

Epidemic Dissemination of Presence Information in Mobile Instant Messaging Systems

Christoph Lindemann and Oliver P. Waldhorst

University of Dortmund, Department of Computer Science,
August-Schmidt-Str. 12, 44227 Dortmund, Germany
<http://mobicom.cs.uni-dortmund.de>

Abstract This paper presents an approach for exchanging presence information between users of an instant messaging system in a mobile ad hoc network. As major feature, presence information is transferred when mobile users get in direct contact, similar to the spread of an infectious disease. By exploiting node mobility, presence information is epidemically distributed throughout the network, effectively overcoming network partitions. We show how to apply the Passive Distributed Indexing Protocol, which implements a general-purpose lookup service for mobile applications building upon epidemic data dissemination, for implementing the exchange of presence information. The effectiveness of the approach is illustrated in a simulation study using the network simulator ns-2. Building upon the results, we present the architecture of a mobile instant messaging system that supports the widely adopted Extensible Messaging and Presence Protocol (XMPP), an IETF standardized protocol for instant messaging.

1 Introduction

Pervasive computing devices such as Personal Digital Assistants (PDAs), mobile phones, and notebooks can be found in every situation of human life. Beside using these devices for the management of personal information, an upcoming application is message-based real time person-to-person communication, better known by the term instant messaging. Beside communication, instant messaging systems enable a user to track the presence state of his contacts, i.e., to determine in real time whether a peer user is currently online and available for a chat. Many instant-messaging systems have been developed for the wired Internet, including the AOL instant messenger, the Microsoft instant messenger, ICQ, or the Jabber instant messaging network. The rapid adoption of the protocol used by the Jabber network resulted in standardization efforts within the Internet Engineering Task Force, that led to development of the Extensible Messaging and Presence Protocol (XMPP, [14], [15]).

Unfortunately, the mechanisms defined by the various approaches to instant messaging are not well suited for self-organizing networks of mobile devices, so called mobile ad hoc networks [12] for several reasons. On the one hand most systems require a centralized server for managing communication and presence information. Unfortunately, in a MANET in general no devices with extraordinary capabilities

exist, that are suited to implement centralized functionality. On the other hand, limited transmission range and node mobility lead to frequent partitions of the network, which hinder the communication with nodes providing centralized functionality. However, as shown by Grossglauser and Tse node mobility does not only hinder communication in MANET, but also supports cost-effective information exchange by epidemic dissemination [4].

In previous work [9], [10], we introduced a distributed lookup service, denoted as Passive Distributed Indexing (PDI) and proposed consistency mechanisms for keeping the distributed index coherent. PDI stores index entries in form of (key, value) pairs in index caches located in each mobile device. Index caches are filled by epidemic dissemination of popular index entries. That is information is transmitted when nodes get in direct contact, similar to the transmission of an infectious disease between individuals. By exploiting node mobility, PDI can resolve most queries locally without transmitting messages to nodes outside the radio coverage of the inquiring node. Thus, PDI can effectively cope with network partitions. For keeping index caches coherent, we introduced two consistency mechanisms for PDI index caches: (1) configurable value timeouts implementing implicit invalidation and (2) invalidation caches implementing explicit invalidation. Simulation results showed that with the suitable integration of both invalidation mechanisms, more than 95% of results delivered by PDI index caches are up-to-date for numerous application scenarios.

In this paper, we show how to apply PDI for implementing a mobile instant messaging system, which relies on epidemic dissemination of presence information in a MANET. As basic concept, the system employs PDI query messages for polling presence information, and a combination of lazy invalidation caches and value timeouts to provide presence state consistency. Performance results show that in systems with more than 50 participating users this approach resolves the presence state of contacts in more than 70% of all cases, with stale results in below 10% of the cases. Motivated by these results we outline the architecture of a mobile instant messaging system that is compliant to the XMPP protocol. The system satisfies all of XMPP's basic requirements, such as message exchange, exchange of presence information, maintenance of subscriptions and rosters, and blocking communication.

