Several research projects are underway on peer-to-peer technologies, but no definitive conclusion is currently available. Comparing to traditional server-client technology on the Internet, the peer-to-peer technology has capabilities to realize highly scalable, extensible and efficient distributed applications. At the same time mobile networks such as i-mode, WAP, wireless LAN and Bluetooth have been developing at breakneck speed. Demand for using peer-to-peer applications over PDAs and cellular phones is increasing. The purpose of this study is to explore a mobile peer-to-peer network architecture where a variety of devices communications each other over a variety of networks. In this paper, we propose the architecture well-adapted to mobile devices and mobile network. The key features of this architecture are 1) supporting for multicast communication; 2) incorporating a mobile devices into a peer-to-peer network; 3) XML-based protocol which has readability and extensibility. In this architecture we have been designing P2P protocols using XML and algorithms for controlling the mobile peer-to-peer network. We are currently designing Java APIs needed for the mobile peer-to-peer applications. We have been implementing the mobile peer-to-peer network prototype system and some peer-to-peer applications.**

I. INTRODUCTION

In the cooperated network, the server-client technology has been used as the traditional way to handle network resources and provide Internet services. It has advantages to regulate Internet services by only maintain limited number of central servers. However some major concerns have been raised, for example overload and expensive operation cost of the central servers, needs of mutual and direct communications between network users. As a result, peer-to-peer technology has become popular and has been used in networks which manage vast amounts of data daily, and balance the load over a large number of servers. Peer-to-peer applications such as distributed search applications [2], file sharing system [3] [4], distributed storage system [5] and group ware [6] have been proposed and developed. Additionally, a generalized platform for peer-to-peer applications has been proposed [7] and developed.

At the same time, mobile Internet services have become very popular. In the past four years, the market of mobile Internet services has considerably grown successful in Japan where i-mode is the most famous example. The mobile environment is different from the fixed Internet in that it is an extremely constrained environment, in terms of both communication and terminal capabilities. This should be taken into account when developing systems which will work in a mobile environment. Additionally, various wireless have been emerging of network environments such as IMT-2000 (International Mobile Telecommunications-2000, for example FOMA [8]), Wireless LAN and Bluetooth, and users can select them to satisfy their network demands. In the near future, an environment where many sensors, users and different kind of objects exist, move and communicate with one another, called “ubiquitous communication environment”, will appear.

In fact, peer-to-peer technology is one of the most important and suitable networking technologies for ubiquitous communications since it supports easily one-to-one communication between devices, free and extensible distribution of resources and distributed search for enormous amount of resources.

The principal goal of our work is thus to design a mobile peer-to-peer architecture and a general peer-to-peer platform that enhances communication capabilities for mobile clients, by utilizing network resources efficiently and supporting mobility in an integrated and practical way.

The rest of this paper is organized as follows. Section 2 gives an overview of our peer-to-peer architecture, describes each key element of our architecture, i.e., multicast communication and Mobile device adaptation. Section 3 describes the protocol design. We report the current status of our prototyping implementation in section 4. Finally, a conclusion is given in section 5.

II. ARCHITECTURE

A. *Architecture overview*

The proposed mobile peer-to-peer architecture is shown in Fig. 1. All of the peer-to-peer communication entities that have a common set of interest and obey a common set of policies construct one peer-to-peer community. This architecture consists of the following basic components:

Peer-to-peer node: The peer-to-peer node is an independent communication entity in the peer-to-peer network. It can be a mobile device, a PDA, a personal computer, a server or a workstation, or any of a variety of devices.

Mobile proxy: Theoretically, all the mobile devices (e.g. WAP or i-mode terminals) can be independent nodes in the peer-to-

peer architecture. However some of them are functionally limited and can not act as autonomous nodes. The mobile proxy is a function in a node, which acts as a proxy for the mobile devices with constrained capability, so that these mobile devices can join the peer-to-peer architecture.

