COMPUTING
SCIENCE

A Multicast Service for Mobile Computing

G. Cho and L.F. Marshall

TECHNICAL REPORT SERIES

No. 474 December, 1994
A Multicast Service for Mobile Computing

G. Cho and L.F. Marshall

Abstract

Communication protocols and services are being adapted intrinsic characteristics of mobile computing. This paper identifies the main features which have to be considered when providing a multicast service for mobile computing. It then introduces the idea of group view with host status and presents four possible multicast service types with a description of how they work. By supporting some level of flexibility for disconnection and asynchronous message passing, the service will provide a good basis for building applications for mobile computing.

© 1994 University of Newcastle upon Tyne.
Printed and published by the University of Newcastle upon Tyne, Computing Science, Claremont Tower, Claremont Road, Newcastle upon Tyne, NE1 7RU, England.
Bibliographical details

CHO, Gihwan

A Multicast Service for Mobile Computing

(University of Newcastle upon Tyne, Computing Science, Technical Report Series, no. 474)

Added entries

UNIVERSITY OF NEWCASTLE UPON TYNE,
MARSHALL, Lindsay Forsyth

Abstract

Communication protocols and services are being adapted intrinsic characteristics of mobile computing. This paper identifies the main features which have to be considered when providing a multicast service for mobile computing. It then introduces the idea of group view with host status and presents four possible multicast service types with a description of how they work. By supporting some level of flexibility for disconnection and asynchronous message passing, the service will provide a good basis for building applications for mobile computing.

About the author

G. Cho is currently a PhD student in the Department of Computing Science at the University of Newcastle upon Tyne. He is working with Mobile Computing, especially for Location and Routing issues.

Dr. L.F. Marshall is a Lecturer in the Department of Computing Science at the University of Newcastle upon Tyne.

Suggested keywords

COMMUNICATION PROTOCOLS
MOBILE COMPUTING
MOBILE SUPPORT STATIONS
MULTICAST SERVICE
WIRELESS NETWORK

Suggested classmarks (primary classmark underlined)
Dewey (18th): 621.380413 384.5 621.38415
U.D.C. 621.391 654.165 658.284
A Multicast Service for Mobile Computing

Gihwan Cho and Lindsay Marshall
Department of Computing Science
University of Newcastle
Newcastle upon Tyne
NE1 7RU, England

Abstract

Communication protocols and services are being adapted intrinsic characteristics of mobile computing. This paper identifies the main features which have to be considered when providing a multicast service for mobile computing. It then introduces the idea of group view with host status and presents four possible multicast service types with a description of how they work. By supporting some level of flexibility for disconnection and asynchronous message passing, the service will provide a good basis for building applications for mobile computing.

1 Introduction

Two technological trends in computers and telecommunications today are the emerging use of portable computers and wireless communication systems. It seems reasonable that these trends will combine, and that people will carry small mobile hosts that are continuously networked via a ubiquitous wireless network infrastructure [2].

A mobile host can connect to the network from different locations at different times, so mobile computing has quite different characteristics from distributed computing using a fixed network environment. The design of communication software for such a system must therefore take into account host mobility, disconnection of mobile hosts from the network, the differing bandwidth and mechanisms between wired and wireless connections, and the broadcast and asymmetric nature of communications in a wireless network.

Disconnections are apt to happen in a mobile environment, and can even be expected to be a regular feature. Many of these are voluntary and so a mobile host can inform the system of coming disconnection in advance. This situation can be utilized in the proposed multicast service.

Typical applications of mobile computing such as mail-enabled and information services [4], are likely not to require firm replies. This is due to the low bandwidth of wireless media as well as the fragility of the mobile computing environment. Asynchronous communication has a performance benefit if no replies are needed [10]. Moreover, this advantage becomes a necessity when there are moving hosts and a wireless network.

A protocol was presented in [5] for multicasting a message to multiple mobile destinations. Despite its simplicity it generates too much message traffic, and some destinations unrelated to the multicast may do useless work. It also does not consider disconnection and the nature of applications [1].

