Grenville Armitage
Bellcore
April 21st, 1997

VENUS - Very Extensive Non-Unicast Service
<draft-armitage-ion-venus-02.txt>

Status of this Memo

Abstract

The MARS model (RFC2022) provides a solution to intra-LIS IP multicasting over ATM, establishing and managing the use of ATM pt-mpt SVCs for IP multicast packet forwarding. Inter-LIS multicast forwarding is achieved using Mrouters, in a similar manner to which the `Classical IP over ATM' model uses Routers to inter-connect LISes for unicast traffic. The development of unicast IP shortcut mechanisms (e.g. NHRP) has led some people to request the development of a Multicast equivalent. There are a number of different approaches. This document focuses exclusively on the problems associated with extending the MARS model to cover multiple

clusters or clusters spanning more than one subnet. It describes a hypothetical solution, dubbed `Very Extensive NonUnicast Service' (VENUS), and shows how complex such a service would be. It is also noted that VENUS ultimately has the look and feel of a single, large cluster using a distributed MARS. This document is being issued to help focus ION efforts towards alternative solutions for establishing ATM level multicast connections between LISes.

1. Introduction

The classical model of the Internet running over an ATM cloud consists of multiple Logical IP Subnets (LISs) interconnected by IP Routers [1]. The evolving IP Multicast over ATM solution (the `MARS model' [2]) retains the classical model. The LIS becomes a `MARS Cluster', and Clusters are interconnected by conventional IP Multicast routers (Mrouters).

The development of NHRP [3], a protocol for discovering and managing unicast forwarding paths that bypass IP routers, has led to some calls for an IP multicast equivalent. Unfortunately, the IP multicast service is a rather different beast to the IP unicast service. This document aims to explain how much of what has been learned during the development of NHRP must be carefully scrutinized before being re-applied to the multicast scenario. Indeed, the service provided by the MARS and MARS Clients in [2] are almost orthogonal to the IP unicast service over ATM.

For the sake of discussion, let's call this hypothetical multicast shortcut discovery protocol the `Very Extensive Non-Unicast Service' (VENUS). A `VENUS Domain' is defined as the set of hosts from two or more participating Logical IP Subnets (LISs). A multicast shortcut connection is a point to multipoint SVC whose leaf nodes are scattered around the VENUS Domain. (It will be noted in section 2 that a VENUS Domain might consist of a single MARS Cluster spanning multiple LISs, or multiple MARS Clusters.)

VENUS faces a number of fundamental problems. The first is exploding the scope over which individual IP/ATM interfaces must track and react to IP multicast group membership changes. Under the classical IP routing model Mrouters act as aggregation points for multicast traffic flows in and out of Clusters [4]. They also act as aggregators of group membership change information - only the IP/ATM interfaces within each Cluster need to know the specific identities of their local (intra-cluster) group members at any given time. However, once you have sources within a VENUS Domain establishing shortcut connections the data and signaling plane aggregation of Mrouters is lost. In order for all possible sources throughout a

VENUS Domain to manage their outgoing pt-mpt SVCs they must be kept aware of MARS_JOINs and MARS_LEAVEs occuring in every MARS Cluster that makes up a VENUS Domain. The nett effect is that a VENUS domain looks very similar to a single, large distributed MARS Cluster.

A second problem is the impact that shortcut connections will have on IP level Inter Domain Multicast Routing (IDMR) protocols. Multicast groups have many sources and many destinations scattered amongst the participating Clusters. IDMR protocols assume that they can calculate efficient inter-Cluster multicast trees by aggregating individual sources or group members in any given Cluster (subnet) behind the Mrouter serving that Cluster. If sources are able to simply bypass an Mrouter we introduce a requirement that the existence of each and every shortcut connection be propagated into the IDMR decision making processes. The IDMR protocols may need to adapt when a source's traffic bypasses its local Mrouter(s) and is injected into Mrouters at more distant points on the IP-level multicast distribution tree. (This issue has been looked at in [7], focussing on building forwarding trees within networks where the termination points are small in number and sparsely distributed. VENUS introduces tougher requirements by assuming that multicast group membership may be dense across the region of interest.)

This document will focus primarily on the internal problems of a VENUS Domain, and leave the IDMR interactions for future analysis.

2. What does it mean to `shortcut' ?

Before going further it is worth considering both the definition of the Cluster, and two possible definitions of `shortcut'.