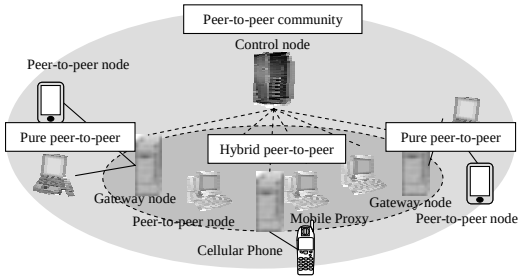

Fig. 1. The mobile peer-to-peer architecture

Based these basic components, the hybrid peer-to-peer network administrated by a control node (central point) and the pure peer-to-peer network without a control node, are defined.

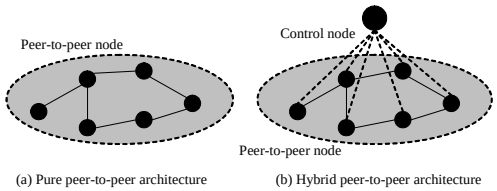

Fig. 2. Pure peer-to-peer and hybrid peer-to-peer

Pure peer-to-peer architecture: There are only peer-to-peer nodes in the pure peer-to-peer architecture. The proposed pure peer-to-peer architecture is shown in Fig. 2 (a). The connection between peer-to-peer nodes is established on their mutual trust. Each peer-to-peer node is an independent entity and can participate in and leave the peer-to-peer network at its convenience. Messages are sent from a peer-to-peer node to another one directly or by passing them via some intermediary peer-to-peer nodes.

Hybrid peer-to-peer architecture: The hybrid peer-to-peer architecture resolves the disadvantages of the pure peer-to-peer architecture such as inefficient routing, splits of network and lack of security, by introducing a control node. The proposed hybrid peer-to-peer architecture is shown Fig. 2 (b). In our architecture, the control node provides the functions for providing routing information to a destination node, discovering the first peer-to-peer node, recovering from the splitting of the peer-to-peer network, improving the network topology and security such as authentication, in order to improve the inefficiency of the pure peer-to-peer architecture.

To realize the hybrid peer-to-peer architecture, the control node and the gateway node are defined.

Control node: Control node is an administrative entity which manages a peer-to-peer community in the peer-to-peer network. It provides several functions independent of particular applications such as name resolution, route information provision, the first per-to-peer discovery, network topology optimization, node authentication and multicast group management.

Gateway node: Gateway node is a connection entity linking between a pure peer-to-peer network and a hybrid peer-to-peer network. It provides for nodes in pure peer-to-peer network with several proxy functions such as routing information provision, node authentication, and multicast group management.

A control node receives a request from a peer-to-peer node and provides it with routing information, topology optimization function and security function. A gateway node collects topology information on a pure peer-to-peer network and reports it to the control node. A peer-to-peer node in a hybrid peer-to-peer network reports its existence and adjacent nodes to the control node and can efficiently communicate with each other by using the routing information provided by the control node. A gateway node supports the seamless communication between a node in a pure peer-to-peer network and a node in a hybrid peer-to-peer network.

B. Peer-to-peer multicast communication

In peer-to-peer networks, multicast communication is available for a variety of applications, such as communications among work group members in group ware applications and so on. Peer-to-peer multicast architecture is represented in Fig. 3. As shown in this figure, a multicast network overlays a peer-to-peer network. Multicast forwarding is carried out by multi-hop or multi-destination unicast in a peer-to-peer network. Therefore it is not necessarily for all nodes in peer-to-peer network have to support multicast.

