Instant Messaging & Presence Management in Mobile Ad-Hoc Networks

Derek Greene, Donal O'Mahony
Network and Telecommunications Research Group
Department of Computer Science, Trinity College, Ireland.
{Derek.Greene, Donal.OMahony@cs.tcd.ie}

Abstract

The combination of real-time message delivery and presence awareness has to lead the rapid rise in popularity of Instant Messaging (IM) in recent years. The extension of IM to mobile ad-hoc networks certainly offers considerable potential. However, the dynamic topology and decentralized nature of these networks requires several major issues to be addressed. Specifically, difficulties exist in providing an effective reliable service in the absence of a continuously available server. We present here the Dawn IM system, which takes a decentralized approach to bringing the functionality inherent in fixed-network IM systems to an ad-hoc environment. In addition we discuss a test-bed implementation that supports collaboration between users equipped with handheld devices.

1. Introduction

The rise in popularity of Instant Messaging (IM) in recent years has gradually led to its acceptance as a valuable method of communications in both academic and corporate environments. In particular, collaborative groups have made use of the technology to interact in a more immediate manner than afforded by other methods of electronic communication such as e-mail, while avoiding the disruption inherent in dedicated voice conversations [1].

We present here the *Dawn IM* system, a framework for the development of applications that brings the functionality inherent in popular Instant Messaging systems to a wireless ad-hoc environment. Our primary goal is to facilitate spontaneous conversations between mobile hosts where no centralized server exists.

In section 2 we examine the origins of modern IM systems. Section 3 outlines the issues involved in implementing an IM system for mobile ad-hoc networks. This is followed by a description of the core design of our system in section 4. Finally section 5 describes a test-bed implementation that supports ad-hoc collaboration between users equipped with handheld devices.

2. Development of IM Systems

2.1. Origins

Instant Messaging involves synchronous electronic communications where two or more users exchange textual or voice messages via electronic devices. Primitive messaging systems first became available many years ago, for example, UNIX *Talk* in 1973 and *Internet Relay Chat* in 1988. From the end-user's perspective, these systems operate in a manner similar to the wired telephone network in that the end-user is required to make "blind" calls, hoping that the recipient is available to take the call. Such systems lack presence-awareness.

Instant Messaging in its current form was first made widely available when the ICQ network was introduced in 1996. Subsequently companies such as AOL and Microsoft have established public consumer networks based on various similar but incompatible protocols, while others such as IBM have focused on providing private corporate messaging systems. Recently, organization such as the IETF and the Jabber Software Foundation [2] have invested considerable resources in the development of proposals detailing protocols such as *SIP for Instant Messaging and Presence Leveraging Extensions* (SIMPLE) and the *Extensible Messaging and Presence Protocol* (XMPP), each primarily aimed at providing a standard for Instant Messaging on fixed networks.

2.2. Features of IM Systems

In its simplest form, IM combines bi-directional communications with information describing the status of each user on the network. A textual conversation may be formed via the exchange of short, simple messages that are delivered in approximately real-time. While *Talk* involved one-to-one conversations and relay-type messaging focused on multi-user "chat rooms", most IM systems combine these approaches, providing more flexible user interaction.

The most important aspect of IM that differentiates it from earlier systems is the integration of presence awareness, providing the ability to monitor the status of other users on the network. In most IM client applications, this information is presented in the form of a "buddy list", which displays a list of contacts along with their current status. If a particular user is "online" and willing to receive communications, an indicator will display this information on the buddy lists of those users who have subscribed to that user's presence information.

It is this combination of immediacy with the use of status information to provide integrated Presence and Availability awareness that has been pivotal in encouraging the swift adoption of this medium for communications.

2.3. Presence and Availability

Presence information refers to data describing an individual's current online status and ability to receive various forms of communications. For example, this information might describe whether a user is capable of receiving a telephone call or an IM message at any given time. It may also extend to supplementary information such as preferred medium for contact or current geographical location.

Availability denotes the willingness of a user to communicate depending upon a specific context, usually determined by the type of communications being requested and the identity of the other user making the request. For example while an IM user may be able to receive messages from any other user on the same network, in practice he may wish to be unavailable to all but his known contacts to prevent intrusion by unsolicited messages.

In IM systems, the concepts of presence and availability are usually closely linked. A typical client application will inform the user when a contact's state changes, and also provides him with the ability to easily alter his state. For the remainder of this paper I use the term "presence awareness" to refer to knowledge concerning both presence and availability.

3. Instant Messaging in Ad-hoc Networks

3.1. Overview

Ad-hoc networks allow the instantaneous formation of networks without the cost and time required to establish a physical infrastructure. There has been particular interest in the area of mobile ad-hoc networks (MANETs) due its potential for enabling ubiquitous computing. Possible applications include spontaneous discussions between work group members and cooperation between emergency services personnel working in the field.

