Network Working Group
Internet-Draft
Expires: August 25, 2008

J. Lei
X. Fu
D. Hogrefe
Univ. Goettingen
February 22, 2008

DMMP: Dynamic Mesh-based Overlay Multicast Protocol
draft-lei-samrg-dmmp-03.txt

Status of this Memo

Copyright Notice

Abstract

This document describes a Dynamic Mesh-based overlay Multicast Protocol (DMMP) framework to support multicast data delivery applications without relying on classic IP multicast, including multicast group management, overlay hierarchy establishment, multicast tree construction and data forwarding scheme from a source to a number of receivers. The DMMP framework builds on control plane functions which dynamically manage an overlay core and a multicast

tree. The key idea is a number of end hosts self-organize into an overlay mesh, and dynamically maintain such a mesh. Based on the constructed mesh, some core-based clusters are formed with capacity-aware trees inside. Then, a multicast tree consisting of DMMP-aware end hosts (and/or specific routers) is built on the top of the overlay core for the efficient delivery of the multicast data.

Table of Contents

1. Introduction

Over the recent years, a lot of research efforts have been focusing on moving multicast support out of the network core, since the deployment of network layer multicast has been obtructed by both technical and opertional issues [3], [4]. To solve these issues of IP multicast, various application level multicast approaches have been proposed, which can be largely summarized into two categories, namely, application layer multicast (ALM) and overlay multicast (OM). As a matter of fact, network layer multicast requires changes in IP routers, while ALM and OM approaches rely on network unicast and does not need network layer infrastructure support from intermediate nodes (e.g. router).

In ALM approach, end hosts form a virtual network, and multicast delivery structures are constructed on top of such a virtual overlay. A basic ALM approach is to form and maintain an overlay for data transmission, where all end hosts in a multicast session are involved without considering the heterogeneities of them, e.g. computation power, bandwidth and access possibilities. For instance, all end hosts join the full mesh construction of ESM (Narada) [5] and multiple connections exist between any two nodes. The main advantage of constructing such a mesh is easy implementation and being relatively stable. However, ESM's sole dependence on the mesh structure results in that it could only be applied well into a small or medium-sized group [6]. NICE [7], in contrast, introduces a hierarchical management scheme to create a scalable ALM overlay. This hierarchical design simplifies the membership management of the application layer multicast and makes it scale better than the full mesh-based structure. Nevertheless, the joining procedure in NICE causes a very high control overhead, which not only limits the scalability of deployment, but also is likely vulnerable to single node failures (e.g. possible failures caused by the node at the highest layer of hierarchy). As described above, ALM approaches address some practical/-deployment issues in network layer multicast but there is a general concern about its efficiency and scalability.

Observing the weaknesses from ALM approaches, an alternative approach, i.e., overlay multicast or OM, by using a kind of "infrastructure-based" solution, has been proposed to improve multicast efficiency and maximize the resource usage (e.g, bandwidth). Proposals of such an approach include OMNI [8] and TOMA [9]. The design issues of OM can be summarized in the following two aspects: On one hand, OM approaches employ some fixed or long-term infrastructure-based nodes to simplify membership management and multicast tree construction. This advantage can become a weakness since the assumption of these fixed nodes in the infrastructure limits the extensibility and flexibility of deployment. For example,

the infrastructure must be re-established based on other long-term measurements before constructing a new multicast trees to adapt to the requirements imposed by a different metric. On the other hand, TOMA and OMNI need dedicated infrastructure deployment and costly servers, which could not be adaptive to dynamic network changes and group member changes such as new members join. Therefore, it is relatively difficult to implement them into the current Internet environment although they are proposed to provide multicast support for group communication applications. Obviously, to develop a practical, efficient and resilient multicast framework is the essential way towards wide deployment of multicasting services.

Additionally, the explosive growth of multimedia services and applications over Internet necessitates streaming media to a large population of users. However, with current media streaming technology, it's hard to develop a comprehensive on-demand media streaming system due to the following two challenges [10]. First, the total number of concurrent clients the system can support is limited by the resources of the streaming supplier. Second, current media streaming proposals usually have limitations in reliability and scalability. The reliability concern arises from the fact that only one entity is responsible for all clients. The scalability issue arises from the fact that adding internet-scale potential users requires the commensurate amount of resources to the supplying server. Meanwhile, aforementioned proposals could not explicitly support real-time media streaming applications in a large scale.

Motivated by above studies, in this draft we present a new overlay multicast framework which manages a dynamic mesh-based overlay core

and only involves participating end hosts without relying on the availability of the OM-aware infrastructure nodes, while providing certain degree of efficiency, reliability and resilience.

2. Features of DMMP

DMMP is organized into a two-level hierarchy and the mechanisms of DMMP are introduced to dynamically manage and maintain the hierarchy. The key idea behind DMMP is to let a few end hosts selected and self-organize into an overlay mesh during the multicast initialization phrase and also when group member changes, and dynamically maintain such a mesh. Although routers may also be manually designated (e.g. by ISPs) to construct the mesh, this document initially discusses the approach via end hosts. Specifically, there are three design issues to be addressed in DMMP:

- Host heterogeneity - Previous research has shown that a large proportion of free-riders (i.e. group member who can only receive data from incoming sessions) may exist in the network [11], [12]. DMMP considers the heterogeneous capacities of group members by evaluating their available bandwidth during the runtime. In the DMMP framework,it is possible that only a small number of high-capacity end hosts are selected to construct the overlay mesh when there are a large proportion of free-riders [13]. This operation may help maximizing the usage of available bandwidth for the overlay tree.
- Scalability - Scalability is one of the main problems to be solved in the multicast applications. In DMMP, each end-host may act as a potential server for other clients and the number of possible servers increases at the same rate as the end host clients. As peer to peer (p2p) technologies have been deployed to support various services over the Internet, it is possible that more end hosts resources are available in the network. Once a node joins the DMMP multicast session, additional resources are available to the whole system. Therefore, the DMMP system is scalable as it can potentially support a number of clients.
- Delay optimization - DMMP considers end-to-end delay for end hosts. When forming the local cluster, non-super nodes select the

nearest super node in term of e2e delay measurement. This aspect is essential while supporting delay sensitive applications (e.g. media streaming). Furthermore, high-capacity nodes are given high priority to stay at the higher level of the tree when constructing the overlay multicast tree. In return, it allows to produce the tree as short as possible and hence the overall delay could be reduced.

- Resilience to dynamic changes - DMMP considers the transient nature of end hosts and tries to prevent incapable or short-lived nodes from staying close to the center of the multicast tree. Consequently, the DMMP overlay structure is relatively stable and resilient to dynamic network changes. Moreover, network situation changes (such as multicast members joining/leaving) within a local cluster will not have any impact on other clusters. The failure of a single node may result in a transient instability in a small subset of participants, but it will not cause a catastrophe in the whole overlay.

In order to overcome the two challenges in Section 1, media streaming task in DMMP is accomplished through the following two phases: (1) An on-demand overlay core(or mesh) is established to achieve the optimized performance; (2) Based on the structured mesh, several clusters are formed to connect with selected mesh members, namely, super nodes. Here, DMMP applies the concept of locality (e.g., clusters) into the group management so that it can dramatically reduce the control overhead and the complexity of the overlay

maintenance. Basically, a source-based DMMP architecture consists of a sender, several receivers, one or many Rendezvous Points (RPs) and Domain Name Systems (DNSs). Compared with existing application level multicast approaches, DMMP is designed to be more stable, efficient and applicable to support large-scale groups without relying on predetermined intermediate nodes in the network and potentially get better performance. Note that DMMP currently considers source-specific multicast [1], any-source multicast is left for future studies.