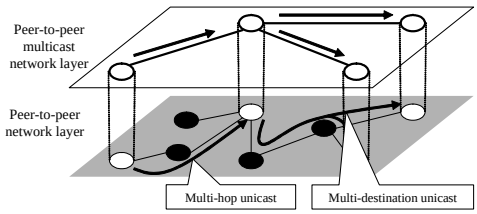

Fig. 3. Peer-to-peer multicast mechanism

Fig. 4 shows the peer-to-peer multicast path construction procedure. When a peer-to-peer node tries to join a certain multicast group, (1) a peer-to-peer node sends a join message to the nearest member node. (2) The nearest member node sends back a join response message to it and (3) the two nodes establish the relationship regarding the multicast group. (4) After that, a multicast message propagates along the multicast paths. In the pure peer-to-peer architecture, a node looks for the nearest member node using a flooding message when it tries to join a multicast group. In the hybrid peer-to-peer architecture, a control node informs a node of the nearest member node in response to the node's request.

Peer-to-peer multicast is similar to ALM (Application layer multicast) [9] [10] [11]. These forwarding mechanisms might be adopted in peer-to-peer multicast with some modifications. Additionally, bi-directional shared minimum spanning tree

might be well-suited for peer-to-peer multicast because many peer-to-peer nodes tend to be senders. MST (Minimum Spanning Tree) has the property that sum total of distances among all nodes are minimized. Therefore, each distance between nodes is well optimized.

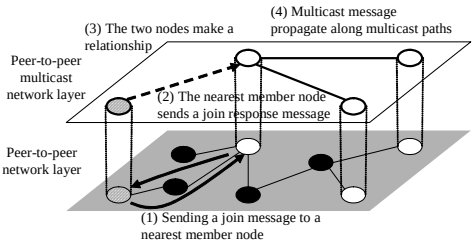

Fig. 4. Procedure of joining multicast group

C. *Mobile device adaptation*

Another distinct characteristic of this architecture is that it allows mobile devices to take part in the peer-to-peer network via a mobile proxy node. While a mobile device, such as a cellular phone, may have enough capabilities to act as an independent peer-to-peer node in the future, it still has the following limitations at this time:

- Limited storage
- Small runtime heap
- Modest processor performance
- Constrained electrical power

Thus, a current mobile device can't fully perform the role of a peer-to-peer node that offers services to other peer-to-peer nodes in a peer-to-peer network. In order to incorporate a mobile device into a peer-to-peer network, some functions must be done by other nodes on behalf of a mobile device. Through the cooperation of a mobile proxy, a mobile device can virtually act as a peer-to-peer node and can perform the necessary functions in the peer-to-peer architecture. From the point of view of the peer-to-peer architecture, mobile devices are modeled in the three ways.

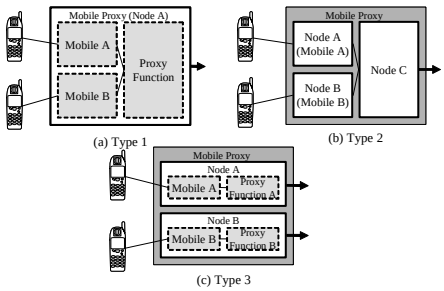

Fig. 5. Mobile proxy architecture

In Fig. 5 (a), mobile devices share the same proxy node. From the point of view of the network, the mobile proxy acts as one peer-to-peer node. In Fig. 5 (b), a mobile device has its own node name and acts as a separate peer-to-peer node in the peer-to-peer architecture. For realizing this type of mobile proxy, some proxy functions should be implemented on Node C such as transforming a message received from a mobile device into a message of a peer-to-peer protocol. In Fig. 5 (c), a pair of a mobile device and a proxy function constructs a peer-to-peer node in the peer-to-peer architecture. In this case, a mobile device has its own node name, and acts as a separate node through a mobile proxy. The mobile proxy does not act as an independent node. It will be decided by requirements of peer-to-peer applications, as to which type of mobile proxy model will be preferable.