The extension of Instant Messaging to mobile ad-hoc networks certainly offers considerable potential. However, the dynamic topology and decentralized nature of these networks requires several major issues to be addressed. In particular, difficulties exist in providing an effective reliable IM service in the absence of a continuously available server. In this section we discuss two recent proposals for ad-hoc systems supporting presence awareness and messaging.

3.2. Related Work

The *DoMo* pervasive computing environment [5] is a proposal to extend the *AIDA* Jabber-based Instant Messaging client for Palm OS to provide a generic means of sharing presence information between devices on wireless and fixed networks. This model covers not only the status of end-users, but also extends the concept of presence awareness to shared resources such as documents and peripheral devices. The network architecture in the *DoMo* environment relies primarily on the existence of a central Jabber server to handle message exchange and presence distribution. This server is in turn connected to a wireless access point that serves as a gateway for wireless devices and provides connectivity to resources on a fixed network. By requiring a permanent fixed server and communications via a gateway node, the range and flexibility of this approach are limited, offering little more freedom than afforded by a wired network.

An alternative approach [6] describes a fully decentralized system for the provision of Instant Messaging on ad-hoc community networks. It does not depend on a fixed server or gateway, as state information is managed in a distributed manner. Each node maintains a view of the topology of their "neighborhood" of the network. Peer location is achieved by using a modified link-state routing protocol to find a path to any given node. Presence awareness is closely coupled to the routing protocol, as the link-state table is also used to store the online status of known nodes. The exchange of chat messages and presence updates takes place over an ad-hoc "connection", which involves sending packets over the optimal path between nodes. This path may be adapted over time as the network topology changes to maintain stability without disrupting the operation of node.

System [6] also extends to the provision of offline message delivery, a useful feature found in some commercial IM systems, where chat messages sent to an offline user may be buffered and delivered when they rejoin the network. The authors suggested the use of proxy nodes to temporarily store unsent messages for offline recipients. When a node chooses to leave the network, it may nominate another node to receive messages on its behalf. In general, the decentralized approach chosen by this system provides greater opportunity for spontaneous collaboration, as it is independent of any fixed infrastructure.

4. Design Issues

4.1. Overview

In this section we discuss our approach to addressing the major issues in designing a fully decentralized IM system to be deployed in the *Dublin Ad-hoc Wireless Network* (DAWN). In particular, we focus on methods for locating other users, distributing presence information and forming multi-user group conversations.

4.2. Decentralized Operation

Current consumer and corporate IM systems are primarily based on variations of the client-server architecture. In this configuration the complexity resides on the server side, with clients sending text, presence or other control messages to a central server. The server is then responsible for handling the messages and taking a suitable action, such as forwarding a text message to its intended recipient or distributing presence updates.

The communications model we have chosen differs significantly from proposals such as *DoMo* or the model often employed by contemporary wireless networks, where roaming mobile hosts communicate via gateways that connect to established servers running on a fixed LAN. Our primary goal is to enable the spontaneous collaboration of small-scale working groups without the need for complex configurations or previously installed infrastructure. Any reliance on fixed centralized servers would appear to negate many of the benefits inherent in wireless ad-hoc, limiting the range and versatility of the system. For the *Dawn IM* system we have opted for a fully decentralized approach to communications between nodes, emphasizing reliability and robustness.

4.3. Peer Location

Methods for peer location form a vital part of the design of peer-to-peer networks, and the choice of algorithm or protocol can be a major factor in the reliability and efficiency of the overall network. In a decentralized Instant Messaging system, peer discovery involves finding other users on the network so that text messages and presence information can be routed to/from those users from/to the local user at a given node.

One of the major challenges of a fully decentralized system is how to route messages to their destination, as no central authority exists to manage address resolution. This problem is exacerbated in mobile ad-hoc networks, as nodes are constantly moving and may be rapidly coming in and out of transmission range. To address this problem we have chosen to use *Nom*, a resource location and discovery system proposed by Doval [4]. It combines a custom peer-to-peer protocol with the construction of an overlay network to efficiently query the location of resources.

4.4. Presence Management and Distribution

In a fixed IM network presence information for all entities is managed by one or more continuously available servers. It is the responsibility of a server to track the state of each entity on the network, and respond to changes in their state (e.g. an entity leaving the network) by supplying presence updates to interested parties. However, a fully decentralized network lacks this form of central repository. In a mobile ad-hoc network the problem is more complex due to the highly dynamic topology of the network.