The remainder of this paper is organized as follows. Related work in epidemic data dissemination is discussed in Section 2. To make the paper self-contained, Section 3 summarizes the basic concepts of PDI. Section 4 discusses how PDI can be employed to implement the exchange of presence information in a mobile instant messaging system. Performance curves illustrating the effectiveness of this approach are presented in Section 5. Section 6 outlines the architecture of an XMPP compliant instant messaging system based on PDI. Finally, concluding remarks are given.

2 Related Work

As a first approach to epidemic information dissemination in mobile environments, Papadopouli and Schulzrinne introduced seven degrees of separation (7DS), a system for mobile Internet access based on Web document dissemination between mobile users [13]. To locate a Web document, a 7DS node broadcasts a

query message to all mobile nodes currently located inside its radio coverage. Recipients of the query send response messages containing file descriptors of matching Web documents stored in their local file caches. Subsequently, such documents can be downloaded with HTTP by the inquiring mobile node. Downloaded Web documents may be distributed to other nodes that move into radio coverage, implementing an epidemic dissemination of information.

Using a related approach, Goel, Singh, Xu and Li proposed broadcasting segments of shared files using redundant tornado encoding [3]. Their approach enables nodes to restore a file, if a sufficient number of different segments have been received from one or more sources. In [8], Khelil, Becker, Tian, and Rothermel presented an analytical model for a simple epidemic information diffusion algorithm inspired by the SPIN-1 protocol [5]. Both systems implement a push model for information dissemination. That is, shared data is advertised or even actively broadcasted without a node requesting it. Hanna, Levine, and Mamatha proposed a fault-tolerant distributed information retrieval system for P2P document sharing in mobile ad hoc networks [6]. Their approach distributes the index of a new document to a random set of nodes when the document is added to the system. The complete index of a document, i.e., all keywords matching it, constitutes the smallest unit of disseminated information. Recently, Small and Haas proposed an epidemic approach for collecting information in a hybrid network consisting of mobile nodes and fixed infostations [16]. Their architecture, denoted as Shared Wireless Infostation Model (SWIM), actively transfers information among wireless nodes on each contact, until information is unloaded to one of the infostations.

Opposed to all related work on epidemic data dissemination in mobile networks, PDI cannot only be employed for document and information sharing, but also for numerous other mobile applications. The ability of PDI to implement search functionality in a mobile peer-to-peer file sharing application has been demonstrated in [9]. In this paper, we show how PDI can be applied to disseminate presence information in a mobile instant messaging system.

3 Epidemic Data Dissemination by Passive Distributed Indexing

To keep the paper self-contained, we recall the basic concepts of PDI as introduced in [9] [10] in this section. All functionality described below is specified in an upcoming Internet draft [11], and thus ready to be used in various mobile application. PDI is designed for a system consisting of several mobile nodes, e.g. mobile users equipped with notebooks or PDAs and wireless network interfaces as illustrated in Figure 1. All mobile nodes collaborate in a shared application that uses a distributed lookup service. Radio coverage is small compared to the area covered by all nodes, e.g., less than 5% of the covered area. Typically, communication is performed using several intermediate hops as in mobile ad hoc networks (MANET, [12]). Subsequently, we assume IEEE 802.11x in the ad hoc mode as underlying radio technology [7]. However, we would like to point out that PDI could be employed on any radio technology that enables broadcast transmissions inside a node's radio coverage.

PDI implements a general-purpose lookup service for mobile applications. In general, PDI stores index entries in the form of pairs (k,v) . Keys k and values v are