3. Terminology and Abbreviations

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and

"OPTIONAL", in this document are to be interpreted as described in BCP 14, RFC 2119 [2]. Other terminologies and abbreviations used in this document are used as follows:

- Overlay Multicast - A multicast data delivery scheme depending on end hosts to form an overlay core for message control and a multicast tree for data delivery.
- Rendezvous Point (RP) - A designated node for a multicast group, which assists managing group members and stores some required information (e.g. performance required).
- Source - The multicast service sender. It could be a video stored server or some video distributed servers in one service domain, which delivers the data traffic to the source-based multicast group members; DMMP in this document only provides the source-specific mechanism to realize the single source-based overlay multicast.
- Super nodes - Some end hosts are chosen to manage the multicast group and relay data from the mesh to receivers within clusters. Currently, only end hosts can serve as super nodes; future version of this document may specify the case when some routers (e.g., first-hop routers) are used as super nodes.
- Receivers - Multicast group members who want to receive the data from the source.
- Mesh - An overlay core, which is responsible for group member management and multicast tree configuration.
- Clusters - Based on each super node, end hosts organize themselves into a core-based multicast tree. [14].
- Stress - A tree metric that counts the number of identical packets sent by the protocol over a single link or a single node.
- Out-degree - Available connections, namely, the available number of connections that a node can establish.

- Uptime - The time duration from a node joining in a multicast session to its leaving the multicast session.

Within each cluster, there are some terminologies and abbreviations which are used as follows:

- Parent - the direct upstream node of a node is called the parent of that node, e.g. end host 1.2 is the parent of end host 1.2.1 in

Figure 1.

- Parent level nodes (PLN)- nodes (exclusive parent) at the same level as the parent of some node, e.g. 1.1 and 1.3 are parent level nodes of 1.2.1 in Figure 1.
- Child - the direct downstream node of some node, e.g. in Figure 1, 1.2.1 is the child of 1.2.
- Children level nodes (CLN)- nodes (exclusive children) at the same level as children of some node, e.g. 1.1.1, 1.1.2, 1.3.1 are children level nodes of 1.2 in Figure 1.
- Siblings - nodes at the same level of a node are called siblings, e.g. in Figure 1, 1.1.1, 1.1.2, 1.2.2, 1.2.3 and 1.3.1 are siblings of 1.2.1.

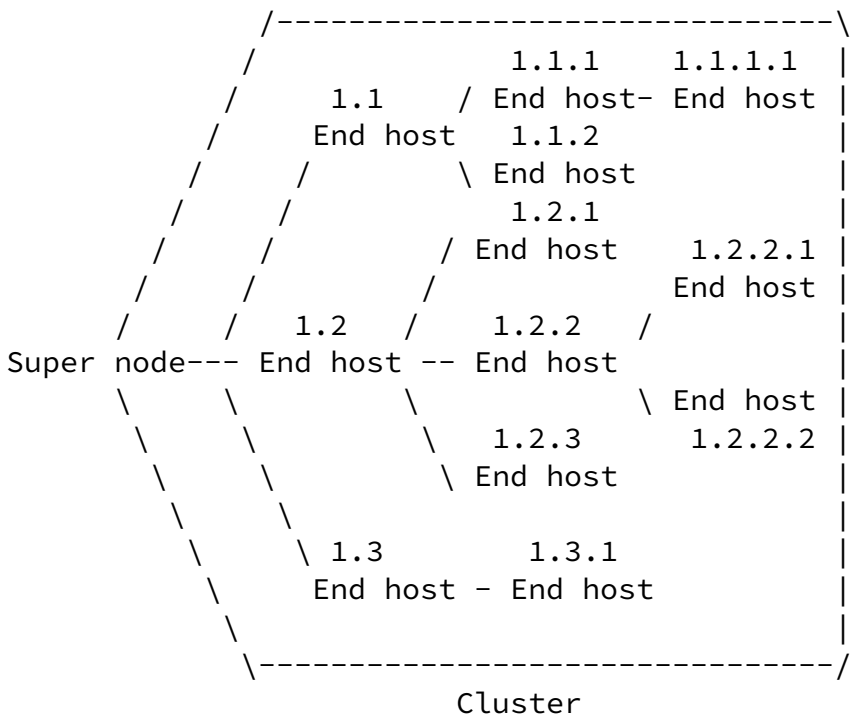


Figure 1: An example of local Cluster

4. DMMP Overview

The DMMP framework consists of two types of functionalities: control plane and data plane. The control plane composes one overlay mesh and some core-based clusters. Data plane is, then, built on the top of the structured control plane. Meanwhile, the source entity and a set of super nodes form the overlay mesh, in which each super node supervises one cluster. Although the Tree-based overlays are regarded as the most efficient approach for data distribution in a stable network, they are not effective for dynamic scenarios since pure tree structure has difficulties to meet both high bandwidth and high reliability requirements. The first difficulty is the delivered service quality to downstream end hosts is limited by the minimum bandwidth among the upstream connections from the source; for instance, a member in the OMNI tree receives data only from its upstream node and the data reception rate of this member can not be greater than its upstream node. It is even more difficult in the core network where each MSN should be responsible for data delivery to a whole cluster. For the second requirement, a tree is less robust than a mesh because a single node failure or a loop can partition the tree and disable communications among the members. To increase the throughput of the whole overlay network, an DMMP-aware mesh as one of multiple forwarding solutions is introduced in this draft. Such a mesh will allow the reception of packets from multiple nodes other than from only the upstream node.

4.1. Control plane in DMMP

In this source-specific overlay multicast protocol, the combination of available bandwidth [15] and uptime represents the capacity of each node [13]. For media streaming system, available bandwidth resources possessed by a multicast group may be insufficient during runtime [16]. It is reason why DMMP needs to consider the degree bound in streaming applications, which can be easily observed from the available bandwidth. For example, on the assumption that the bit rate of media is B and the outbound bandwidth of an end host i is b(i), the total number of connections it can establish is b(i)/B which is also the maximum degree of the end host. Moreover, the usage of available bandwidth in overlay routing has become possible, based on recent advances in available bandwidth measurement techniques and tools [17], [18]. Obviously, if an application has additional requirements on end-to-end delay or loss rate, these metrics can be jointly considered during the overlay hierarchy construction.

- Step 1: After initialization, RP will calculate the out-degree of each host and distribute them into two categories: leaf nodes (whose out-degree is less than 2) and non-leaf nodes. If one's

out-degree is less than 2, the end host can act as leaf node because it can only receive data from the incoming connection.

- Step 2: The information of two-category nodes are respectively stored at the RP. Meanwhile, all non-leaf nodes are placed in the order of out-degree and reported to the source. On receiving the list of ordered non-leaf nodes, the source selects an application-specific number of them as super nodes. Those nodes have higher capacities as defined in Section 6.2, and are used to manage the multicast group. In the initialization stage, nodes with higher bandwidth support will be selected as super nodes since the current uptime is zero. The capacities of each super node are also stored at the source and the RP.
- Step 3: After being selected, the super nodes organize themselves into a mesh rooted at the source. The issue of overlay mesh contruction is mostly motivated from [5].