In this paper, we consider a service paradigm that allows messages to be sent to a group of mobile hosts whose physical location changes with time. We present four possible multicast service types and
describe the details of how they work by introducing a group view with host status. The proposed service has the benefit of utilizing much of the features mentioned above by providing some flexibility at the interface level.

Section 2 discusses the design considerations. In section 3, we present four possible service types taking into consideration the mobile computing environment. In section 4, we introduce the concept of group view with host status and describe the details of how the service works for each type. Section 5 concludes the paper.

2 Design consideration

The system model used in this paper is based upon the model developed for mobile internetworking in [3]. The infrastructure machines that support the mobile hosts are called Mobile Support Stations (MSSs). These are fixed hosts connected to a fixed network with a wireless interface that provides a communication link between the mobile user and the rest of the network.

This model assumes that a static host with no wireless interface is a location server. A cell is a logical or geographical coverage area serviced by a MSS. The mobile host use its MSS address to represent location information because messages destined to a mobile host always pass through its MSS; thus, an address for a mobile host consists of the mobile host identifier and its MSS address. Each host will be permanently registered under one of the MSS - its home MSS. As a host moves, it may register as a visitor under some other MSS - its visiting MSS.

Assuming that the location information for each mobile host is available at the source MSS, it sends a multicast message over the fixed network to just those MSSs, i.e. destination MSSs, that handle at least one of the destination mobile hosts. The architecture used allows the MSS to maintain a large part of computation and related states on behalf of the mobile hosts.

Mobile computing, which includes wireless links and mobile hosts, introduces new considerations that were not present in the static network infrastructure. The features described below deserve to be considered when designing a multicast service for mobile computing. The important issue here is how we adapt the inherent characteristics of mobile computing to a multicast service.

- The addressing mechanism has to be considered so as to minimize the number of messages transmitted
- Disconnection is presumed to be voluntary, for example so as to reduce power consumption of mobile hosts
- The Wireless medium has broadcast and asymmetric communication with bandwidth constraints
- Many applications do not require a firm reply, e.g. mail, news

A number of proposals have been put forward which are concerned with host mobility [3, 8, 9]. The basic concepts underlying each proposal are similar: a division of the IP address as both a device identifier and a physical location and a mechanism to forward packets to the mobile host’s current
location. It is generally agreed that host mobility is best managed by some form of tunneling\(^1\) between the source and destination addresses.

If the source host knows the whereabouts of the destination mobile host, tunneling can occur directly with bypassing the mobile host’s home network. Therefore, the MSS which maintains a visiting mobile host has to have the autonomy to handle a multicast service; only if necessary would contact be made with the home MSS. A group view with host status is presented to achieve this.

A mobile host can across cell boundaries in the midst of data transfer. Also a message would be delivered to its home MSS rather than new visiting MSS until the sender can use new location information relevant to the moving host. So the redirecting period is inevitable in the host moving procedure. Some messages which are received in the redirecting period of Figure 1 must be guaranteed to be forwarded to the visiting MSS. In instance, the multicast service proposed will maintain a forwarding pointer for an external host.

\[
\begin{align*}
\text{(MSS}_i) & \quad \text{redirecting period} \quad \text{(MSS}_k) \\
\text{left} & \quad \text{hand-off} \quad \text{address} \quad \text{propagation} \\
\text{check} & \quad \text{hand-off} \quad \text{authorization}
\end{align*}
\]

Figure 1: Host moving procedure

Disconnection and wireless medium features might be overcome by exploiting an asynchronous message passing between the source MSS and the destination MSS. The source MSS can finish a multicast service early without gathering all replies. Then, the destination MSS has the responsibility of maintaining related status information for the mobile host, and to deliver messages to its local hosts. The group view will have a host status to handle efficiently disconnected and/or moving hosts.

With host mobility, another problem for delivering a multicast message comes from the network latency of the wired network. Copies of the same message sent over the wired network by a source may reach destination MSSs at different times. Because of network latency and host mobility, any moving host might 1) be able to receive the same message two or more times, 2) not want to receive it for a while, or 3) never receive the message.