III. PROTOCOL DESIGN

A. *Protocol overview*

Fig. 6. shows the protocol stack. The proposed protocols have been designed over HTTP, TCP and Bluetooth in two layers. P2P Core Protocol is defined to process peer-to-peer message based on peer-to-peer communication model, six protocols realizing peer-to-peer multicast, communication with a control node and control of a peer-to-peer session and so on are defined over the P2P Core Protocol. Based on the layered approach of the protocol design, it is easy to design a new P2P application protocol based on the requirements of peer-to-peer applications. Furthermore, the proposed protocols are defined using XML. Since XML has a capability to design general tree-structured data, it is possible to design complicated protocol messages required by peer-to-peer applications, and layered protocols independently using XML Namespace. Therefore XML is well suitable to design such an application protocol.

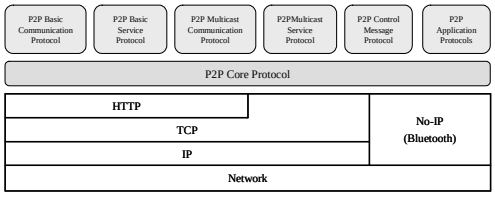

Fig. 6. Protocol stack

B. *P2P Protocols*

The P2P Core Protocol has been designed to process a peer-to-peer message according to peer-to-peer communication model. We defined three message types to realize peer-to-peer communication. Request and response messages are defined for the reactive communication mode and advertise message is defined for proactive communication mode. Additionally, we defined three communication types: unicast, multicast and broadcast. A unicast message is sent to the destination node directly, using multi-hop unicast or multi-destination unicast. When a node receives a multicast message from a multicast member node, it forwards the received message to remaining adjacent member nodes using multi-hop unicast. A node sends a broadcast message to all adjacent nodes. The forwarding of a broadcast message is controlled by its hop count. Since the naming and the message routing mechanism of the P2P Core Protocol are defined to be independent of transport protocols, the P2P core protocol can be defined over various transport

protocols. Table I shows protocols defined over the P2P Core Protocol.

TABLE I
METHODS AND MESSAGES

Protocol	Method	Message
P2P Basic Communication Protocol	Hello method	Hello
		HelloResponse
	Bye method	Bye
	Resource Information Exchange method	ResourceInformationAdvertise
		ResourceInformationRequest
		ResourceInformationResponse
P2P Basic Service Protocol	Service Provide method	ServiceAdvertise
		ServiceRequest
		ServiceResponse
P2P Multicast Communication Protocol	Join method	Join
		JoinResponse
	Leave method	Leave
P2P Multicast Service Protocol	Multicast Service Provide method	MulticastServiceAdvertise
		MulticastServiceRequest
		MulticastServiceResponse
P2P Control Message Protocol	Error Report method	ErrorReport
	Diagnose method	Diagnose
		DiagnoseResponse
	Lookfor method	Lookfor
		LookforResponse

The P2P Basic Protocol is defined to realize the establishment and the release of a peer-to-peer session. The P2P Multicast Communication Protocol is defined to construct a multicast distribution tree among multicast member nodes and to forward a multicast message over it. In the hybrid peer-to-peer architecture, the P2P Basic Service Protocol is used between a peer-to-peer node and a control node to efficiently enhance communications between peer-to-peer nodes. Additionally, the P2P Multicast Communication Protocol is used between a peer-to-peer node and a control node to efficiently enhance multicast communication between peer-to-peer nodes. The P2P Control Message Protocol is defined to provide ancillary functions such as notification of a message forwarding error, a keep-alive of a peer-to-peer session and the first peer-to-peer node discovery.

C. An example of protocol sequences

Fig. 7. shows a message exchange sequence when a peer-to-peer node participates in a peer-to-peer network and joins a multicast group. Node A sends a Lookfor message using P2P broadcast and receives the corresponding LookforResponse messages. Node A sends a Hello message to one of discovered nodes to participate in a peer-to-peer network. If Node A sends a Hello message to Node B, Node A then establishes a peer-to-peer session with Node B and exchanges its own resource information each other. Peer-to-peer nodes may exchange their own resource information. And then Node A sends a ResourceInformationRequest message using P2P broadcast to search a member node of a certain multicast group. If Node A finds a member node, it sends a Join message to the member node and receives a JoinResponse message from it. At last, Node A has joined a certain multicast group.