2.1 What is a Cluster?

In [2] a MARS Cluster is defined as the set of IP/ATM interfaces that are willing to engage in direct, ATM level pt-mpt SVCs to perform IP multicast packet forwarding. Each IP/ATM interface (a MARS Client) must keep state information regarding the ATM addresses of each leaf node (recipient) of each pt-mpt SVC it has open. In addition, each MARS Client receives MARS_JOIN and MARS_LEAVE messages from the MARS whenever there is a requirement that Clients around the Cluster need to update their pt-mpt SVCs for a given IP multicast group.

It is worth noting that no MARS Client has any concept of how big its local cluster is - this knowledge is kept only by the MARS that a given Client is registered with.

Fundamentally the Cluster (and the MARS model as a whole) is a

response to the requirement that any multicast IP/ATM interface using pt-mpt SVCs must, as group membership changes, add and drop leaf nodes itself. This means that some mechanism, spanning all possible group members within the scopes of these pt-mpt SVCs, is required to collect group membership information and distribute it in a timely fashion to those interfaces. This is the MARS Cluster, with certain scaling limits described in [4].

2.2 LIS/Cluster boundary `shortcut'

The currently popular definition of `shortcut' is based on the existence of unicast LIS boundaries. It is tied to the notion that LIS boundaries have physical routers, and cutting through a LIS boundary means bypassing a router. Intelligently bypassing routers that sit at the edges of LISs has been the goal of NHRP. Discovering the ATM level identity of an IP endpoint in a different LIS allows a direct SVC to be established, thus shortcutting the logical IP topology (and very real routers) along the unicast path from source to destination.

For simplicity of early adoption RFC2022 recommends that a Cluster's scope be made equivalent to that of a LIS. Under these circumstances the `Classical IP' routing model places Mrouters at LIS/Cluster boundaries, and multicast shortcutting must involve bypassing the same physical routing entities as unicast shortcutting. Each MARS Cluster would be independent and contain only those IP/ATM interfaces that had been assigned to the same LIS.

As a consequence, a VENUS Domain covering the hosts in a number of LIS/Clusters would have to co-ordinate each individual MARS from each LIS/Cluster (to ensure group membership updates from around the VENUS Domain were propagated correctly).

2.3 Big Cluster, LIS boundary `shortcut'

The MARS model's fundamental definition of a Cluster was deliberately created to be independent of unicast terminology. Although not currently well understood, it is possible to build a single MARS Cluster that encompasses the members of multiple LISs. As expected, inter-LIS unicast traffic would pass through (or bypass, if using NHRP) routers on the LIS boundaries. Also as expected, each IP/ATM interface, acting as a MARS Client, would forward their IP multicast packets directly to intra-cluster group members. However, because the direct intra-cluster SVCs would exist between hosts from the different LISs making up the cluster, this could be considered a `shortcut' of the unicast LIS boundaries.

This approach immediately brings up the problem of how the IDMR

protocols will react. Mrouters only need to exist at the edges of Clusters. In the case of a single Cluster spanning multiple LISs, each LIS becomes hidden behind the Mrouter at the Cluster's edge. This is arguably not a big problem if the Cluster is a stub on an IDMR protocol's multicast distribution tree, and if there is only a single Mrouter in or out of the Cluster. Problems arise when two or more Mrouters are attached to the edges of the Cluster, and the Cluster is used for transit multicast traffic. Each Mrouter's interface is assigned a unicast identity (e.g. that of the unicast router containing the Mrouter). IDMR protocols that filter packets based on the correctness of the upstream source may be confused at receiving IP multicast packets directly from another Mrouter in the same cluster but notionally `belonging' to an LIS multiple unicast IP hops away.

Adjusting the packet filtering algorithms of Mrouters is something that needs to be addressed by any multicast shortcut scheme. It has been noted before and a solution proposed in [7]. For the sake of argument this document will assume the problem solvable. (However, it is important that any solution scales well under general topologies and group membership densities.)

A multi-LIS MARS Cluster can be considered a simple VENUS Domain. Since it is a single Cluster it can be scaled using the distributed MARS solutions currently being developed within the IETF [5,6].

3. So what must VENUS look like?

A number of functions that occur in the MARS model are fundamental to the problem of managing root controlled, pt-mpt SVCs. The initial setup of the forwarding SVC by any one MARS Client requires a query/response exchange with the Client's local MARS, establishing who the current group members are (i.e. what leaf nodes should be on the SVC). Following SVC establishment comes the management phase - MARS Clients need to be kept informed of group membership changes within the scopes of their SVCs, so that leaf nodes may be added or dropped as appropriate.

For intra-cluster multicasting the current MARS approach is our solution for these two phases.