DMMP considers the heterogeneous capacities of group members by evaluating their available bandwidth during runtime so that high-capacity nodes (i.e., super nodes) which are able to and willing to make more contributions to the network are expected to get better performance. This will maximize the usage of available bandwidth for the overlay tree. To further improve the overlay mesh performance, DMMP allows dynamically adding and deleting links within the mesh, which will be clarified in a future version of this draft. Based on selected super nodes, some core-based clusters will be formed to connect with the mesh.

- Step 1: After constructing the overlay mesh, the next step is to form core-based clusters. Each non-super node will firstly consult its local cache for super node candidates. If there is no suitable candidate, it queries the RP immediately. Then, the requestor caches these newly received candidates, from which it selects the best one based on e2e latency measurements. If there are multiple super nodes which can provide similar e2e latency for the node, one of them with higher out-degree will be chosen.
- Step 2: Those non-super nodes sharing the same super node will form a local cluster. The cluster formation is initiated by the super node which answers for informing the RP and contacting the source. Generally, certain number (due to the super node's available bandwidth) of end hosts with higher capacity will be selected as its immedidate children. This operation guarantees that the multicast tree within each cluster meets the bandwidth need of media streaming applications.
- Step 3: Afterwards, direct children of super nodes choose some nodes with higher capacities (i.e. out-degree, e2e latency) as their children. This selection method will expedite the

convergence of the tree and alleviate the average latency in some senses.

The iteration will continue until all cluster members join the tree, and accordingly the control hierarchy is constructed for the multicast group . For the sake of resilience, each node in the local cluster should keep some information of its relatives in the local cache. In this draft, these entities (parent, PLN, child, CLN and siblings, see in Section 2) are denoted as relatives.

Figure 2 illustrates a basic example for the overlay construction. Assuming that it is the first time to construct the overlay hierarchy, then:

- Step 1: When obtaining the list of non-leaf members from the RP, the source selects six of them as super nodes. After selection, super nodes self-organize into a mesh rooted at the source.
- Step 2: Each non-super node chooses the best super node based on the e2e latency measurements. Those non-super nodes sharing the same super node will then form a local cluster.
- Step 3: According to the super node's capacity, no more than K end hosts with larger capacities are selected as its immediate children. Here, three end hosts, namely, 1.1, 1.2 and 1.3 are chosen as the immediate children of the super node.
- Step 4: Once the capacity of the super node is exhausted, it responds to new requestors with its immediate children and an indication of rejection. For example, upon the receipt of rejections and a list of candidate parents (i.e. 1.1, 1.2 and 1.3) from the super node, end hosts 1.1.1, 1.1.2 and 1.3.1 send Join request to them. In this case, requestors with higher out-degree are likely to be selected as the children. If there are multiple acceptances, the end host will select the one which is "near" in terms of the e2e latency. For example, 1.1.1 and 1.1.2 accept the request and join the cluster as children of 1.1. Then 1.1 will update the associated information to 1.2, 1.3, and as well as to the super node after its children selection.
- Step 5: If there are still some nodes left, they will receive a rejection and need to re-send Join request to the children at the lower level, i.e. 1.1.1, 1.1.2 and 1.3.1. When all receivers confirm their positions in the cluster, the control plane is initially constructed.

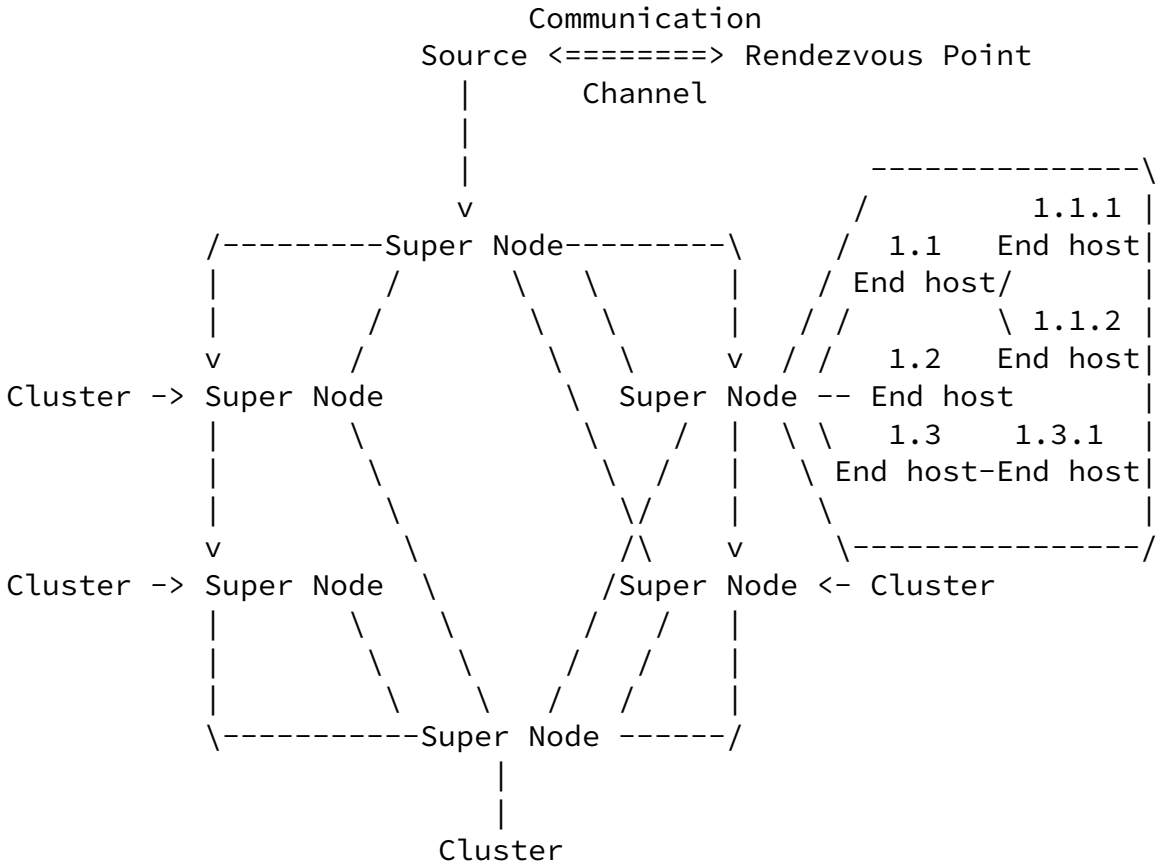


Figure 2: Control Hierarchy in DMMP

After constructing the control hierarchy, only super nodes need to keep the full knowledge among themselves, while non-super nodes only need to keep the knowledge of a small part of the group within each cluster. This will help dramatically reducing the control overhead of the whole multicast tree compared with that each member keeps the full knowledge of the entire group whilst providing certain

resilience. Therefore, the overlay hierarchy of DMMP incurs a low delay penalty, makes effective use of network resources, and considers the heterogeneities of end hosts during multicast sessions.

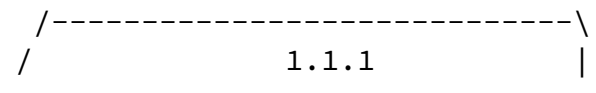
4.2. Data plane in DMMP

Corresponding to the constructed control plane, the data plane can be divided into two parts. The first part, which is within the overlay mesh, is built through the reverse shortest path between each super node and the source, identical to DVMRP [19]. For example, a super node A receives the packet from the source through its neighbor B only if B is the next hop on the shortest path from A to the source. In this case, super node A will replicate and forward the packets directly to super node B. Likewise, B will replicate and forward the packet to the next super node as well. When receiving the data, the super node will replicate and forward the data to its direct children

in the cluster. Therefore, data can be delivered throughout the whole cluster.