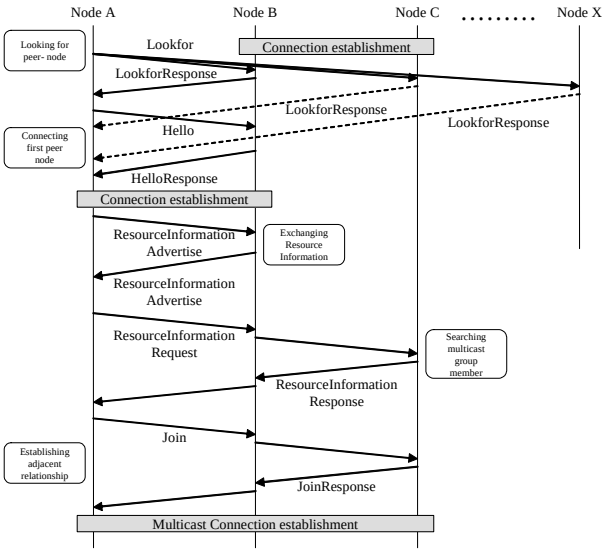

Fig. 7. Example of sequence

IV. PROTOTYPE IMPLEMENTATION

A. Tools and Open APIs

We are currently implementing a prototype for the peer-to-peer node, the gateway node and the control node. All nodes are implemented in Java (J2SE 1.3.1) on Microsoft Windows 2000 and Red hat Linux 7.2. Protocol APIs have been provided for application developers. This prototype provides the basic functions of the mobile peer-to-peer protocols and some utility tools. Fig. 8. shows the API design within a peer-to-peer node. We prepared three levels of API. The application developers can easily design a new P2P Application Protocol for a certain application using these APIs.

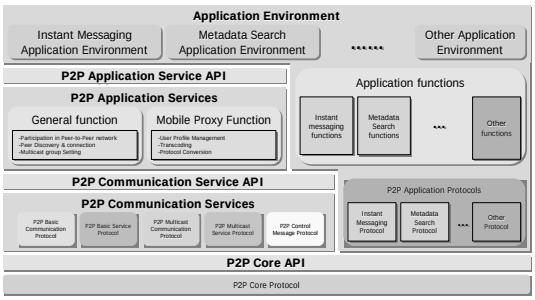

Fig. 8. Software architecture

B. Mobile peer-to-peer instant message

We have developed some peer-to-peer applications over the mobile peer-to-peer platform. In this section, we describe P2P Instant Message application using peer-to-peer multicast. This application illustrated in Fig 9., realizes instant message without central server. Each peer-to-peer node manages information of other peer-to-peer nodes and their users. Users join a certain peer-to-peer multicast group to share presence information among the members of multicast group and perform the exchange of an instant message using P2P

multicast or unicast. Additionally, this application is resilient to failure of peer-to-peer node through a recovery mechanism of mobile peer-to-peer multicast. This application might show behavior and good usage of peer-to-peer multicast.

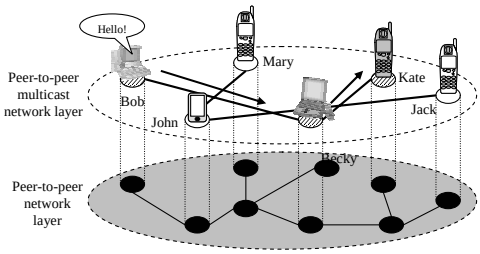

Fig. 9. P2P Instant Message application

The protocol of this application has been designed as follows. Table I shows the P2P Instant Message Protocol. It consists of two methods: Presence method and Instant Message method.