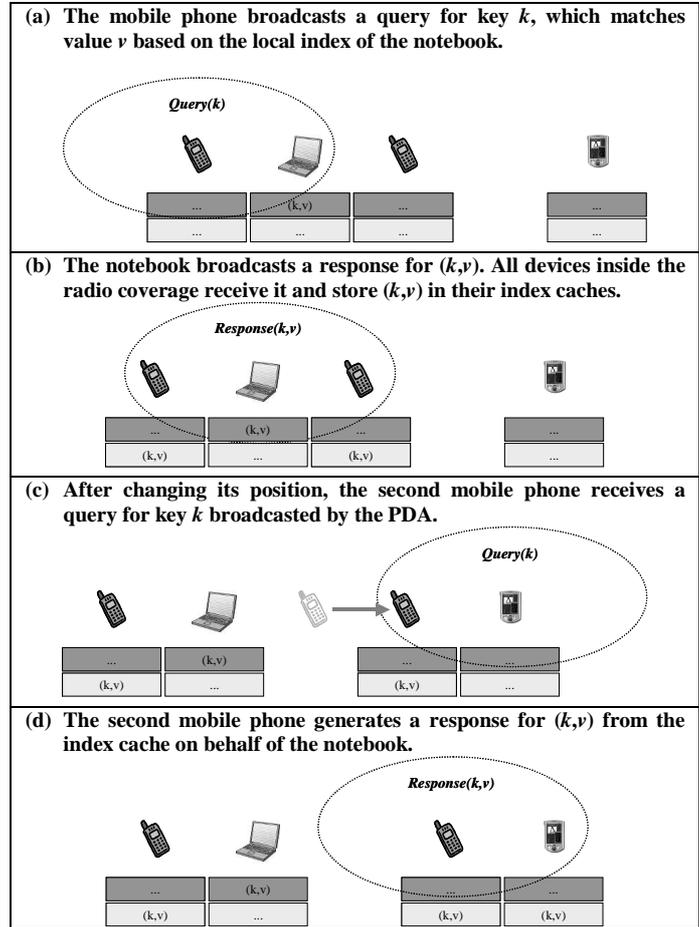


Fig.1. Illustration of epidemic information dissemination with PDI

both defined by the mobile application. For example, in case of a file sharing application, keys are given by keywords derived from the file name or associated meta-data. Values are given by references to files in form of URIs. PDI does not require a one-to-one matching between keys and values. However, some mechanisms implemented in PDI require that a value is unique in the system. That is, it is only added to the system by a single node. This can be easily achieved by extending values by unique node identifiers. A node n may contribute index entries of the form (k,v) to the system by inserting them into a local index. In Figure 1, the local index is drawn as the first box below each mobile device. We refer to such an index entry as supplied. The node n is called the origin node of an index entry. For example, the notebook shown in Figure 1 is the origin node of the index entry (k,v) . A key k matches a value v , if (k,v) is currently supplied to the PDI system. Each node in the system may issue queries in order to resolve a key k to all matching values v_i (see Figure 1a). A node issuing a query is denoted as inquiring node.

Queries are transmitted by messages containing the key to resolve. Query messages are sent to the IP limited broadcast address 255.255.255.255 and a well-

defined port, using the User Datagram Protocol UDP. Using the IEEE 801.11 ad hoc mode, all nodes located inside the radio coverage of the inquiring node receive a query message. Each of these nodes may generate a response message. A response message contains the key from the query and all matching values from either the local index or a second data structure called index cache which is drawn as a second box below the mobile devices in Figure 1. To enable epidemic data dissemination, PDI response messages are sent to the IP limited broadcast address 255.255.255.255 and a well-defined port, too. Thus, all mobile nodes within the radio coverage of the responding node will overhear the message (Figure 1b). Not only the inquiring node but also all other mobile nodes that receive a response message extract all index entries and store them in the index cache (see Figure 1b). Thus, the distributed index is maintained in a passive way. In Figure 1, index caches are drawn as the second box below mobile devices. Index entries from the index cache are used to locally resolve queries, if the origin nodes of matching values reside outside the radio coverage of the inquiring node (see Figures 1c and 1d). Obviously, the index cache size is limited to a maximum number of entries adjusted to the capabilities of the mobile device. The replacement policy least-recently-used (LRU) is employed if a mobile device runs out of index cache space. By generating responses from index caches, information is disseminated to all other nodes that are in direct contact, similar to the spread of an infectious disease.