In the Dawn IM system, each entity manages its own presence information and is responsible for informing other parties of its current state. We have chosen to use an existing presence and availability protocol, namely that described in the *XMPP* specification [9]. This protocol is currently used by *Jabber* IM systems and includes a description of transport semantics for the exchange of presence and availability data. The generic nature of the specification is such that it is not tied to any specific architecture, making it suitable for use in a decentralized network.

While the core *XMPP* protocol describes semantics for presence messages, it does not extend to specifying how the information should be distributed. We therefore examined the two models for presence management described in RFC-2778 [8]. The "subscription" approach involves the subscriber asking other entities to notify it of any future changes in their presence state, while the "fetching" model involves making single unconnected requests to query an entity's current status. Fetching on a periodic basis effectively constitutes "polling". Wired-network IM systems make use of presence distribution based on subscription methods, where subscriptions are initiated and fulfilled through a central server.

However, subscriptions as described above are temporary, rendering them ineffective in a fully decentralized system. For example, when an entity joins the network, it will have to inform the others of its interest in subscribing to their presence information. In a centralized system, the IM server can keep track of these requests and provide notifications when entities change their status. In contrast, an entity in a decentralized system will need to directly inform another entity of its interest in receiving presence requests. If the target node is not contactable (e.g. switched off, out of range), the sender will need to wait an arbitrary period of time before making another request. Effectively the node may need to resort to polling in order to establish a subscription. In addition, if a node becomes unavailable, it may be difficult to ascertain whether the user has closed the client application or has merely gone out of transmission range. If the same node subsequently appears again, we cannot be sure whether that node has retained previous subscription information or has discarded it.

For a small-scale private network, the use of fetching on a periodic basis provides reliable distribution of presence and availability information, without adding unnecessary complexity to the protocol and with an acceptable level of bandwidth usage.

4.5. Conversation Management

In their approach to message exchange, existing wired-network IM systems generally fall into two categories. All major consumer IM systems make a clear distinction between one-to-one conversations and multi-user “chat rooms”. However, many corporate systems allow private chat sessions to be extended via invitations. In either case, the list of chat sessions and members for each conversation are maintained on a central server. As result this information may be easily queried at any time by entities involved in a given conversation. However, since we have opted not to make use of a centralized architecture, the burden is on each node to maintain state information relating to all group conversations in which it currently participates.

We have chosen to follow a “group-based” approach to managing IM conversations, using a single protocol for both one-to-one and multi-user message exchange. A basic multicast scheme is used to send messages to the other members of the group. Group views are maintained at each node, and are synchronized via control messages sent to all members whenever a node joins or leaves the group. “Leave” messages indicate that an entity wishes to explicitly leave the group, while presence updates are used to detect when nodes within a group have gone out of range, and group members are informed of their temporary unavailability by the underlying presence layer. Messages are routed in a peer-to-peer manner using the underlying *Nom* layer to locate nodes. Group members may invite additional users to join the conversation at any time, so that one-to-one conversations may be extended to involve many users.

5. Implementation

The system was implemented as an extension to the generic protocol stack model developed by researchers at the NTRG, Trinity College Dublin [10]. This model involves the creation of networking applications through the creation of individual protocol “layers” that are assembled into the desired configuration for a given task and platform. The configuration for the *Dawn IM* application is illustrated in figure 1.

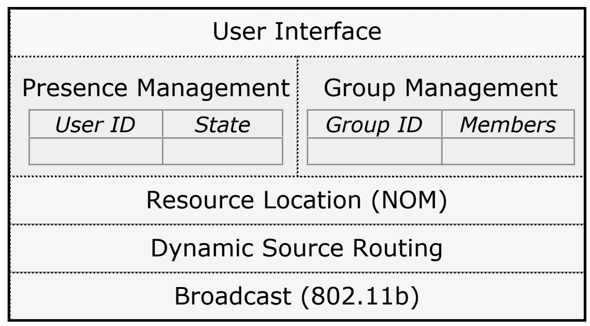

Figure 1. Protocol stack (all nodes)

The system was initially prototyped as a desktop application, where the *JEmu* network emulator [11] was used to simulate the behavior of multiple nodes in a wireless network. The system was then ported to the Microsoft Windows CE operating system and deployed on commercial off-the-shelf PDA devices such as the Compaq iPAQ.

6. Discussion

Tests involving groups of users equipped with *Dawn IM* clients on PDA devices proved successful, with presence information and text messages being distributed reliably via wireless adapters on the handheld devices. The facility to easily extend one-to-one conversations to involve multiple users allowed participants to easily join and leave groups, facilitating spontaneous collaboration.

The system that we have proposed was tested in a small-scale deployment in an internal environment where users trust each other. However, we envisage that the solution would scale to a larger-scale deployment with adjustments to parameters in the presence layer. In particular the underlying implementation of *Nom* facilitates the automatic construction of an overlay network for efficient peer location in a large network.