The second part is data delivery within core-based clusters. In each cluster, data are firstly forwarded from the super node to its direct children. Then, the receivers will replicate the data and forward them to its children at the lower level. The whole iteration terminates with success if all receivers within each cluster receive the data.

Figure 3 depicts an example for data delivery within a DMMP-aware cluster. Here, a respective data channel between two entities is established by exploiting the existing protocol stacks such as UDP/IP or TCP/IP. The data channels utilize IP unicast according to the underlying IP transport scheme. As shown in the figure, data are firstly replicated into three copies, respectively delivered from the super node to its direct children 1.1, 1.2 and 1.3 using unicast. Similarly, 1.1 replicates copies of the data according to the number of their children (e.g. two copies), sending to 1.1.1 and 1.1.2. In the next iteration, the receiver will similarly make copies and deliver to its children.



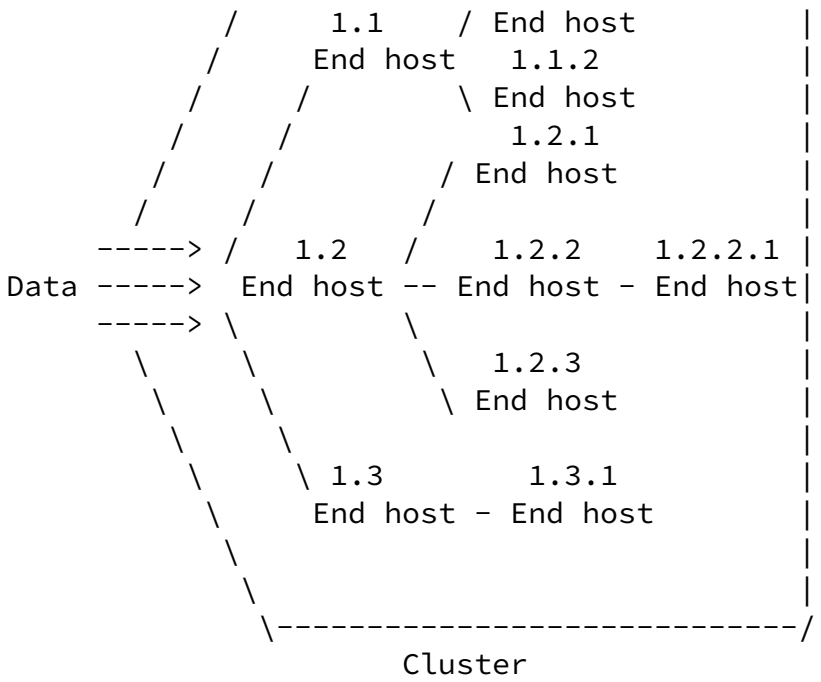
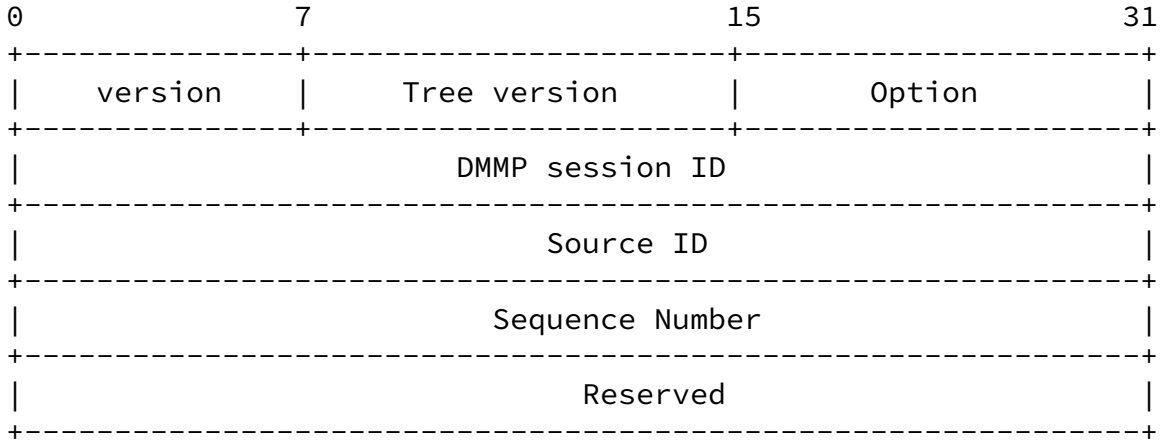


Figure 3: Data delivery in DMMP

5. DMMP Messages

DMMP handles tasks related to overlay hierarchy management, multicast tree configuration and maintenance. It uses a common format to carry both data and control packets as shown in Figure 4.



```
|                          Payload Data length                  |
+---------------------------------------------------------------+
```

Figure 4: DMMP Packet Header Format

Session ID and Source ID are generated by the RP and guaranteed to be collision free. The tree version field is used to prevent loops and partitions from the multicast tree. Since the multicast session tree is initialized and controlled by the RP, a loop free topology may be generated. Moreover, since tree update messages are independently disseminated to all group members, there is possibility that some messages might be lost and received out-of-order by different group members. These members may act on Refresh message with updating capacities. All these events could cause loops and tree partition. In order to avoid these failures, the RP will assign a monotonically increasing version number to each newly generated multicast tree.

The Option fields in the header defines various types of operation messages. Currently, there are seven pairs of control messages as shown in Table 1. Each pair of control messages will be exchanged between the DMMP-aware entities in a request-and-response way.

- Subscription Request and Response - Group members get the address of RP from Domain Name Systems (DNSs).
- Ping_RP Request and Response - During bootstrapping, each member of the group gets a list of available super nodes from the RP, containing at least one active node.
- Join Request and Response - A newly joining member sends request in order to join the multicast session and gets corresponding information from active group members.

- Status Request and Report - To request the status reports from neighbors or relatives, and accordingly to send reports to them.
- Probe Request and Response - To probe whether the target node is still active or not.
- Inactive Report and Response - To inform the other group members that the target node is inactive.
- Refresh Request and Response - To maintain the overlay hierarchy, they are used to periodically update the capacities (such as uptime, out-degree) of group members.