To extend the dissemination of information beyond the radio coverage of the inquiring node, PDI includes a message forwarding mechanism. When message forwarding is enabled, queries are relayed a certain number of hops specified by the inquiring node in a time-to-live field (TTL_{query}). Similarly, response messages are forwarded TTL_{query} hops. Since response messages may be considerably larger than query messages, PDI incorporates a concept called selective forwarding. That is each node that receives a response message will search the index cache for each index entry contained in the message. When an entry is found, in most cases the node itself has already responded with this index entry. Therefore, forwarding this index entry constitutes redundant information. Using selective forwarding, each relay node removes all index entries found in its local index cache from the response message, before the message is forwarded.

Caching index entries in mobile devices may introduce inconsistency. PDI incorporates a message-based explicit invalidation mechanism to remove stale index entries for a given value v from the index caches. Flooding is a straightforward way to propagate invalidation messages. Unfortunately, a flooded invalidation message might not reach all nodes in case of network partitions. Subsequently, stale index entries remain in index caches of nodes not reached by invalidation message. To cope with this problem, we introduced a passive approach for epidemic propagation of invalidation messages denoted as lazy invalidation. Each node maintains a data structure called invalidation cache. When a node receives an invalidation message for a value v it does not only relay it, but stores v in the invalidation cache for epidemic dissemination of invalidation messages. Cached invalidations are sent when a node receives a response containing a stale value. To cope with stale information due to sudden departures of nodes, we introduced the concept of value timeouts. Value timeouts limit the maximum time an entry (k, v) for a given value v will be stored in an index cache. Value timeouts rely on epidemic dissemination of recent timeout

information for each value. Simulation results showed that with the suitable integration of both invalidation mechanisms, more than 95% of results delivered by PDI index caches are up-to-date for a mobile peer-to-peer file sharing application [10].

4 Applying PDI for Mobile Instant Messaging

Recall that a major function of an instant messaging system is the monitoring of the presence state of users participating in the system. That is, a user is able to determine if one of his contacts is online, busy, free for chat, etc. Such information is denoted as the presence state of a contact. In state-of-the-art instant messaging systems, e.g., in systems that use the Extensible Messaging and Presence Protocol (XMPP, [14],[15]), the instant messaging software will notify the user when a contact changes his presence state. XMPP specifies the usage of dedicated servers that send presence state updates to all other users that have a particular user on their contact list, if this user changes his presence state. Unfortunately, when implementing an instant messaging system in MANET, a centralized approach using a dedicated server for communication of presence state cannot be employed. An alternative solution might communicate presence state directly between the users using direct connections. Nevertheless, frequently occurring network partitions will hamper this solution, as we will show in Section 5.

To cope with these shortcomings of MANET, we employ epidemic dissemination of state information based on PDI. We assign a unique identifier to a user, denoted as his instant messaging identifier (IM-ID). Each mobile user contributes his current presence state to the system as only value. The value matches only a single key, given by the user's IM-ID. Note that this results in a one-to-one matching of keys and values. The IM client software periodically polls the presence state of each contact on a users contact list, using a PDI query for the user's IM-IDs. Presence information received in PDI response messages will be displayed to the user. Subsequently, on a presence state change of one user, the client software will not explicitly notify other users, but sends a PDI invalidation message for the old present state, and subsequently contributes the new state to the system. The invalidation will be disseminated epidemically, using lazy invalidation caches. For additionally increasing the coherence of reported presence states, the client software uses a value timeout, which is set to the average time between presence state changes. The client software can easily calculate this time. We will show in the Section 5 that this methodology enables the determination of the current presence state in many cases. In Section 6, we show how to implement an entire XMPP compliant instant messaging system based on PDI.

5 Simulation Study

5.1 Simulation Methodology

In this section, we show that PDI can be effectively employed to implement a communicate presence information in an instant messaging system in MANET. For

this purpose, we conduct simulation studies based on the network simulator ns-2 [2]. We have developed an ns-2 application implementing the concept of PDI as described in Section 3. An instance of the PDI application is attached to each simulated node, using the UDP/IP protocol stack and a MAC layer according to the IEEE 802.11 standard for wireless communication. All MAC layer parameters are configured to provide radio coverage with radius of 125m for each mobile node. We assume that N mobile nodes, for $N = 20, 35, \dots, 110^1$, move in an area of $1000 \text{ m} \times 1000 \text{ m}$ according to the random waypoint mobility model [1]. Maximum node speed is 1.5m/s and a pause time between two movement epochs is 50s. The random waypoint model is commonly used to mimic the movement of pedestrians.