Figure 2 shows an example. Let us consider the delivery of a multicast message sent from MSS\(_p\). The intended recipients are \(m_{h_{i1}}\), \(m_{h_{k1}}\), \(m_{h_{k2}}\) and \(m_{h_{k3}}\), and the destination MSSs are MSS\(_i\), MSS\(_k\), and MSS\(_p\). A dotted circle means broadcast delivery of the message over a MSS’s wireless cell.

\(^1\)Tunneling was named for the technique of sending packets from one part of a network to another, when the in-between routers do not know how to route the packet. This is realised by adding information to the packets so that they can traverse the part of the network that does not know how to route them.
Figure 2: Network latency and host mobility

- \( mh_{i1} \) moved to \( MSS_n \), which is not included in the destination set, before it receives the message, so cannot receive the message.

- \( mh_{k1} \) received the message at \( MSS_k \), then moved into \( MSS_i \) which had not yet received the message over the wired network, and so received the same message again.

- \( mh_{k2} \) moved to \( MSS_p \) before receiving the message at \( MSS_k \), however \( MSS_p \) had already sent the message into its local cell, so it cannot receive the message.

- \( mh_{k3} \) is disconnected for a moment, but wants to receive the message on reconnection later if it is still valid.

Message sequence numbers (\( M_{li} \)) and message logs (\( msg\_log \)) are used to resolve the above problems in the message delivery semantics. The message sequence number consists of the MSS name and a sequence number. The source MSS generates one for each service invocation, and the destination MSS uses it to provide exactly once semantics for message delivery with mobility.

The \( msg\_log \) is built from messages received by the destination MSS, and erased when all replies from destination hosts have arrived or on a timeout. This will be used to confirm the delivery of a message to disconnected or moving hosts. The \( msg\_log \) is a list of \( (M_{li}, message\text{ text, timeout, set of not-replied hosts}) \). Thus, the timeout represents a message’s life span.

3 Multicast Service

Disconnections are the most noticeable feature in a mobile computing environment and it is hard to predict their duration and frequency. However, the system should be able to utilize their voluntary nature for efficiency and flexibility. The asymmetric and low bandwidth communications of the wireless network must also be taken into consideration.

These features have to be incorporated into a multicast service that also takes into account application characteristics such as where replies to messages are not required. This can be achieved by specifying
a reply count and a timeout at the service interface level. Thus with the benefit of the user’s intuition, the multicast service will be able to take advantage of mobile computing’s characteristics. The service interface is

\[\text{mobile\_multicast (group name, message text, reply count, timeout)}\]

The reply count specifies how many replies are required to terminate the multicast service. Timeout means how long the message remains alive. From the combination of reply count and timeout, we consider four possible service types. Each type depends on how many of the service duties are moved into the destination MSS from the source MSS.

- All replies needed: A message is sent to the destination MSSs. Each of them tries to deliver it to all destination hosts, and reply to the sender with all replies from the hosts. The sender terminates the service on receiving all replies.

- Subset reply: A message is sent to the destination MSSs. Each of them tries to deliver it to all destination hosts, and reply to the sender with replies from the hosts or with a status for a disconnected host without waiting its reply. The sender terminates the service if the replies satisfy the reply count. The MSSs that maintain a msg_log will try to deliver the message to disconnected hosts during the supplied timeout. This service is well matched with the disconnection situation.

- Early reply: A message is sent to the destination MSSs. Each of them immediately replies to confirm receipt of the message. The sender terminates the service if he receives all destination MSSs’ replies. The MSS that maintains a msg_log has a responsibility to deliver the message to mobile hosts during the specified timeout. This service provides high performance for applications where guaranteed delivery is not important.

- No reply: A message is sent to the destination MSSs, the sender terminate the service with no waiting for any replies. The MSS that maintains a msg_log has a responsibility to deliver the message to mobile hosts during the specified timeout. This service provides high performance, but its usability is highly depend on network reliability.

