ATM Multicast in Mobile Environment using Hierarchical Servers

Sudha Ramesh
Telcordia Technologies (formerly Bellcore)
Red Bank, NJ

1. INTRODUCTION

In multicast, there are two problems to be considered: identification of a group of users participating in multicast and establishment of virtual connections for the flow of traffic. In the accepted ATM standards [1], [3], [8]-[9], there is no definition of an address format that can represent all the destination users[1]. There has been no serious consideration given to the support of true multipoint-to-multipoint connections in ATM. The schemes used for multicast must involve point-to-multipoint connections. The common approaches to multicast in ATM are the Virtual Channel (VC) Mesh and use of Multicast Servers, both of which have limitations. With the traditional Multicast Server approach, the single server maintains and manages the group membership information and can become the bottleneck. There is a single point of failure. The scalability issues and other limitations suggest we should use multiple servers.

We have extended the concept of multicast servers by arranging a set of multicast servers hierarchically to exploit the natural hierarchy of tactical networks. We propose the use of composite parameters to identify multicast sessions. We address mobility issues in tactical networks that go beyond the one-hop user/network mobility addressed by the WATM working group of the ATM Forum [5]. We have designed the Hierarchical Multicast Server (HMS) Protocol which defines the control messages exchanged among servers and between users and servers. This paper is a description of the hierarchical architecture and the procedures of the protocol.

2. HIERARCHICAL MULTICAST SERVER ARCHITECTURE

The hierarchy of the multicast servers may be mapped to the administrative hierarchy, some geographical partitioning of the network or any other partitioning of the network. The term "server" is used instead of "multicast server" in many places in this paper.

The servers are addressable entities that are able to originate and terminate switched virtual connections. The servers at all levels may serve ATM users, each of which has a default multicast server with which it registers. A server or user maintains a bidirectional point-to-point control path to its parent server for exchange of control messages. In addition, there may be a unidirectional point-to-multipoint control path from a server to its descendants which is used if the same message is to be sent to all descendants. These control paths may be virtual channels setup in the user plane and used to transport control signaling messages. Another approach is to use the connection oriented bearer independent transport service of the generic transport services [10], if implemented. The hierarchical multicast servers are illustrated in the following diagram in Fig. 2-1. The links shown represent the control paths and are not necessarily one-hop links. In subsequent sections, for the sake of simplicity, we will not show ATM users at the higher levels of hierarchy.

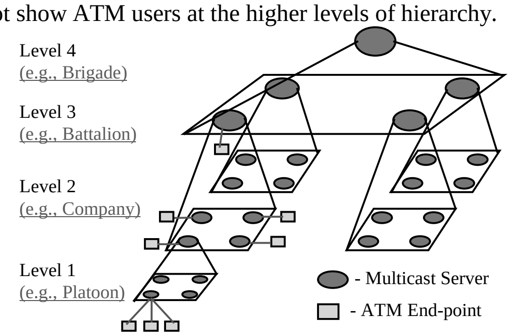

Fig. 2-1. Hierarchical Structure of Multicast Servers

The principal idea is that the multicast groups of interest resemble some sort of hierarchy and a server is responsible for some multicast groups. For example, in the tactical network, if some information is to be disseminated to everyone in a brigade, we could view this as distributing the information to everyone in each of the battalions in this brigade.

We desire the multicast to be represented by a tree of multicast servers and not a forest, for several reasons. First of all, it is desirable that the traffic from any particular source be carried to a server along just one virtual connection. In a forest of servers we need to setup multiple connections from each source to the roots of the different trees in the forest. Secondly, since the roots of the different trees of a forest are likely to be at the same level, we can always consider a server at the next higher level to be the root of the transmission and the other servers at the same level who do not want to participate in the group can be deleted from the multipoint connections setup. Thirdly, with multiple sources, the AAL frames for the different sources have to be sequentialized at the root of a multicast transmission. With a forest, this will have to be done at many server roots.

3. IDENTIFICATION OF A GROUP

Each multicast session, also called "group" (analogous to IP groups), is identified by an identifier and a scope and is denoted by a tuple, (G_i, G_s). The identifier, G_i, is similar to the group address in IP and may represent some kind of service. G_s, defines the scope for the group in the hierarchy and identifies the multicast tree in the hierarchy. If the scope is different but the identifier is the same, it may refer to the same service in a different domain.