For the rest of this document we will focus on what VENUS would look like when a VENUS Domain spans multiple MARS Clusters. Under such circumstances VENUS is a mechanism co-ordinating the MARS entities of each participating cluster. Each MARS is kept up to date with sufficient domain-wide information to support both phases of client operation (SVC establishment and SVC management) when the SVC's

endpoints are outside the immediate scope of a client's local MARS. Inside a VENUS Domain a MARS Client is supplied information on group members from all participating clusters.

The following subsections look at the problems associated with both of these phases independently. To a first approximation the problems identified are independent of the possible inter-MARS mechanisms. The reader may assume the MARS in any cluster has some undefined mechanism for communicating with the MARSs of clusters immediately adjacent to its own cluster (i.e. connected by a single Mrouter hop).

3.1 SVC establishment - answering a MARS_REQUEST.

The SVC establishment phase contains a number of inter-related problems.

First, the target of a MARS_REQUEST (an IP multicast group) is an abstract entity. Let us assume that VENUS does not require every MARS to know the entire list of group members across the participating clusters. In this case each time a MARS_REQUEST is received by a MARS from a local client, the MARS must construct a sequence of MARS_MULTIs based on locally held information (on intra-cluster members) and remotely solicited information.

So how does it solicit this information? Unlike the unicast situation, there is no definite, single direction to route a MARS_REQUEST across the participating clusters. The only `right' approach is to send the MARS_REQUEST to all clusters, since group members may exist anywhere and everywhere. Let us allow one obvious optimization - the MARS_REQUEST is propagated along the IP multicast forwarding tree that has been established for the target group by whatever IDMR protocol is running at the time.

As noted in [4] there are various reasons why a Cluster's scope be kept limited. Some of these (MARS Client or ATM NIC limitations) imply that the VENUS discovery process not return more group members in the MARS_MULTIs that the requesting MARS Client can handle. This provides VENUS with an interesting problem of propagating out the original MARS_REQUEST, but curtailing the MARS_REQUESTs propagation when a sufficient number of group members have been identified. Viewed from a different perspective, this means that the scope of shortcut achievable by any given MARS Client may depend greatly on the shape of the IP forwarding tree away from its location (and the density of group members within clusters along the tree) at the time the request was issued.

How might we limit the number of group members returned to a given MARS Client? Adding a limit TLV to the MARS_REQUEST itself is

trivial. At first glance it might appear that when the limit is being reached we could summarize the next cluster along the tree by the ATM address of the Mrouter into that cluster. The nett effect would be that the MARS Client establishes a shortcut to many hosts that are inside closer clusters, and passes its traffic to more distant clusters through the distant Mrouter. However, this approach only works passably well for a very simplistic multicast topology (e.g. a linear concatenation of clusters).

In a more general topology the IP multicast forwarding tree away from the requesting MARS Client will branch a number of times, requiring the MARS_REQUEST to be replicated along each branch. Ensuring that the total number of returned group members does not exceed the client's limit becomes rather more difficult to do efficiently. (VENUS could simply halve the limit value each time it split a MARS_REQUEST, but this might cause group member discovery on one branch to end prematurely while all the group members along another branch are discovered without reaching the subdivided limit.)

Now consider this decision making process scattered across all the clients in all participating clusters. Clients may have different limits on how many group members they can handle - leading to situations where different sources can shortcut to different (sub)sets of the group members scattered across the participating clusters (because the IP multicast forwarding trees from senders in different clusters may result in different discovery paths being taken by their MARS_REQUESTs.)

Finally, when the MARS_REQUEST passes a cluster where the target group is MCS supported, VENUS must ensure the ATM address of the MCS is collected rather than the addresses of the actual group members. (To do otherwise would violate the remote cluster's intra-cluster decision to use an MCS. The shortcut in this case must be content to directly reach the remote cluster's MCS.)

(A solution to part of this problem would be to ensure that a VENUS Domain never has more MARS Clients throughout than the clients are capable of adding as leaf nodes. This may or may not appeal to people's desire for generality of a VENUS solution. It also would appear to beg the question of why the problem of multiple-LIS multicasting isn't solved simply by creating a single big MARS Cluster.)

3.2 SVC management - tracking group membership changes.

Once a client's pt-mpt SVC is established, it must be kept up to date. The consequence of this is simple, and potentially devastating: The MARS_JOINs and MARS_LEAVEs from every MARS Client in

every participating cluster must be propagated to every possible sender in every participating cluster (this applies to groups that are VC Mesh supported - groups that are MCS supported in some or all participating clusters introduce complications described below). Unfortunately, the consequential signaling load (as all the participating MARSs start broadcasting their MARS_JOIN/LEAVE activity) is not localized to clusters containing MARS Clients who have established shortcut SVCs. Since the IP multicast model is Any to Multipoint, and you can never know where there may be source MARS Clients, the JOINs and LEAVEs must be propagated everywhere, always, just in case. (This is simply a larger scale version of sending JOINs and LEAVEs to every cluster member over ClusterControlVC, and for exactly the same reason.)