TABLE II
METHODS AND MESSAGES OF P2P INSTANT MESSAGE PROTOCOL

Protocol	Method	Message
P2P Instant Message Protocol	Presence method	PresenceInformationAdvertise
		PresenceInformationRequest
		PresenceInformationResponse
	Instant Message method	InstantMessage

The Presence method has three messages that are PresenceInformationAdvertise, PresenceInfromationRequest and PresenceInformationResponse messages. A Presence-InfromationAdvertise message is used to inform other users of sits presence. The PresenceInfromationRequest and Presence-Infromation-Response messages are used to exchange presence information between users. The Following is a sample of a PresenceInformationAdvertise message.

```
<PresenceInformationAdvertise>
  <PresenceDataStatus>Join</PresenceDataStatus>
  <PresenceData>
    <UserName>Johan</UserName>
    <UserStatus>Very busy</UserStatus>
  </PresenceData>
</PresenceInfromationAdvertise>
```

Fig. 10. A sample of PresenceInformationAdvertise

The Instant Message method has one message that is an InstantMessage message. An InstantMessage message is used to send an instant message to other nodes.

C. Simple P2P Protocol for Cellular Phones

We have designed the P2P protocol for cellular phones with a simple text format over HTTP, since they can only support HTTP and can not process the protocol based on XML. The P2P protocol for cellular phones is provided by mobile proxy. Each mobile proxy acts as a virtual peer-to-peer node for a cellular phone and converts the XML based P2P protocol to the simple P2P protocol for the cellular phone. An example of a P2P message for cellular phone is shown in Figure 12. A message included in the HTTP body, is written in a simple text format and is composed of two parts. The first part of the message corresponds to the P2P core protocol, and the second part corresponds to the application protocol.

```
P2PFRM 92
Source: Node ID of source node
MulticastGroupID: ID of multicast group
ApplicationURI: http://www.mml.yrp.nttdocomo.co.jp/ED/2003/03/p2p_instantmsg_app

InstantMessage
UserMessage: Hello All!
FRMEND
```

Fig. 12. A sample of P2P message for cellular Phone

V. CONCLUSION

We presented a mobile peer-to-peer architecture that allows a variety of devices to communicate each other using XML based peer-to-peer protocols. In this architecture, a mobile device can be incorporated into a peer-to-peer network via a mobile proxy. The seamless interconnection of nodes across various networks such as Internet and ad-hoc networks can be achieved using our peer-to-peer technology. We also described the current status of our prototype development. Currently, we are designing the APIs needed for peer-to-peer application developments.

Peer-to-peer security is now a topic we are interesting. In future ubiquitous communication environment, a lot of devices freely communicate with one another. Peer-to-peer security will be an important issue for such an environment. We are considering the incorporation of peer-to-peer security into our mobile peer-to-peer architecture. Additionally, we continue to develop new mobile peer-to-peer applications and will evaluate efficiency and performance of our peer-to-peer protocols.

VI. REFERENCES

[1] NTT DoCoMo, Inc., "i-mode", http://www.nttdocomo.co.jp/english/i/index.html.

[2] Gnutella, http://gnutella.wego.com/.

[3] NAPSTER, http://www.napster.com/.

[4] WinMX, http://www.frontcode.com/.

[5] FREENET, http://frenet.sourceforge.net/.

[6] GROOVE, http://www.groove.net/.

[7] Sun Microsystems, "Project JXTA", http://wwwjxta.org/

[8] NTT DoCoMo, Inc., “FOMA (Freedom Of Mobile multimedia Access)”, http://foma.nttdocomo.co.jp/English/.

[9] J. Jannotti, D. K. Gifford, K. L. Johnson, M. F. Kaashoek, and J. W. O'Toole Jr., “Overcast: Reliable multicasting with an overlay network,” Operating System Design and Implementation (OSDI), Oct. 2000.

[10] Y. Chawathe, S. McCanne, and E. A. Brewer, "An architecture for internet content distribution as an infrastructure service", Feb. 2000, Unpublished work.

[11] Yang-hua Chu, Sanjay G. Rao and Hui Zhang, "A Case For End System Multicast", Proceedings of ACM SIGMETRICS, Santa Clara,CA, Jun. 2000.