Our scope encompasses three parameters: the level of the highest server, the level of the lowest servers and some identification of the extent of the highest server's influence. The lowest level for the servers is useful when the transmission need not go all the way down to the bottommost server. The third parameter is used to identify a server within the highest level. It may be something like saying "the 3rd brigade"; identifying one of the brigades. Within the

This paper is the result of work done under the Contract DAAB07-98-C-D007 from U.S.Army CECOM.

[1] ATM anycasting addresses identify one of several members of a group and not all members, as we would like in multicast.

scope, there may be place for other parameters (e.g., security and authentication parameters). But a discussion of other parameters is beyond the scope of this paper.

4. OVERVIEW OF HMS PROTOCOL

In the rest of the paper, we discuss procedures defined in the HMS protocol. Each subsection describes an aspect of the protocol. The control messages used for these various functions are mentioned in these sections. Detailed description of the encodings for these messages and their sub-units is beyond the scope of this paper. Messages used for query, registration, deregistration, and join operations in the following text are all timed and may be retransmitted several times.

4.1 FORMATION OF THE HIERARCHICAL STRUCTURE

One possibility for forming the hierarchical structure of the servers is to manually configure the structure. This is not an appealing approach, especially in a dynamically changing environment like the tactical network. The servers, serving the different units of the network, are either elected or selected based on some other kind of configuration procedures. We use an election scheme which is like most election algorithms.

ATM users know their level in the network. These users can either use auto-discovery procedures to determine their default multicast servers or be configured with the addresses of such servers. The auto-discovery procedures may employ ILMI [2] procedures similar to those described in [6] and [7] or the ATM Forum's alternative procedures, being developed as part of the PNNI Augmented Routing (PAR) specifications [4]. Once the address of the default server is known, the ATM user proceeds to setup a bidirectional point-to-point control path to this server. Upon successful establishment of the control path, it sends a USER-REGISTER message to the server over this control path, the response to which from the server is the USER-REGISTER-RESP message indicating the success or failure of the registration. From this point on, the identified server serves all the ATM multicast needs of the user.

When servers boot up they know the group addresses (including identifiers and scope) that they are responsible for. If they know of their direct descendant users, they can setup the point-to-point control paths to these users. Otherwise, they can wait for the users to register with them. A server determines its parent server using some auto-discovery procedures, sets up a point-to-point control path to it and sends a SERVER-REGISTER message that includes a list of all the groups served by this server and a list of all its descendant servers. When the parent receives this message, it adds this server to its list of direct descendant servers, adds this server to the point-to-multipoint control path, updates its list of descendant servers and the groups served by the descendants, and sends a SERVER-REGISTER message to its parent containing the received information. The message is progressed all the way to the top of the hierarchical structure. Before progressing a SERVER-REGISTER message to its parent server, a server may accumulate information from more than one of its descendant servers and send a single SERVER-REGISTER message. A simple example of this is illustrated in Fig. 4-1, where the sequence of events is labeled (1) through (6).

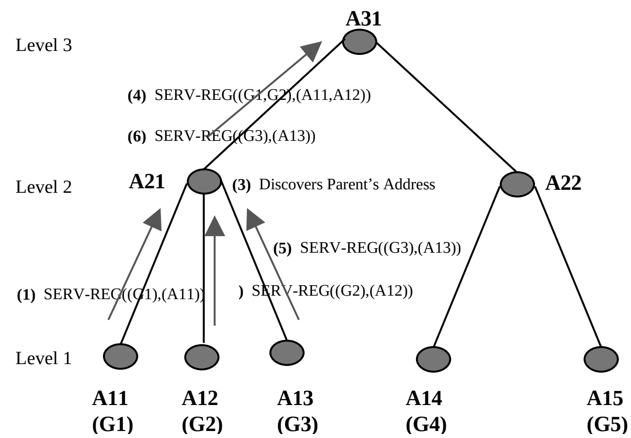

Fig. 4-1 An Example of Server Registration

4.2 REGISTERING GROUPS

The servers, on boot up, may know the groups they serve. But this may dynamically change. To handle the dynamic changes, the GROUP-REGISTER and GROUP-DEREGISTER messages are used. The GROUP-REGISTER message is sent from a server which gains control for a group (G_i,G_s) to its parent server. The response to this message is GROUP-REGISTER-RESP. When a server no longer serves a group, it sends a GROUP-DEREGISTER message to its parent. The parameters in this message are the same as the ones for the GROUP-REGISTER message. The response to this message is GROUP-DEREGISTER-RESP. The REGISTER and DEREGISTER messages are propagated up the hierarchy.