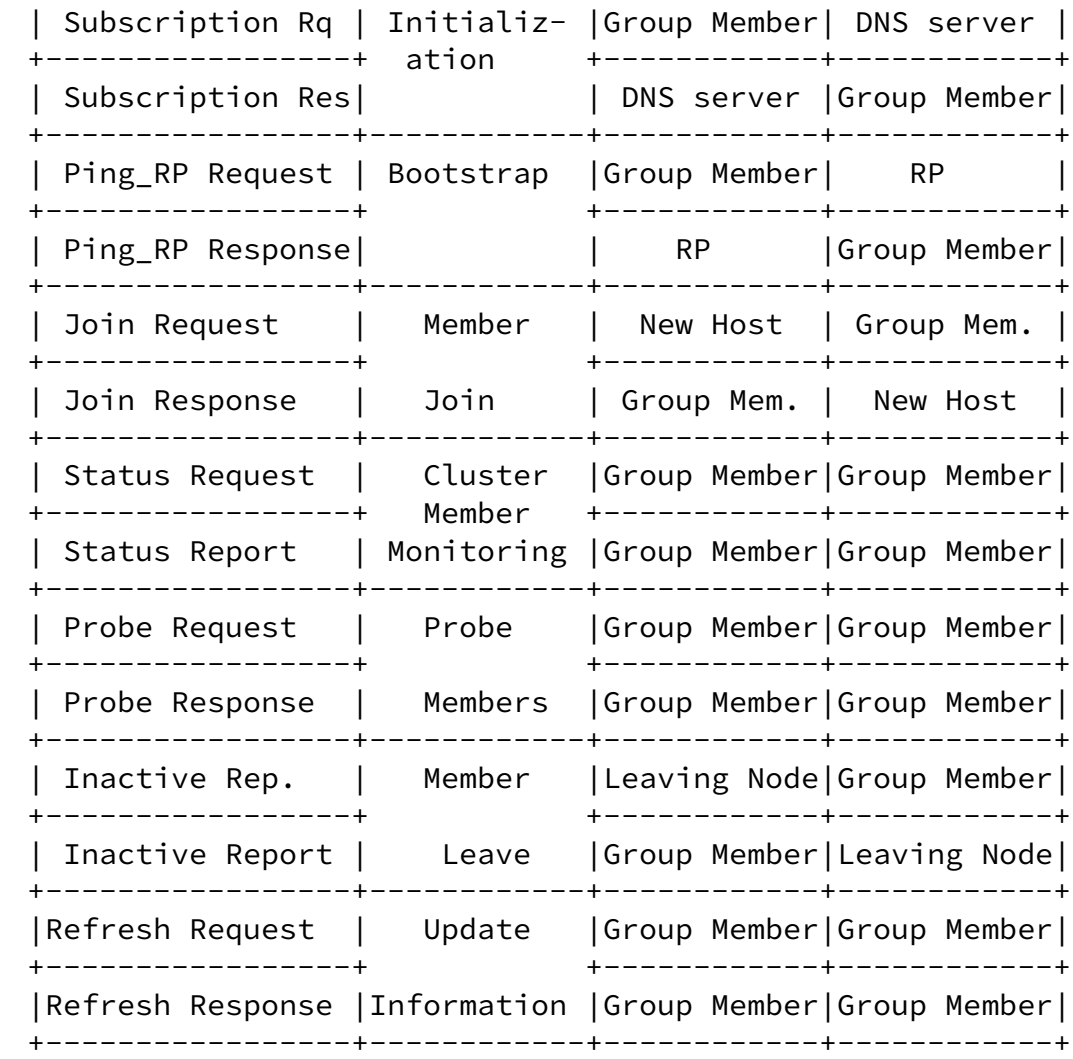
To adapt to dynamic network changes, each end host maintains the

overlay core by periodically updating its capacities. For example, it periodically exchanges Refresh message with its neighbors. If node A cannot receive this message from node B within the Refresh timer, node A will send a Probe request to node B. If there is still no Response returned, node B will be confirmed to be inactive by a certain time. Then the Status report, indicating node B being inactive, will be used to inform the rest of group members.

Table 1 lists the DMMP messages according to the associated DMMP operational phases.

Table 1: DMMP Messages

Messages	Operation	From	To



Although TCP provides a generic protocol for a guaranteed, in-order delivery of stream-based messages, this reliability comes at a price in performance. Besides, the communication pattern in DMMP is strictly in a request-response mode, and most messages have a small fixed maximum size. Thus, it is preferred to encapsulate all DMMP messages over UDP to provide the required delivery guarantees without extra network burdens.

6. DMMP: Protocol Details

All DMMP-aware nodes and the source are assumed to be able to know

the address of the RP. Also, once a source node starts, a direct communication channel will be established between the source node and the RP, so that the necessary information like the capacities of group members and active super nodes could be exchanged between them during the lifetime of an overlay multicast session. Furthermore, a DNS namespace is required to maintain the RP information for a specified multicast group.

Before initialization, each group member and the source send out Subscription request containing the specified group name and domain name, to the DNS for the address of associated Rendezvous Point (RP). Since RP does not participate in the data forwarding, the location of RP has no significant impact on the performance of data distribution. If there is no existing RP which is serving for this multicast group, DNS will allocate a new one for this multicast group based on application requirements. Otherwise, the RP's address will be sent back. It is possible that multiple RPs serve for the same multicast group, e.g. for the purpose of load balancing and fault tolerance. For simplicity, this document initially considers the case where there is only one RP involved.

6.1. Initialization

During the initialization phase, certain application related software has been distributed to the prospective DMMP-aware entities within the DMMP control and data transport models.

Before the multicast session starts, the source and RP must be ready to give response to requests from DMMP-aware end hosts. The source and RP will take no further reactions to any DMMP requests once the session stops. The active session time should be the period from the service starts until it stops. Then the out-of-band channel between the RP and source should be active during this active session so that the source can monitor the current status of memberships. However, the detailed mechanism for implementing this out-of-band bootstrapping is out of the scope of this document.

Moreover, session-related information should be obtained before the session starts and all prospective group members use out-of-band bootstrapping mechanism to get necessary information, for instance, Group ID and location of RP including the port number serving for certain sessions before the application begins. Then DMMP-aware entities can start receiving data after they join the overlay hierarchy.

6.2. Super Node Selection

During the mesh construction, the selection of the super nodes is to ensure that a newly joining member is able to quickly find its appropriate position in the multicast tree using a very small number of queries such as Join request. As the super node selection is the first step towards mesh establishment, this section gives more details.

To select super nodes for better performance while maintaining scalability, the following distribution requirements need to be taken into account [20].

- Connections: Super nodes have relatively higher capacties and are expected to be strong to perform additional tasks such as resources control, load balance and fault tolerance.
- Number: To be more efficient, the number of active super nodes is no more than one hundred, otherwise it may cause high control overhead and high stress [5]. Assuming that each super node can manage, in average, hundreds of cluster members, it is sufficient to support totally more than thousands of end hosts. Otherwise, multiple sources will be deployed into media streaming by multiple overlay multicast sessions. To balance the tradeoff between the efficiency and the reliability, it is reasonable to select an application-specific number of super nodes to construct the overlay mesh. For example, in the case of 110 end hosts, it may need 10 super nodes if the required ratio is 10%. In this case, 10 end hosts with higher capacity will be chosen as super nodes.
- Downstream: To adapt to bandwidth requirements, super nodes should not serve more than K non-super nodes as its immediate children, where K is respectively determined by the available out-degree of each super node and service specifications.

In order to deal with factors from a large-scale and dynamic network environment, the following three conditions are outlined in addition to above requirements.

- Stable: One cause for current multimedia streaming serivces which cannot guarantee required QoS mainly from unstable network status. Thus, super nodes should be relatively stable because, otherwise, its cluster members are easily partitioned from the tree.

- Resilience: Super nodes are responsible for detecting dynamic changes and for handling them quickly, e.g. one super node leaves the group ungracefully, which should be detected by at least one of other active nodes. Then, a new super node should be quickly selected to replace the leaving super node in the position. The time for detection and recovery process is also constrained by certain service requirements.

- Security: Super nodes should be fundamentally invulnerable to attacks; otherwise, they will easily disrupt the multicast service by forwarding wrong messages or failing to accept information.

6.3. Member Join

DMMP is resilient to dynamic network changes, for examples, events like a member joining/leaving.

The newcomer first checks its local cache for super node candidates. If there are no suitable candidates in the cache, it requests the RP for the addresses of the source and super node candidates. After receiving the source address and a list of active super nodes, it caches their capacities, i.e. uptime, out-degree. Then, it will measure the e2e latency between them and itself, and sends the Join Request message to some super node which can provide smaller e2e latency.