To model a mobile instant messaging application implementing the concepts described in Section 4, we assume that each mobile user maintains a list of his/her favorite contacts. Each user u is on the contact list of each other user with probability Cu^{-1} , where C is a constant. I.e., the social contacts are described by a Zipf distribution. To account for state changes, we assume that users change availability state in intervals of exponentially distributed length with mean 300s. On a state change, the contributed value expires and is replaced by a new one. Furthermore, function we assume that users arrive and depart with exponentially distributed arrivals. Arrival rates are chosen such that 30% of the nodes arrive and depart during a simulation run.

We conduct transient simulations starting with initially empty caches. For each run, total simulation time was 2 hours. To avoid inaccuracy due to initial warm-ups, we reset all statistic counters after a warm-up period of 10 min. simulation time. For each point in all performance curves, we performed up to 100 independent simulation runs and calculated corresponding performance measures at the end of the simulation. In all curves 99% confidence intervals determined by independent replicates are included.

5.2 Performance Results

Before looking at PDI performance, we provide some evidence that the random waypoint mobility model applied in a simulation area of one square kilometer result in sparsely connected networks for all considered numbers of nodes. Figure 2 plots the number of partitions and the number of nodes that are in each other's transmission range (i.e., the node degree), respectively, as a function of the number of nodes. We find that even for a high number of nodes, the network comprises of about 4.8 partitions on average. Clearly, the number of partitions is highest for the lowest node density, i.e., on average about 10.16 partitions for 20 nodes. Note that the high number of partitions implies that neither an approach using a central presence server nor an approach relying on direct communication of presence information among the nodes can be applied. Fortunately, with increasing number of nodes the average node degree increases. That is, each node encounters an increasing number of direct contacts per time unit, fostering epidemic dissemination of presence information.

¹ Note that 20 nodes constitute a small community, while a community of 110 IM users on a square kilometer is fairly large.

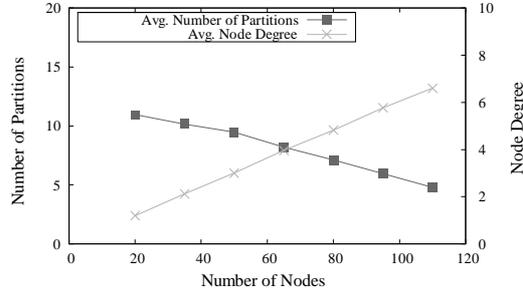


Fig. 2: Connectivity statistics for the considered scenarios.

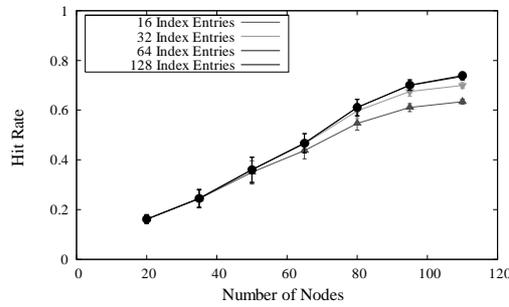


Fig. 3: PDI Hit rate for different index cache sizes.

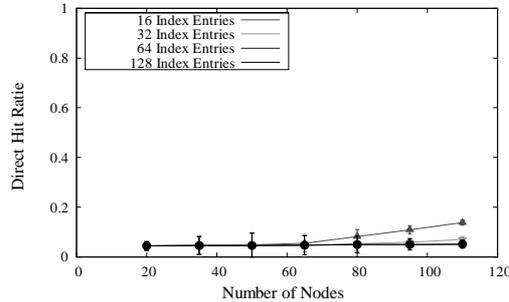


Fig. 4: Direct hits received for different cache sizes.

Thus, Figure 2 implies the conjecture that PDI is well suited to implement dissemination of presence information in such scenario.