4.3 ESTABLISHING MULTICAST SESSIONS

In order to establish a multicast session, there are three basic problems: 1) locating the highest level multicast server for this group, called the "group server" 2) communicating a request to establish the multicast session, and 3) establishing virtual connections for data transmission.

Locating the Group Server:
There are a couple of ways of finding the group server, which is the root of the multicast tree. If the user is registered with a default multicast server, it can send the query message, MC-QUERY, to locate the group server for group (G_i,G_s), over its control path. The server replies to this message with the MC-QUERY-RESP message containing the ATM address of the group server, A_{serv} as well as an indication of the success or failure of the request. The default server can directly answer the query if the server or one of its descendants serves the group. If not, the query message is propagated to its parent server and the response received is then relayed to the user. The message flow for this query is illustrated below in Fig. 4-2 where the server S2 responds to the query with the server address, A_{serv}, which is not necessarily the address A_{S2} of S2, but can be the address of one of its descendants. The addresses A_{s1} and A_{user} in the messages shown are used to route the messages correctly.

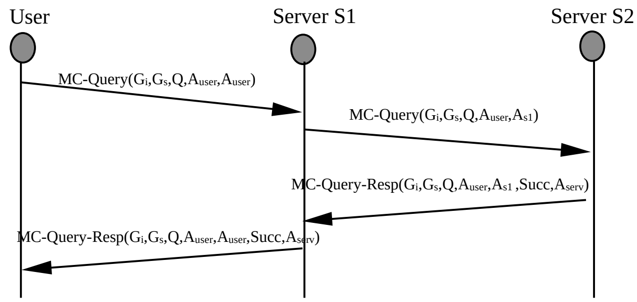

Fig. 4-2 Flow Diagram for Server Discovery

Another way of locating a multicast server for a multicast group is to use the service registry MIB in the ILMI [2] as in some auto-discovery procedures.

Requesting the Establishment of a Multicast Session:
Once the group server is identified, the source in a multicast session sends the request to join using the MC-JOIN message over its control path. If there is no connection to the group server via the default server, it may establish a temporary point-to-point control signaling association with the group server. The MC-JOIN message propagates up and down the hierarchy to reach the group server and the group server replies with the MC-JOIN-RESP message, which indicates the success or failure of this join request. This is possible because the servers maintain a list of groups served by the descendant servers. The response message travels the same control paths back to the source of the request. The reply also includes a parameter that identifies the point-to-point call in the user-plane that the group server establishes to the source, which is used for multicast transmission from the source to the group server.

User Plane Connections for the Multicast Session:
If there is already an active multicast session for this group, the group server sets up a point to point connection in the user plane to the source, sends an MC-JOIN-RESP message back towards the source identifying this call and adds the source to the list of sources. There are no more user-plane connections to setup. The server distributes traffic from the new source using the already established virtual channels for the multicast transmission.

If there is no active multicast session for the specified group, the multicast group server proceeds as follows to setup the multicast session:

1. It adds the source to the list of sources, sets up a point to point connection in the user plane to the source, and returns an MC-JOIN-RESP message towards the source identifying the point-to-point call just established.
2. It sets up a point-to-multipoint call in the user-plane to all the ATM users registered with it, links this call with the point-to-point call it setup with the source in Step 1 above, and sends MC-SER-JOIN message to these receivers. If the group server is at the lowest level specified, then the establishment of the multicast session is complete. Otherwise the following is done.
3. It adds its direct descendant servers to the call setup in Step 2, if there was one, or it sets up a new point-to-multipoint call to all its direct descendant servers linking the new call just setup with that in Step 1. It sends the message MC-SER-JOIN identifying the multipoint call, to all its descendant servers.