Upon receiving Join Request from a newcomer, the super node will check its current out-degree. If it is possible to accept the newcomer to as its immediate child, the super node will respond with an indication of acceptance. In this case, the information about newly joining nodes will only be propagated to the existing children of this super node since no child of the new member exists. When the super node cannot accept it as its immediate child, it will redirect this Join Request message to its active children with the largest available out-degree. If one of them responses to the super node, this response will be relayed to the new member. If there are more potential parents, the new member selects the one with smallest tree depth as its parent. If there are multiple potential parents at the same depth, it chooses the best one in terms of their uptime. Once finding the appropriate parent, the new member starts data delivery. At this time, the information about the new member will be propagated

from the parent to its PLN, siblings and the super node. The process will be terminated until the new member finds its position and accordingly updates its related information at corresponding nodes.

After joining the tree, the newcomers who have higher capacities could "climb" from the bottom to a higher level after some switch stages. For example, a newcomer at the lower level could switch with its parent if its capacity exceeds (over a predefined threshold) the current parent. The appropriate threshold will be defined to avoid unnessary switching since if the child has a smaller bandwidth support, it will be ultimately placed below the high capacity parent.

In case the newcomer fails to find an appropriate position in any cluster to meet application requirements, it can sell itself as a potential super node and report its own capacities to the RP.

Regarding its capacity and the current number of super nodes, it could be entitled as a super node. In this way, end hosts have more flexibility to get optimal services, which will be specified in a future version of this document.

6.4. Refresh Information

In DMMP, each member is responsible for maintaining the overlay hierarchy, by periodically sending Refresh message. The Refresh mechanism in the overlay mesh has a little difference from that in the local clusters. To efficiently manage the overlay hierarchy, both active and passive models are utilized in DMMP. Within each cluster, end host starts to exchange Refresh message with its PLNs, siblings and CLNs once it joins the cluster. In addition that each member has to periodically update its information, members in the local cluster are able to request refresh message from their relatives, e.g., PNLs or CNLs. This operation guarantees the reliability and the stability of the overlay hierarchy.

For the Refresh message in the mesh, each super node sends its current information to all mesh members including the source. Once receiving updated information, the source will correspondingly update the information at the RP. If one mesh member stops receiving Refresh message from another beyond the Mesh_Refresh_Timer, it assumes this neighbor to be either inactive or leaving. In order to confirm the status, it may initiate a Probe message as stated in

Section 5.

6.5. Member Leave

In most cases, two situations for a member leaving the group, either gracefully or ungracefully, are distinguished from each other. In each cluster, the leaving member should at least send an Inactive Request to its parent or one of its children. After receiving the confirmation, it can leave the group gracefully. Then the notified node will propagate this Inactive message to its relatives so that they can update their service membership tables. In the second case, the Inactive status will be detected by periodically exchanging Refresh messages. If any member within the cluster, say p, fails to receive a Refresh Report message from one of its required relatives, say q, within the refresh timeout Refresh_Timer, then p sends a redundant Probe Request message to q. If there is still no Probe Response message returned, p assumes q to be inactive and propagates this Status_Inactive message throughout the whole cluster. Afterwards, one of it children with relatively high capacity will replace its place, and other children will accordingly change their positions. Nevertheless, ungraceful leaving may cause the crash of whole multicast tree. DMMP is able to handle different situations by

detecting the failures and recovering quickly from them as shown in Section 6.7.

Compared with the handling in the local cluster, the operation is even tougher in the core mesh since all its cluster members are partitioned from the tree. When there is no end hosts connecting to the leaving super node, no further changes to the overlay are required. Before a super node gracefully leaves the group, it must recommend a replacement leader for the cluster it owns and inform other super nodes in the overlay mesh before leaving. To detect unannounced leaves, DMMP relies on the periodic Refresh message exchanges. If the failed peer happens to be a super node, the overlay hierarchy has to be repaired, which will be depicted in Section 6.7.

6.6. Data Delivery Control

After the multicast tree configuration, the new member will ask its immediate parent to send the data. Generally, parent nodes will

delete data in their local cache after they have forwarded them to their children. If the parent still holds the data, the new member can quickly get the data from it. If the parent has not received data yet, either it waits until the parent forwards the data after receiving, or it directly inquires the super node to deliver the data. The former option is preferred in DMMP as its overhead is likely lower than the latter one. Upon receiving the data, the new member will firstly forward them to its parent if its parent hasn't received the data yet. If the parent has deleted the data, it will then ask its siblings to forward data directly. In order to alleviate the redundant data transmission, the new member needs to wait for a certain while before it asks for the data from its siblings.

Different from joining as a cluster member, the new member may join the multicast session as a super node. In this case, it will firstly ask its neighbors in the overlay mesh to send the data. In addition, it may query the data from its children when they are already in its local cluster. If any of them receives the data, this new super node will also get the data. A third possibility is to let the new member directly ask the source to send the data. To alleviate the control overhead, it is recommended that this new node waits for a certain while until one of its mesh neighbors receives the data.

6.7. Failure Recovery

Maintenance of such a multicast tree in DMMP faces a key problem that non-leaf nodes in the tree are end hosts who are more likely to fail than routers and may join/leave the tree at will. It does not happen

in IP multicast since non-leaf nodes in the delivery tree are routers which do not leave the multicast tree without notification. Thus, one challenge in DMMP is to reconstruct the overlay multicast tree after a node's departure. If one non-leaf node leaves the group ungracefully, its downstream nodes will be inevitably affected. Two possible means can alleviate such impacts: one is to reduce the possibility of failures; the other is to reduce the number of possible affected nodes. In practice, however, the first way might be very difficult since end hosts may leave the group at will. For the second option, DMMP proposes a proactive mechanism by periodically pushing high-capacity nodes to higher levels of the tree by capacity comparison mechanism. Detailed mechanisms are described

in Section 6.8. Thereby, it is very likely that long-lived super nodes and their immediate children form a stable and efficient cluster core after a certain time. The longer a node remains in the multicast session, the more it becomes attaching to other long-lived nodes with similar uptime. In other words, higher capacity nodes form a well-connected core with relatively more bandwidth support and being more stable, whereas peers with less available bandwidth and shorter uptime will be placed out of the core as possible.

In order to improve the performance of DMMP, especially when there are high packet losses or host failures, a reactive recovery technique is also implemented after failure detection. Recovery from failures regarding a member crash is similar to handling a member leaves. The difference is that surviving members usually do not receive prior notification of a crash. Thus, Refresh message is periodically exchanged between each member and its neighbors. It is even more difficult for super nodes to maintain the cluster since all of its local members are partitioned from the multicast tree and can not receive the multicast data until it is repaired.

In the local cluster, each immediate child of the super node must find a backup parent in advance, either the source or a group member. Once the super node leaves the group, its children try to contact with their alternative parents to re-join the multicast tree. This approach can facilitate the recovery process and strengthen the reliability of the overlay hierarchy. In addition, cluster members periodically estimate their relatives (i.e. PNLs, CNLs) within the cluster and evaluates the number of losses that it shares with these nodes. While in the core mesh, each super node maintains state information about all other mesh members, no additional discovery of nodes is necessary. Using this mechanism, packet delivery ratios can be increased with a high probability. To handle different scenarios of failures, more mechanisms need to be defined in the near future.

6.8. Self-improvement

We suggest the self-improvement mechanisms and hence a cluster member having a higher capacity than its parent could be promoted. Basically, the children and their parents can swap their positions.