To confirm this conjecture we illustrates the accuracy of the presence information returned by PDI in Figure 3. accuracy is measured by hit rate HR , i.e., $HR = H_F / K_F$, where H_F denotes the number of up-to-date presence information and K_F the total number of all up-to-date information currently in the system. Note that hit rate can be compared to the information retrieval measure recall. Figure 3 plots hit rate as a function of the number of nodes for different index cache sizes, i.e. 16, 32, 64, and 128 index entries. We find that for low number of nodes, the hit rate is independent of the index cache size, since all presence information fit into the smallest index cache. Nevertheless, the low node degree in such scenarios limits the epidemic information dissemination, resulting in hit rates of about 0.2. Recall from Figure 2 that node

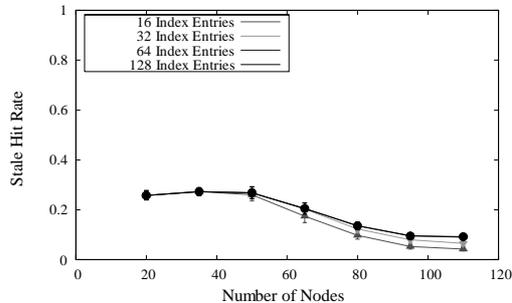


Fig. 5: Fraction of responses containing a stale presence state.

degree increases linearly with the number of nodes, resulting in a hit rate increasing up to 0.73 for 110 nodes. Note that increasing index cache size beyond 64 entries does not increase hit rate even for high number of nodes, since value timeouts limit the time an index entry is stored in a large cache [9]. We conclude from Figure 3 that PDI provides reasonable hit rates for scenarios with more than 50 nodes. To further illustrate the benefit of PDI, we plot the direct hit rate, i.e. the relative number of up-to-date hits received from the origin nodes of each reported index entry, in Figure 4. We find that direct hit rate is constant at about 0.05 for all number of nodes. Only for small index cache sizes and high node densities, direct hit rate increases due to increasing network connectivity. Recall that connectivity fosters selective forwarding. For larger index caches, selective forwarding filters out the direct hits if a nearby index cache, reducing bandwidth consumption. We conclude from Figure 4 that PDI increases hit rate by more than 14 times for scenarios with a high number of nodes.

In a last experiment shown in Figure 5, we illustrate coherence of the results returned by PDI. Coherence is measured by stale hit rate SHR , i.e., $SHR = H_S / (H_S + H_F)$, where H_S denotes the number of stale hits returned on a query. Note that stale hit rate is related to the information retrieval measure precision by $precision = 1 - SHR$. We find that stale hit rate is about 0.2 for scenarios with few nodes, which is somewhat higher than in P2P file sharing applications [9]. Nevertheless, stale hit rates drops below 0.1 for scenarios with 110 nodes, indicating that PDI will report only few stale presence states in a mobile instant messaging application.

6 Implementation of a Mobile Instant Messaging System based on PDI

Motivated by the results obtained from the simulation study, we have designed an entire mobile instant messaging systems based on PDI. The system is compliant to a proposed standard for instant messaging in the wireline Internet, namely the Extensible Messaging and Presence Protocol (XMPP Core, [14]) together with its extensions for instant messaging (XMPP Messaging and Presence, [15]). [16] defines extensions to the core XMPP protocol specifications that enable a user to establish sessions, exchange messages and presence information with other users, manage subscriptions and items in a contact list, and block communications to or from specific other users. Development of XMPP originates from the popular Jabber

instant messaging system. Thus, the user identifiers used in XMPP are denoted Jabber Ids (JIDs). A JID is of form *user@domain/resource*. Here, the jabber domain typically specifies server that manages a users contact list and relays IM communication messages on the user's behalf. The jabber resource specifies the device that the user employs to connect to the IM system. A JID of form *user@domain/resource* is denoted as full JID, whereas *user@domain* is denoted as bare JID. Full JIDs can be associated with a priority, which specifies to which resource a message should be delivered if it is addressed to a bare JID.