A descendant server does the following:

1. Upon receiving an MC-SER-JOIN message, if it is the lowest level server, it sets up a point-to-multipoint call in the user-plane to all the ATM users it serves, links the incoming call to the new call it just setup, and sends MC-SER-JOIN message to these receivers. This ends the chain of point-to-multipoint calls used for the multicast session.
2. If the server is not the lowest level server for this group, it does the following:
   a) It sets up a point-to-multipoint call in the user-plane to all the ATM users registered with it, links this call with the call indicated in the MC-SER-JOIN it received, and notifies these receivers by sending the MC-SER-JOIN message.
   b) It adds its direct descendant servers to the call setup in Step 2a, if there was one, or sets up a new point-to-multipoint call to its direct descendant servers, linking the new call with the call indicated in the MC-SER-JOIN it received. It sends the control message MC-SER-JOIN to its direct descendant servers identifying multipoint call.

An example message flow is illustrated in Fig. 4-3. Here, the group server is A32 (also labeled G) and the source depicted is the first source to contact the group server. The MC-JOIN message is progressed towards G through the servers A31, A41 and A32. The server A32 sets up a point-to-point user-plane connection to the source and then proceeds to setup the connections down the tree. The MC-SER-JOIN messages are carried over the point-to-multipoint control path and the MC-SER-JOIN-RESP messages are sent over the point-to-point control paths. In this diagram we have not shown the users at intermediate levels of the hierarchy, for the sake of simplicity.

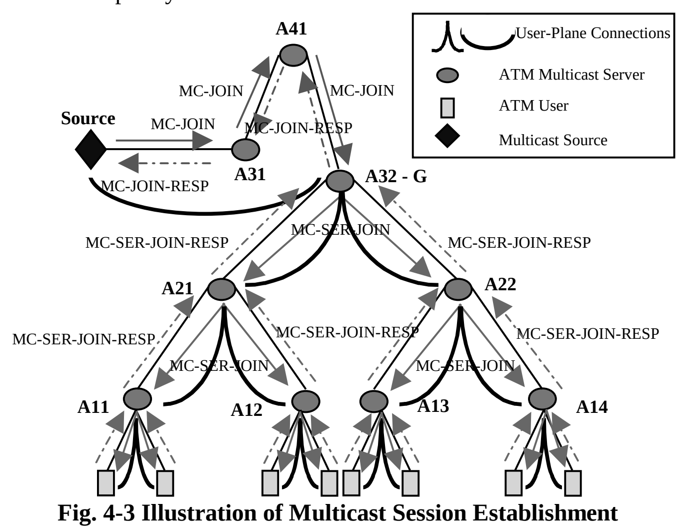

Fig. 4-3 Illustration of Multicast Session Establishment

It is clear that if there are *n* levels of multicast servers, then the ATM cells from a source go through *(n+1)* calls in the user plane (in a traditional multicast server approach there will be two calls). Therefore, this scheme is most useful when the depth of the tree involved is small, which is likely most of the times. If many levels are involved, it is more like a broadcast (or a limited broadcast) and in such cases some delays are tolerated. The depth of the tree involved is determined by how the hierarchy is established. If there is a clear pattern in which groups are allocated, we may arrange the hierarchy to minimize the depth of the structure.

The number of direct descendant servers and the ATM uesrs registered limits the number of virtual channels used at any server. Therefore, it is unlikely that at any server, the limit of virtual channels, imposed by the hardware, is exceeded. This is one of the advantages of this scheme.

Receiver Join:
When a receiver is within the domain of the group server, it does not have to do anything extra to join the multicast session. In case a receiver has moved or if it had not participated when the channels were originally setup, a receiver may join a group using the MC-RX-JOIN message, which is propagated towards the group server until a server that knows of this group is reached. That server adds the receiver to the group. Otherwise, the group server receives this message and adds it to its user-plane connection, or informs the user to join somewhere else, or rejects the request (especially if there is no active session and/or the receiver is outside the scope for the group). The response message MC-RX-JOIN-RESP contains all the necessary information.

Any other ATM user (which will be authenticated) may serve as a proxy agent and add a receiver to a multicast session. In that case the proxy sends the MC-RX-JOIN message to the group server.

5. MOBILITY

This section discusses location management and mobility of sources, receivers, servers and switches in a multicast session.