After promotion, the former child becomes the new parent and its former parent becomes the current child. However, there are some aspects impact the mechanism:

- the number of nodes involved in the promotion: the more nodes change their positions, the less stable the overlay tree becomes.
- the reliability of participated nodes: the promoted node may leave ungracefully, and its children will be partitioned from the existing tree.
- complixity of re-construction: after promotion, the re-construction of the existing tree may be complicated since the promoted child may have not enough bandwidth to accept all exisiting end hosts as its children.

For simplicity, we describe the idea by taking the following example. In Figure 5, suppose that node 1.1.2 has much higher capacity than its parent 1.1 based on the capacity comparison. Then, node 1.1.2 sends a promotion request to node 1.1. After a certain proof, node 1.1 acknowledges the request and sends back a status report which contains the address of node s. Here, it is necessary that node 1.1 waits till node s has received the breakup request. Otherwise, the join request from 1.1.2 may arrive earlier, which will cause a loop in the overlay tree.

Then, node 1.1 breaks the connections with node s and 1.1.2. However, node 1.1 keeps node s as its backup parent in case node 1.1.2 is leaving or unreachable. Moreover, node s considers node 1.1 as it temporary child. At the same time, node 1.1.2 contacts node s and notifies node 1.1 to be its child. Once node 1.1 receives the notification and rejoins the tree as the child of node 1.1.2, it may break the connection with node 1.1.1 if node 1.1.2 still has available capacity. In the following example, node 1.1.2 can support at least three children. Therefore, after the first swap, the node 1.1.1 requests to join as one child of node 1.1.2. The message flow of the promotion example is shown in Figure 6.

Above mechanism can be tolerant to dynamic changes. Considering the scenario that node 1.1 and 1.1.1 can still quickly rejoin the tree even if node 1.1.1 leaves ungracefully. The optimization and extension of the mechanism will be studied in the near future.

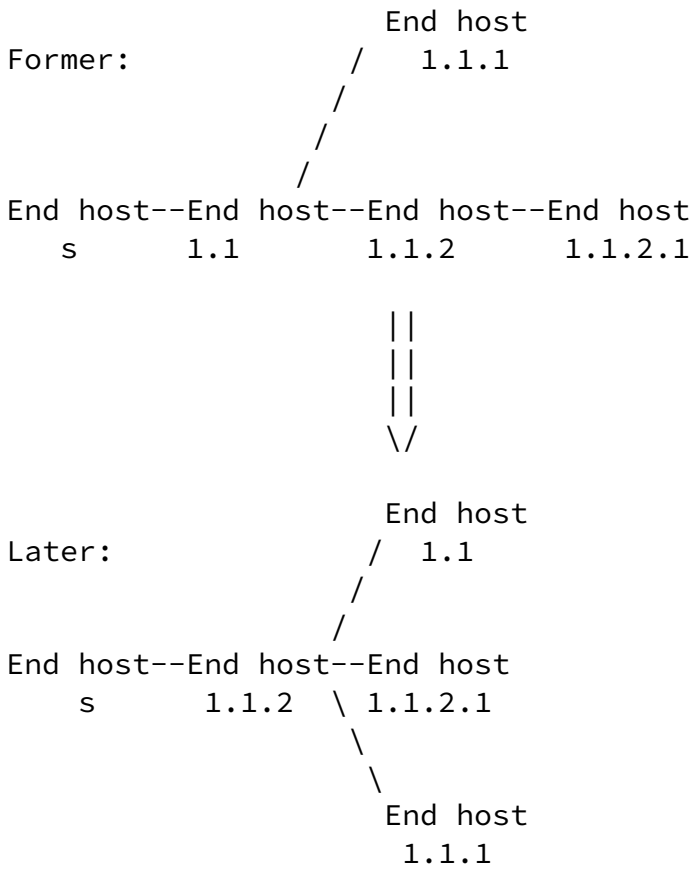


Figure 5: An example of self-improvement

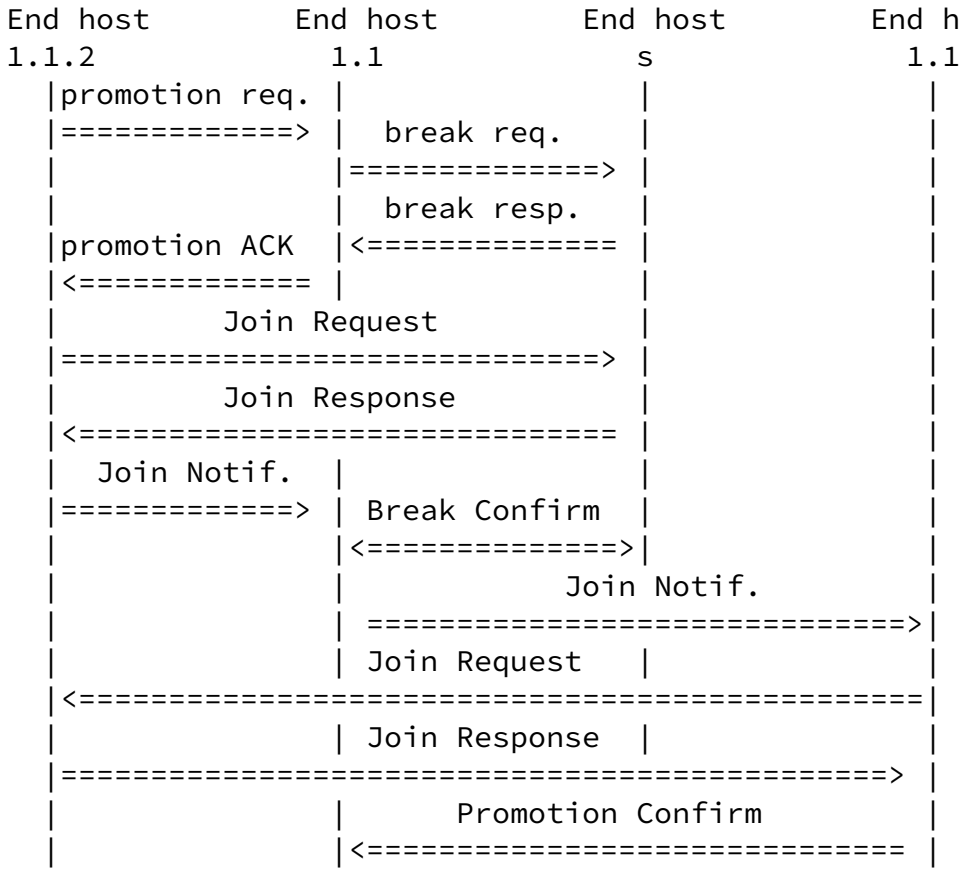


Figure 6: Message flow of self-improvement example

Moreover, nodes at the bottom level are either transient nodes, leaf nodes or new comers. The newcomers who have higher capacities could "climb" from the bottom to a higher level after some switching stages. For example, a newcomer at the lower level could switch with its parent if its capacity exceeds (over a predefined threshold) the current parent. Nevertheless, an appropriate threshold will be defined to avoid unnecessary switching since if the child has a smaller bandwidth support, it will be ultimately placed below the parent. The main goal of doing this is to reducing the impacts of frequent changes in the overlay so that only a small part of the overlay multicast tree will be affected and needs to be re-constructed after dynamic changes.