Alternative methods for presence distribution could yield increased efficiency by reducing bandwidth usage. An approach to be considered in future work is the assignment of additional duties to certain nodes, such as acting as a repository of presence information for all nodes in a "neighborhood". Effectively the scheme would function in a similar manner to "super-peers" in P2P networks such as Gnutella. The establishment of a limited hierarchy could bring many of the performance benefits of centralization to a mobile ad-hoc network [12]. However the degree of improvement would be heavily dependent on the choice caching and distribution algorithms.

We also recognize that any public deployment would require that the privacy concerns involved in the aggregation of presence information and inter-personal messaging would be addressed using appropriate security measures. The stack-based architecture upon which the DAWN IM system is developed is specifically designed to facilitate the addition of layers that provide supplementary capabilities such as enhanced security. A stack component suitable for dealing with many of these privacy issues is proposed in [13], which provides secure group formation using cryptographic techniques suitable for small devices with limited processing power.

7. Conclusions

In this paper we have presented a system that implements the core elements of Instant Messaging – presence awareness and group conversations – in a mobile ad-hoc environment. However, the dynamic nature of these networks makes the centralized model prevalent in fixed-network IM systems unsuitable. In response we have suggested an alternative peer-to-peer approach affording similar functionality without the need for a continuously available server or gateway.

In general, our experience with the *Dawn IM* system suggests that the integration of Instant Messaging and wireless ad-hoc networks has the potential to create a new form of application that encourages spontaneous communications in mobile computing environments. Specifically, the main advantage of such an application lies in the enablement of interaction between roaming users without the need for dedicated servers or expensive physical infrastructure. Furthermore, we believe that the provision of presence awareness in applications for mobile devices will serve to enrich the end-user experience and facilitate more efficient collaboration between members of working groups.

References

[1] B. Nardi, S. Whittaker, E. Bradner, "Interaction and Outeraction: Instant Messaging in Action," in *Proceedings of ACM 2000 Conference on Computer Supported Cooperative Work*, 2000.

[2] Jabber Software Foundation website, 2003, http://www.jabber.org.

[3] P. Godefroid, J.D. Herbsleb, L.J. Jagadeesan, Du Li, "Ensuring Privacy in Presence Awareness Systems: An Automated Verification Approach," in *Proceedings of ACM 2000 Conference on Computer Supported Cooperative Work*, Philadelphia, Pennsylvania, 2000.

[4] D. Doval, D. O'Mahony, "Nom: Resource Location and Discovery for Ad Hoc Mobile Networks," in *Proceedings of the First Annual Mediterranean Ad Hoc Networking Workshop,* Sardegna, Italy, 2002.

[5] J. Favela, C. Navarro, M. Rodriguez, "Extending Instant Messaging to Support Spontaneous Interactions in Ad-hoc Networks," in *Proceedings of ACM 2002 Conference on Computer Supported Cooperative Work* (CSCW '02), New Orleans, Louisiana, 2002.

[6] T. Ueda, S. Bandyopadhyay, K. Hasuike, "Implementing Messaging Services on Ad Hoc Community Networks using Proxy Nodes," in *Proceedings of 2nd Swedish Workshop on Wireless Ad-Hoc Networks*, Stockholm, 2002.

[7] M. Khambatti, S. Akkineni, "Location management in peer-to-peer mobile ad-hoc networks," Arizona State University, Tech. Rep, 2002.

[8] P. Saint-Andre, J. Miller, "XMPP - Core Protocol (Draft)," IETF Network Working Group, 2003.

[9] M. Day, J. Rosenberg, H. Sugano, "A Model for Presence and Instant Messaging (RFC 2778)," IETF IMPP Working Group, 2000.

[10] D. O'Mahony, L. Doyle, "An Adaptable Node Architecture for Future Wireless Networks," in *Mobile Computing: Implementing Pervasive Information and Communication Technologies*, Kluwer series in Interfaces in OR/CS, Kluwer Publishing, 2001.

[11] J. Flynn, H. Tewari, D. O'Mahony, "A real time emulation system for ad-hoc networks," in *Proceedings of the Communication Networks and Distributed Systems Modeling and Simulation Conference 2002*, San Antonio, Texas, 2002.

[12] B. Yang, H. Garcia-Molina, "Designing a Super-Peer Network," in *Proceedings of the 19th International Conference on Data Engineering*, 2003.

[13] R. Verma, H. Tewari, D. O'Mahony, "Wanderer: Secure Group Formation and Communication in Ad Hoc Networks", in *Proceedings of the 2nd Joint IEI/IEE Telecommunications Systems Research Symposium*, Dublin, 2003.