5.1 LOCATION MANAGEMENT

The location of a user or a server need only be known to its parent and hence location management becomes a localized operation. When a user/server changes its attachment point, it is likely that its address changes. We expect the location management functions defined in the WATM specification [5] will handle these and other mobility functions. Our concern is its association with the server.

If a direct descendant, an ATM user or a child server, moves and its address changes, there are two possibilities. After movement, if the parent server is still the same, after changing its point-to-point control path to the parent (if not already done as part of the WATM procedures), the user/server sends USER_ADDR-CHANGE or SERVER-ADDR-CHANGE to inform the parent of the address change. When SERVER-ADDR-CHANGE is used, the information will be propagated up the hierarchical structure.

The second possibility is that it moves outside the scope of the old parent server into the domain of a new server, in which case, it has to establish a point-to-point control path to the new parent server and register with the new parent using the registration procedure detailed earlier. The control path to the old server is either torn down or lost during the move. In either case, upon the loss of the virtual channel, the old server knows that it no longer serves the node that just moved and deletes this node from its database. The impact of movement on the server assignment to nodes is minimal.

5.2 SOURCE AND RECEIVER MOBILITY

When a source moves, the only portion of the multicast session affected is the user-plane point-to-point call between the source and the group server and the control path to its default server or the group server. The call in the user plane can be extended from the previous location, completely re-established or rerouted, as appropriate, as part of the WATM handover specifications. The control signaling association is also re-established with its default server. This is the advantage of using a multicast server scheme, as opposed to a VC mesh where rerouting or changing the many point-to-multipoint calls will be cumbersome, to say the least.

The receiver mobility is more involved than source mobility and falls under one of the following three cases:

1. The receiver can move within the domain of its immediate predecessor server, or
2. The receiver can move within the domain of some predecessor server, which is not the immediate predecessor (parent), or
3. The receiver can move completely outside the domain of the group server serving this group.

In all cases the control path is handled the same way. If a user-plane connection is used for the control path, it can be handed off using WATM procedures. If a generic transport mechanism is used, the association is re-established.

We have worked out the details for handling the three scenarios. But without getting into too many details, the procedures involve, re-routing, partial clearing and re-establishing of the connection, or path extension. We use the handover procedures of the WATM specifications whenever possible. Otherwise, we use the receiver join procedures using the MC-RX-JOIN message. In some cases, we also advocate attempting both path extension and receiver join procedures and keeping the one that succeeds first.

The second and third scenarios are illustrated in Fig. 5-1.

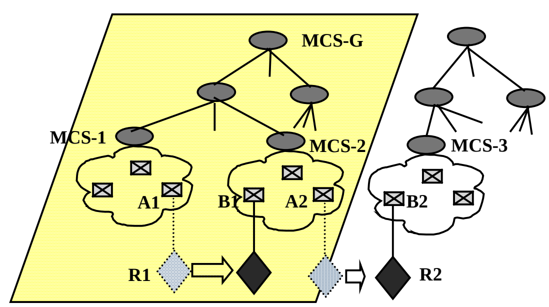

Fig. 5-1 Illustration of Receiver Movement within and outside the domain of the Group Server

In Fig. 5-1, the shaded area represents the scope of the group server labeled MCS-G. In case 2, the receiver R1, changes its attachment point from switch A1 to switch B1, thereby changing its default server from MCS-1 to MCS-2. In case 3, the receiver R2 moves outside the domain of the group server MCS-G. Here, if R2 is no longer interested in participating in the multicast, it can simply clear the connection to the old multicast server MCS-2. If R2 decides to continue participating in the multicast session it can employ one of the above mentioned methods or, if a similar service is offered in its new domain, R2 can join such a service in its new place, via its new default server MCS-3.

5.3 Server and Switch Mobility

When a switch or a server moves it may affect more than just the node which moves. With the switch the effects can be considered in terms of the disruption of a link between a server and one of its descendants. With the server, there can be three situations. The server may move without affecting the hierarchy, the server may move along with its subtree into a new domain or the server may move by itself without its descendants. We have worked out the details of these various scenarios. These cases can be simplified in this paper in terms of the movement of a user or a server, connecting to a new parent server, the actions of the previous parent, and the actions of orphaned nodes.

When a server moves resulting in no change to the hierarchy, the control-paths to its parent and descendants and the user-plane connections for active groups have to be re-established or rerouted using WATM procedures. If upon movement, the address changes, the SERVER-ADDR-CHANGE message will be propagated up the hierarchy.