Note that, due to the structure of JIDs, [14] and [15] rely on one or more dedicated nodes running server software to manage user's contacts and communication. As shown in Section 5, such architecture cannot be employed in MANET. Thus, we propose an architecture consisting of one virtual XMPP enabled IM Server, which physically comprises of many distributed server entities (DSEs). One DSE is running on each mobile device participating in the IM application. To connect to the IM system, users can configure any XMPP enabled IM client to connect to the local DSE. Similar to communications between the IM client application and the DSE, the DSEs use XMPP for message-oriented communication with other DSEs. As a novel feature, DSEs use PDI to implement efficient resolution of JIDs to presence and device address information. An overview of the system is shown in Figure 6. In the remainder of this section, we show how the distributed IM server employs PDI to implement each of the features described in [15].

Session establishment: [15] requires that a user establishes a session by logging in to the XMPP server managing his jabber domain. The user is identified by his JID and a password. Similarly, in the mobile IM system, a users logs in to the DSE running on his personal mobile device, using his JID and password. Account information is stored by the DSE and password protected to ensure confidentiality when the mobile device is lost. Furthermore, account information contains a user certificate signed by a trusted certification authority. The certificate is used to authenticate and encrypt messages exchanged between DSEs.

Message exchange: According to [15], XMPP messages are transmitted using TCP connections. Similarly, the mobile IM system uses TCP connections that are established using a state-of-the-art MANET routing protocol. Before a TCP connection can be established, the DSE must resolve a user's JID to the device IP address of the terminal running the user's DSE. This is done by issuing a PDI query for a key given by the user's JID. The query is resolved to a value representing the user's terminal address. Furthermore, a PDI query is used to resolve bare JIDs to full JIDs. Note that it may happen that the PDI queries cannot be resolved, i.e., the recipient of a message is currently unavailable or unreachable. In this case, the message is stored by the local DSE. Subsequently, the DSE periodically employs PDI to poll for recipients of stored messages.

Exchange of presence information: As in the simulation study, the mobile IM system uses PDI queries to resolve JIDs to presence information. Encrypting the information with the public keys of all intended recipients ensures confidentiality of presence information. When a user logs in to the local DSE according to [15], the IM client sends initial presence information to the DSE. The DSE stores the presence state and responds to PDI queries by remote DSEs that try to obtain the presence information for the JID. When the user changes his presence state, the client notifies

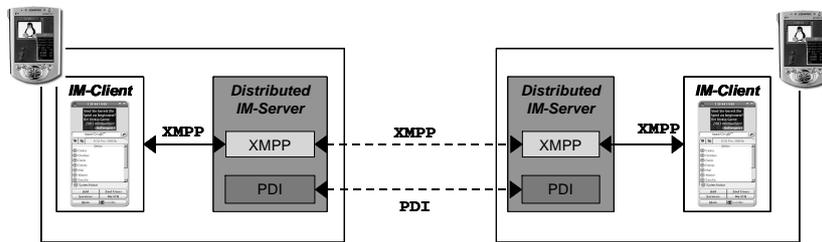


Fig.6. Architecture of a PDI-based distributed instant messaging server.

the DSE. The DSE generates a PDI invalidation message for the last presence information and responds to successive presence queries using the new presence state. Furthermore, a value-time-out implicitly invalidates old presence information. A DSE periodically polls the presence state of all contacts in the roster of a user and send presence updates to the client in case of a presence state change.

Managing subscriptions and rosters: According to [15], a user must be authenticated by another user in order to see the user's current presence state, a process denoted as subscription. Subscribing a presence state requires a two-way handshake using XMPP messages. Again, before message transmission JIDs are resolved to device addresses using PDI. Presence subscriptions are stored in rosters together with contact information. [15] requires that a user that is logged on to a server using multiple resources (devices) has a consistent view of his roster from all resources. Unfortunately, a user might be logged on with different clients on multiple mobile devices, i.e., using multiple DSEs. DSEs periodically poll for local users that are also logged on remote devices using PDI queries for users JIDs. If the user is discovered with a resource currently managed by another DSE, the DSEs synchronize the rosters using XMPP communication.

Blocking communications: [15] defines privacy lists that enable users to allow or deny users to view presence state and send or receive messages based on JIDs, servers, or resources. The current implementation the mobile IM system does not support privacy lists. However, consistent privacy lists can easily handled by the mechanisms used for roster management.