The use of MCSs in some clusters instead of VC Meshes significantly complicates the situation, as does the initial scoping of a client's shortcut during the SVC establishment phase (described in the preceding section).

In Clusters where MCSs are supporting certain groups, MARS_JOINs or MARS_LEAVEs are only propagated to MARS Clients when an MCS comes or goes. However, it is not clear how to effectively accommodate the current MARS_MIGRATE functionality (that allows a previously VC Mesh based group to be shifted to an MCS within the scope of a single cluster). If an MCS starts up within a single Cluster, it is possible to shift all the intra-cluster senders to the MCS using MARS_MIGRATE as currently described in the MARS model. However, MARS Clients in remote clusters that have shortcut SVCs into the local cluster also need some signal to shift (otherwise they will continue to send their packets directly to the group members in the local cluster).

This is a non-trivial requirement, since we only want to force the remote MARS Clients to drop some of their leaf nodes (the ones to clients within the Cluster that now has an MCS), add the new MCS as a leaf node, and leave all their other leaf nodes untouched (the cut-through connections to other clusters). Simply broadcasting the MARS_MIGRATE around all participating clusters would certainly not work. VENUS needs a new control message with semantics of "replaced leaf nodes {x, y, z} with leaf node {a}, and leave the rest alone". Such a message is easy to define, but harder to use.

Another issue for SVC management is that the scope over which a MARS Client needs to receive JOINs and LEAVEs needs to respect the Client's limited capacity for handling leaf nodes on its SVC. If the MARS Client initially issued a MARS_REQUEST and indicated it could handle 1000 leaf nodes, it is not clear how to ensure that subsequent joins of new members wont exceed that limit. Furthermore, if the SVC establishment phase decided that the SVC would stop at a particular

Mrouter (due to leaf node limits being reached), the Client probably should not be receiving direct MARS_JOIN or MARS_LEAVE messages pertaining to activity in the cluster `behind' this Mrouter. (To do otherwise could lead to multiple copies of the source client's packets reaching group members inside the remote cluster - one version through the Mrouter, and another on the direct SVC connection that the source client would establish after receiving a subsequent, global MARS_JOIN regarding a host inside the remote cluster.)

Another scenario involves the density of group members along the IDMR multicast tree increasing with time after the initial MARS_REQUEST is answered. Subsequent JOINs from Cluster members may dictate that a `closer' Mrouter be used to aggregate the source's outbound traffic (so as not to exceed the source's leaf node limitations). How to dynamically shift between terminating on hosts within a Cluster, and terminating on a cluster's edge Mrouter, is an open question.

To complicate matters further, this scoping of the VENUS domain-wide propagation of MARS_JOINs and MARS_LEAVEs needs to be on a per-source- cluster basis, at least. If MARS Clients within the same cluster have different leaf node limits, the problem worsens. Under such circumstances, one client may have been able to establish a shortcut SVC directly into a remote cluster while a second client - in the same source cluster - may have been forced to terminate its shortcut on the remote cluster's Mrouter. The first client obviously needs to know about group membership changes in the remote cluster, whilst the second client does not. Propagating these JOIN/LEAVE messages on ClusterControlVC in the source cluster will not work - the MARS for the source cluster will need to explicitly send copies of the JOIN/LEAVE messages only to those MARS Clients whose prior SVC establishment phase indicates they need them. Propagation of messages to indicate a VC Mesh to MCS transition within clusters may also need to take account of the leaf node limitations of MARS Clients. The scaling characteristics of this problem are left to the readers imagination.

It was noted in the previous section that a VENUS domain could be limited to ensure there are never more MARS Clients than any one client's leaf node limit. This would certainly avoid the need to for complicated MARS_JOIN/LEAVE propagation mechanisms. However, it begs the question of how different the VENUS domain then becomes from a single, large MARS Cluster.

4. What is the value in bypassing Mrouters?

This is a good question, since the whole aim of developing a shortcut connection mechanism is predicated on the assumption that bypassing

IP level entities is always a `win'. However, this is arguably not true for multicast.

The most important observation that should be made about shortcut connection scenarios is that they increase the exposure of any given IP/ATM interface to externally generated SVCs. If there are a potential 1000 senders in a VENUS Domain, then you (as a group member) open yourself up to a potential demand for 1000 instances of your re-assembly engine (and 1000 distinct incoming SVCs, when you get added as a leaf node to each sender's pt-mpt SVC, which your local switch port must be able to support).