If a server finds that its parent server has changed, the server will send SERVER-DEREGISTER message to the old parent (if that control path is still open) and register with the new parent. For active groups which involved the old parent, all user-plane connections are torn down. All other user-plane connections are maintained. In addition, if the server is no longer responsible for its previous domain, all user-plane connections to the descendants are disconnected and the descendant information removed. The new parent server, upon the receipt of the SERVER-REGISTER message, adds this server to its descendants, updates its control paths to include the new descendant, and scope permitting, for all active groups, adds this server to the active multicast session.

The previous parent, which lost a child server, removes the server from its descendants, updates all information, and sends a SERVER-DEREGISTER up the hierarchy indicating the server it just lost. If the control paths to the server are not already torn down (because the child server may have already done it), it removes the control paths and affected user plane connections.

The only other kinds nodes affected are the orphaned server or user. The orphaned user will clear all connections, determine a new default server and register with the new one. The orphaned server, will clear only those user-plane connections which are for groups which came from the parent.

Although, in the above discussion we propose clearing of connections, the detailed procedures involve a recovery period when a server (parent or child) can come back up. The control paths are cleared only after this period. We also try to use the user-plane connections for other active groups in the new location. This is true if the descendants are still the same and only the parent changes.

6. SUMMARY AND CONCLUSIONS

In this paper an overview of the Hierarchical Multicast Server (HMS) protocol used for ATM multicasting is presented. A description of the architecture of servers, how the hierarchy is formed, establishment of multicast sessions and mobility of various components of the architecture is presented. In the current ATM standards, there is no comprehensive solution to the ATM Multicast problem. Our proposal, represents a workable solution in a hierarchical setup suitable for the tactical network. In this proposal, we have used the solutions being developed by the WATM working group of the ATM Forum and addressed issues that are not yet considered by the standards bodies. We feel that this solution can be implemented in the networks, because we do not need to replace other standardized protocols but add to those.

We are currently working towards enhancing the protocol design by incorporating more receiver-oriented concepts into the design. We are also working on an implementation of this protocol in our wireless testbed.

ACKNOWLEDGMENT

The author expresses great appreciation to Charles Graff and Michael Bereschinsky (of U.S.Army CECOM) for their continued support and active involvement in this work. Their knowledge, especially on tactical networks proved invaluable in this project. The author had many interesting discussions with them and John Strange (also of U.S.Army CECOM) throughout this project.

REFERENCES

[1] ATM Forum, ATM User-Network Interface (UNI) Specification, Version 4.0, July 1996.

[2] ATM Forum, Integrated Local Management Interface (ILMI) Specification, Version 4.0, Sep. 1996.

[3] ATM Forum, Private Network-Network Interface (PNNI) Specification, Version 2.0, Sep. 1997.

[4] The ATM Forum, PNNI Augmented Routing (PAR) v1.0 Specification, STR-RA-PAR-01.06, October 1998.

[5] ATM Forum, WATM Working Group Document Number: ATM_Forum-BTD-WATM-01.08, “Draft Baseline text for Wireless ATM Capability Set 1 Specification,” July 1998.

[6] M. Davison, “ILMI-based Server Discovery for NHRP,” draft-ietf-ion-discov-nhrp-01.txt, Internet draft, May 1997.

[7] M. Davison, “ILMI-based Server Discovery for MARS,” draft-ietf-ion-discov-mars-01.txt, Internet draft, May 1998.

[8] ITU-T Recommendation Q.2931, “Broadband Integrated Services Digital Network (B-ISDN). Digital Subscriber Signalling System No. 2 (DSS 2). User-Network Interface (UNI) Layer 3 Specification for Basic Call/Connection Control,” 1995.

[9] ITU-T Recommendation Q.2971, “Broadband Integrated Services Digital Network (B-ISDN). Digital Subscriber Signalling System No. 2 (DSS 2). User-Network Interface (UNI) Layer 3 Specification for Point-to-Multipoint Call/Connection Control,” 1996.

[10] ITU-T Recommendation Q.2932.1, “Broadband ISDN – B-ISDN application protocols for access signalling). Digital Subscriber Signalling System No. 2 – Generic functional protocol: Core functions,” 1996.