Conclusion

In this paper, we demonstrated that the Passive Distributed Indexing protocol can be applied for epidemic dissemination of presence information in mobile instant messaging systems. The main contributions of this paper include: (1) Presentation of a basic concept for using PDI for dissemination of presence information, (2) performance results that show that the presented concept provides more sufficient up to date results in a system with more than 50 users, and (3) the outline of a entire PDI-based mobile instant messaging system that is compliant to the Extensible Messaging and Presence Protocol XMPP.

We are currently developing a software prototype of the distributed IM server in Java according to the concepts described in this paper. Future work includes field-

based performance studies based on this software prototype in order to mature the design of the distributed IM server.

References

1. J. Broch, D. Maltz, D. Johnson, Y.-C. Hu, and J. Jetcheva, A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols, *Proc. 4th ACM MobiCom 98*, Dallas, TX, 85-97, 1998.
2. K. Fall and K. Varadhan (editors), *The ns-2 manual*, Technical Report, The VINT Project, UC Berkeley, LBL, and Xerox PARC, 2003.
3. S. Goel, M. Singh, D. Xu, and B. Li, Efficient Peer-to-Peer Data Dissemination in Mobile Ad-Hoc Networks, *Proc. Int. Workshop on Ad Hoc Networking (IWAHN 2002)*, Vancouver, BC, 2002.
4. M. Grossglauser and D. Tse, Mobility Increases the Capacity of Ad-hoc Wireless Networks, *IEEE/ACM Trans. on Networking* **10**, 477-486, 2002.
5. W. Heinzelman, J. Kulik, and H. Balakrishnan, Adaptive Protocols for Information Dissemination in Wireless Sensor Networks, *Proc 5th ACM MobiCom 99*, Seattle, WA, 174-185, 1999.
6. K. Hanna, B. Levine, and R. Manmatha, Mobile Distributed Information Retrieval for Highly-Partitioned Networks, *Proc. 11th IEEE Int. Conf. on Network Protocols (ICPN 2003)*, Atlanta, GA, 2003.
7. IEEE Computer Society LAN MAN Standards Committee, *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE Standard 802.11-1997, New York, NY, 1997.
8. Khelil, C. Becker, J. Tian, and K. Rothermel, An Epidemic Model for Information Diffusion in MANETs, *Proc. 5th ACM Int. Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM 2002)*, Atlanta, Georgia, 2002.
9. C. Lindemann and O. Waldhorst, Consistency Mechanisms for a Distributed Lookup Service supporting Mobile Applications, *Proc. 3rd Int. ACM Workshop on Data Engineering for Wireless and Mobile Access (MobiDE 2003)*, San Diego, CA, 61-68, 2003.
10. C. Lindemann and O. Waldhorst, Exploiting Epidemic Data Dissemination for Consistent Lookup Operations in Mobile Applications, *Mobile Computing and Communication Review (MC2R)* **8**(3), *Special Issue on Mobile Data Management*, 44-56, 2004.
11. C. Lindemann and O. Waldhorst, *Passive Distributed Indexing (PDI)*, Internet Draft (Work in progress), 2004. (forthcoming).
12. Internet Engineering Task Force Working Group Mobile Ad hoc Networks (MANET). <http://www.ietf.org/html.charters/manet-charter.html>
13. M. Papadopouli and H. Schulzrinne, Effects of Power Conservation, Wireless Coverage and Cooperation on Data Dissemination among Mobile Devices, *Proc. 2nd ACM MobiHoc 2001*, Long Beach, NY, 117-127, 2001.
14. P. Saint-Andre, *Extensible Messaging and Presence Protocol (XMPP): Core*. IETF Internet Draft, January 2004.
15. P. Saint-Andre, *Extensible Messaging and Presence Protocol (XMPP): Instant Messaging and Presence*. IETF Internet Draft, January 2004.
16. T. Small and Z. Haas, The Shared Wireless Infostation Model – A New Ad Hoc Networking Paradigm (or Where there is a Whale, there is a Way), *Proc. ACM MobiHoc 2003*, Annapolis, MD, 233-244, 2003.