7. Metrics specification

End host based overlay multicast is more susceptive to dynamic network changes since end hosts may join or leave the group at will. It would be even harder for DMMP to manage and maintain such an overlay mesh because super nodes may leave the group ungracefully as well. To address the instability of mesh, uptime is chosen as an assisted criterion to strengthen its maintenance. Once an end host

joins in the overlay multicast tree, its uptime starts to calculate from 0 until its leaving. Besides, there are several metrics used in DMMP as the criteria of the capacity, which will be specified in this section.

Table 2: Capacity specification

Metric	Operation
Out-degree	Differentiation non-leaf nodes from leaf nodes
	Super nodes selection
	Tree construction within clusters
	New member joins the group
	Failure recovery mechanism
	Self-improving mechanism
E2E latency	Non-super nodes attach to super nodes to form clusters
	New member joins the group
Uptime	New member joins the group
	Self-improving mechanism
	Failure recovery mechanism

As shown in Table 2, out-degree is the main criterion to select the super nodes from end hosts. Then, non-super nodes select one super node which locates "near" to it based on the estimated e2e latency. During the tree construction within clusters, nodes with higher out-degree are likely to join in the tree at the high level. Regarding new members joining procedure, out-degree, e2e latency and uptime are all taken into considerations. To keep the stability of the overlay hierarchy, out-degree and uptime are chosen as comparison metric to self-improve the overlay multicast tree. Furthermore, out-degree and uptime are regarded as the main selection criterion for alternative node during the failure recovery.

8. Security Considerations

Security should be considered during the super nodes selection. In DMMP, the current preference is to use an authority center that qualifies the trust level of end hosts. Only when the end host obtains a security certificate from the authority center, it can be selected as a super node.

Within each cluster, Cluster key, Group key and Private key are proposed as the security scheme to manage the cluster members. Details and discussions related to other security issues are to be explored in a future version of this document.

9. Open Issues

DMMP framework will study necessary extensions or open issues, where security, large-scale efficiency, end-to-end quality-of-service (QoS) provisioning will be likely related. Moreover, initial DMMP does not include support for NATs and firewalls since they impose fundemental restrictions on pair-wise connectivity of hosts on the overlay.

10. Contributors

Ruediger Geib, Xiaodong Yang and David Weiss contributed great efforts to this document.

11. Acknowledgements

We thank Nicolai Leymann, John Buford, Yuji-UG-Imai and Yangwoo Ko for their helpful suggestions.

12. References

12.1. Normative References

[1] Bhattacharyya, S., "An Overview of Source-Specific Multicast (SSM)", RFC 3569, July 2003.

[2] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

12.2. Informative References

[3] Almeroth, K., "The evolution of multicast: From the MBone to inter-domain multicast to Internet2 deployment", IEEE Network 2000, Jan./Feb 2000.

[4] Diot, C., Levine, B., Lyles, J., and D. Balensiefen, "Deployment issues for IP multicast service and architecture", IEEE Network 2000, Jan. 2000.

[5] Chu, Y., Rao, S., and et. al, "A Case for End System Multicast", IEEE JSAC Vol. 20, No. 8, October 2002.

[6] Banerjee, S. and B. Bhattacharjee, "Analysis of the NICE Application Layer Multicast Protocol", UMIACS Technical report TR 2002-60 and CS-TR 4380, June 2002.

[7] Banerjee, S., Bhattacharjee, B., and et. al, "Scalable Application Layer Multicast", SIGCOMM 2002, August 2002.

[8] Banerjee, S., Kommareddy, C., and et. al, "OMNI: An Efficient

Infrastructure for Real-time Applications", Computer Networks, Special Issue on Overlay Distribution Structures and their Applications, Vol. 50, No. 6, April 2006.

[9] Lao, L., Cui, J., and et. al, "TOMA: A Viable Solution for Large-scale Multicast Service Support", IFIP Networking 2005, May 2005.

[10] Hefeeda, M. and B. Bhargava, "On-Demand Media Streaming Over the Internet", FTDCS'03, The Ninth IEEE Workshop on Future Trends of Distributed Computing Systems, ftdcs, p. 279, April 2003.

[11] Saroiu, S., Gummadi, P., and S. Gribble, "A Measurement Study of Peer-to-Peer File Sharing Systems", MMCN'02 Proceedings of Multimedia Computing and Networking, July 2002.

[12] Sripanidkulchai, K., Maggs, B., and H. Zhang, "An analysis of live streaming workloads on the Internet", SIGCOMM IMC Proceedings of the 4th ACM SIGCOMM IMC, Oct. 2004.

[13] Lei, J., Fu, X., and D. Hogrefe, "DMMP: A New Dynamic Mesh-based Overlay Multicast Protocol Framework", accepted in the 2007 IEEE Consumer Communications and Networking Conference, Workshop on Peer-to-Peer Multicasting (P2PM 2007), Jan. 2007.

[14] Ballardie, A. and J. Crowcroft, "Core based trees (CBT)", ACM SIGCOMM Computer Communication Review, October 1993.

[15] Hu, N. and P. Steenkiste, "Evaluation and Characterization of Available Bandwidth Probing Techniques", IEEE JSAC Special Issue in Internet and WWW Measurement, Vol. 21, No. 6, August 2003.

[16] Tan, G., Jarvis, S., and D. Spooner, "Improving Fault Resilience of Overlay Multicast for Media Streaming", DSN'06 IEEE International Conference on Dependable Systems and Networks, June 2006.

[17] Jain, M. and C. Dovrolis, "End-to-end available bandwidth:

measurement methodology, dynamics, and relation with tcp throughput", Transaction'03 IEEE/ACM Transaction on Networking 11 (4), August 2003.

[18] Strauss, J., Katabi, D., and F. Kaashoek, "A measurement of study of available bandwidth estimation tools", IMC'03 Proceedings of ACM SIGCOMM Conference on Internet Measurement, Oct. 2003.

[19] Deering, S., "Multicasting routing in internetworks and extended LANs", ACM SIGCOMM 1988, August 1988.

[20] Lo, V., Zhou, D., and et. al, "Scalable Supernode Selection in Peer-to-Peer Overlay Networks", HOT-P2P'05 Proceedings of International Workshop on Hot Topics in Peer-to-Peer Systems 2005, July 2005.

[21] Handler, G. and P. Mirchandani, "Location on Networks: Theory and Algorithms", The MIT Press Cambridge Massachusetts, Mar. 1979.

[22] Lynch, A., "Distributed Algorithms", Morgan Kaufmann San Francisco, April 1997.

[23] Zhi, L. and P. Mohapatra, "HostCast: A New Overlay Multicasting Protocol", IEEE ICC'03 Proceedings of IEEE ICC 2003, June 2003.

[24] Banerjee, S., Lee, S., and et. al, "Resilient Multicast using Overlays", ACM SIGMETRICS 2003, June 2003.

[25] Paul, S. and K. Sabnani, "Reliable multicast transport protocol (RTMP)", IEEE JSAC, Vol. 15, No. 3, April 1997.

Authors' Addresses

Jun Lei
University of Goettingen
Institute for Informatics
Lotzestr. 16-18
Goettingen 37083

Germany

Email: lei@cs.uni-goettingen.de

Xiaoming Fu
University of Goettingen
Institute for Informatics
Lotzestr. 16-18
Goettingen 37083
Germany

Email: fu@cs.uni-goettingen.de

Dieter Hogrefe
University of Goettingen
Institute for Informatics
Lotzestr. 16-18
Goettingen 37083
Germany

Email: hogrefe@cs.uni-goettingen.de

Full Copyright Statement

Intellectual Property

Acknowledgment

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).