4 Approach

4.1 Group view with host status

We now introduce the notion of a group view with host status. A group view has three main roles. Its first is to maintain MSSs and hosts which participate in the group. Second is to trace delivery history in order to resolve the message delivery semantics for moving or disconnected hosts. The third is to preserve a host status which allow destination MSSs easily to decide their own next action. This view has the advantage in that it utilities an address mechanism in conjunction with each destination MSS’s self-determination.

The group view is assumed to be static at this stage, but it can be extended easily with a membership management protocol. We also only consider closed groups, that is, only members of the group can
send to the group.

\[ (\text{group.name}) \]

\[ (\text{MSS}) : \begin{array}{cccc} \text{MSS4} & \text{MSS8} & \text{MSS5} & \text{MSS2} & \text{MSS7} \\ \text{Host} : & \text{mh3} & \text{mh4} & \cdots & \text{mh1} & \cdots & \text{mh8} \\ & \text{mh5} & \text{mh1} & \cdots & \text{mh8} \\ & \text{mh2} & \text{host.status} \{ \text{stay.in} & \text{stay.out} & \text{stay.alone} & \text{stay.visitor} \} \\ & \text{reply.flag} & \text{msg.log} : \text{list of (M.id, message text, timeout, set of not-acked hosts)} \\ & \text{forwarding address} \end{array} \]

Figure 3: MSS and host information

A group view has two kind of information: MSS and host. The view records all MSSs in the group and the local hosts for each of them. Figure 3 presents a group view at \( MSS_8 \). It should be noted that host information is different from that of the local and remote hosts. To trace message delivery, MSS information includes a \text{msg.log} and a last replied \text{M.id} on the behalf of the sending MSS. A \text{msg.log} consists of \text{M.id}, message text and timeout. A MSS increases the last replied \text{M.id} whenever it has decided to send a reply to the source MSS. This information is used when occurs message conflict.

Local host information consists of a host status and a forwarding address. Host status records the current location or disconnection status of a mobile host: \text{stay.in} for a host staying at home MSS, \text{stay.out} for a host outside, \text{stay.alone} for a disconnected host and \text{stay.visitor} for a host staying at a visiting MSS. The forwarding address points to the visiting MSS (its forwarding destination for the forwarding period) for a \text{stay.out} host, or to the home MSS (its reply destination for hand-off) for a \text{stay.visitor} host.

For remote hosts, a \text{reply.flag} is used so as to gather replies on behalf of a destination MSS. A reply from a visiting host would be sent directly to the source MSS. The sending MSS collects the replies from destination hosts and then decides the time when the service will finish. In this manner, each MSS helps a visiting host act independently of its home MSS, and the address mechanism is utilized in our multicast service.

Figure 4 shows how a group view works in the mobile computing environment. A group view has two different views, the \text{home.view} and the \text{out.view}. When a host moves out, an \text{out.view} is constructed with the host as well as MSS information of its \text{home.view}, and this is moved to the new MSS.

If a MSS gets two group views which has the \text{same group name}, i.e. \text{home.view} and \text{out.view}, \text{out.view} and \text{out.view}, these group views are likely conflict with each other. Then, host and MSS information is used for resolving group view conflicts; thus clearing up the message delivery semantics.

The resolution is simple. For each \( MSS_i \) in the \text{out.view} and the \text{home.view} at the new MSS, if the
Figure 4: Home view and out view

last replied M_id in the home view is less than one in the out view, it deletes the host from the set of un-acked hosts in the msg_log, if the last replied M_id of the home view is greater than one of the out view, it replaces the MSS information of the home view with the out view's data.

4.2 On receiving a message

Any MSS that receives a message first makes a msg_log for it. The MSS acts as following according to the service types and the host status. In our service scheme, a destination MSS has the responsibility for delivering the message to disconnected or moving hosts, so every visiting host has to reply to the source MSS regardless of whichever addressing mechanism is used to deliver the message.

Receiving from visiting host to source MSS leads to a problem, i.e. the home MSS does not know if the stay_out host has received the message. The home MSS will hold the msg_log until it is confirmed that the message was delivered to the all destination hosts which has the same group view. It is significant for two service types, early_reply and no_reply. So, a stay_visitor host has to reply to sending MSS and the home MSS as well.
<table>
<thead>
<tr>
<th>all_reply</th>
<th>stay_in</th>
<th>stay_visitor</th>
<th>stay_out</th>
<th>stay_alone</th>
</tr>
</thead>
<tbody>
<tr>
<td>broadcast</td>
<td>broadcast</td>
<td>forward to visiting MSS and wait reply</td>
<td>wait to change the host status</td>
<td>reply to sender with its status, and wait to change the host status</td>
</tr>
<tr>
<td>reply to sender</td>
<td>reply to sender</td>
<td>MSS</td>
<td>MSS</td>
<td>MSS</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>subset_reply</th>
<th>stay_in</th>
<th>stay_visitor</th>
<th>stay_out</th>
<th>stay_alone</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MSS</td>
<td>MSS</td>
<td>MSS</td>
<td>MSS</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>early_reply</th>
<th>stay_in</th>
<th>stay_visitor</th>
<th>stay_out</th>
<th>stay_alone</th>
</tr>
</thead>
<tbody>
<tr>
<td>reply to sender</td>
<td>broadcast</td>
<td>MSS</td>
<td>MSS</td>
<td>MSS</td>
</tr>
<tr>
<td>MSS</td>
<td>MSS</td>
<td>MSS</td>
<td>MSS</td>
<td>MSS</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>no_reply</th>
<th>stay_in</th>
<th>stay_visitor</th>
<th>stay_out</th>
<th>stay_alone</th>
</tr>
</thead>
<tbody>
<tr>
<td>broadcast</td>
<td>broadcast</td>
<td>MSS</td>
<td>MSS</td>
<td>MSS</td>
</tr>
<tr>
<td>MSS</td>
<td>MSS</td>
<td>MSS</td>
<td>MSS</td>
<td>MSS</td>
</tr>
</tbody>
</table>

### 4.3 Hand-off procedure

A hand-off procedure is invoked on moving a host into other cell. A mobile host would be able to move into a MSS which has a group view, a MSS which does not have a group view, or return back to its home MSS. Each case is treated as follows:

<table>
<thead>
<tr>
<th>initiator</th>
<th>responder</th>
</tr>
</thead>
<tbody>
<tr>
<td>stay_in</td>
<td>if responder has a group view, resolve the group view</td>
</tr>
<tr>
<td>build up an out_view with stay_visitor, set the home_view with stay_out, hand over the out_view to new MSS</td>
<td>else, if any msg_log not-replied from the host, broadcast, reply to the home MSS</td>
</tr>
<tr>
<td>stay_visitor</td>
<td></td>
</tr>
<tr>
<td>if return back to the home MSS, set an out_view with stay_in, hand over the out_view to new MSS</td>
<td>resolve the group view</td>
</tr>
<tr>
<td>else, hand over the out_view to new MSS</td>
<td>if responder has a group view, resolve the group view</td>
</tr>
<tr>
<td></td>
<td>else, if any msg_log not-replied from the host, broadcast, reply to the home MSS</td>
</tr>
</tbody>
</table>

### 4.4 On receiving a re-connection request

A disconnected host could request its re-connection anywhere. If the requester is a local host and the MSS has a group view, after setting the host status to stay_in, if there is a msg_log not replied to by the host, broadcast it. If the requester is not a local host and the MSS has a out_view, request a hand-off procedure to its home MSS, and if the MSS has not any group view for the host, request
a hand-off procedure to its previous visiting MSS.

5 conclusion

In conclusion, we have proposed four possible multicast service types based upon the concept of group view with host status which best serves the mobile computing environment. Also the proposed service has the benefit of utilizing of most of the inherent features of mobile computing by providing some flexibility at the interface level. As it can take advantage of disconnections and asynchronous message passing, the service would provide a good basis for building applications for mobile computing. Nevertheless, we still feel that an evaluation of performance is necessary in addition to a formal validation of the protocol.

References