It should be no surprise that the ATM level scaling limits applicable to a single MARS Cluster [4] will also apply to a VENUS Domain. Again we're up against the question of why you'd bypass an Mrouter. As noted in [4] Mrouters perform a useful function of data path aggregation - 100 senders in one cluster become 1 pt-mpt SVC out of the Mrouter into the next cluster along the tree. They also hide MARS signaling activity - individual group membership changes in one cluster are hidden from IP/ATM interfaces in surrounding clusters. The loss of these benefits must be factored into any network designed to utilize multicast shortcut connections.

(For the sake of completeness, it must be noted that extremely poor mismatches of IP and ATM topologies may make Mrouter bypass attractive if it improves the use of the underlying ATM cloud. There may also be benefits in removing the additional re-assembly/segmentation latencies of having packets pass through an Mrouter. However, a VENUS Domain ascertained to be small enough to avoid the scaling limits in [4] might just as well be constructed as a single large MARS Cluster. A large cluster also avoids a topological mismatch between IP Mrouters and ATM switches.)

5. Relationship to Distributed MARS protocols.

The ION working group is looking closely at the development of distributed MARS architectures. An outline of some issues is provided in [5,6]. As noted earlier in this document the problem space looks very similar that faced by our hypothetical VENUS Domain. For example, in the load-sharing distributed MARS model:

- The Cluster is partitioned into sub-clusters.

- Each Active MARS is assigned a particular sub-cluster, and uses its own sub-ClusterControlVC to propagate JOIN/LEAVE messages to members of its sub-cluster.

- The MARS_REQUEST from any sub-cluster member must return information from all the sub-clusters, so as to ensure that all a group's members across the cluster are identified.

- Group membership changes in any one sub-cluster must be immediately propagated to all the other sub-clusters.

There is a clear analogy to be made between a distributed MARS Cluster, and a VENUS Domain made up of multiple single-MARS Clusters. The information that must be shared between sub-clusters in a distributed MARS scenario is similar to the information that must be shared between Clusters in a VENUS Domain.

The distributed MARS problem is slightly simpler than that faced by VENUS:

- There are no Mrouters (IDMR nodes) within the scope of the distributed Cluster.

- In a distributed MARS Cluster an MCS supported group uses the same MCS across all the sub-clusters (unlike the VENUS Domain, where complete generality makes it necessary to cope with mixtures of MCS and VC Mesh based Clusters).

6. Conclusion.

This document has described a hypothetical multicast shortcut connection scheme, dubbed `Very Extensive NonUnicast Service' (VENUS). The two phases of multicast support - SVC establishment, and SVC management - are shown to be essential whether the scope is a Cluster or a wider VENUS Domain. It has been shown that once the potential scope of a pt-mpt SVC at establishment phase has been expanded, the scope of the SVC management mechanism must similarly be expanded. This means timely tracking and propagation of group membership changes across the entire scope of a VENUS Domain.

It has also been noted that there is little difference in result between a VENUS Domain and a large MARS Cluster. Both suffer from the same fundamental scaling limitations, and both can be arranged to provide shortcut of unicast routing boundaries. However, a completely general multi-cluster VENUS solution ends up being more complex. It needs to deal with bypassed Mrouter boundaries, and dynamically changing group membership densities along multicast distribution trees established by the IDMR protocols in use.

No solutions have been presented. This document's role is to provide context for future developments.

Security Considerations

Security considerations are not addressed in this document.

Author's Address

Grenville Armitage
Bellcore, 445 South Street
Morristown, NJ, 07960
USA

Email: gja@bellcore.com

References

[1] Laubach, M., "Classical IP and ARP over ATM", RFC1577, Hewlett-Packard Laboratories, December 1993.

[2] G. Armitage, "Support for Multicast over UNI 3.0/3.1 based ATM Networks.", Bellcore, RFC 2022, November 1996.

[3] J. Luciani, et al, "NBMA Next Hop Resolution Protocol (NHRP)", INTERNET DRAFT, draft-ietf-rolc-nhrp-11.txt, February 1997.

[4] G. Armitage, "Issues affecting MARS Cluster Size", Bellcore, RFC 2121, March 1997.

[5] G. Armitage, "Redundant MARS architectures and SCSP", Bellcore, INTERNET DRAFT, draft-armitage-ion-mars-scsp-02.txt, November 1996.

[6] J. Luciani, G. Armitage, J. Jalpern, "Server Cache Synchronization Protocol (SCSP) - NBMA", INTERNET DRAFT, draft-ietf-ion-scsp-01.txt, March 1997

[7] Y. Rekhter, D. Farinacci, " Support for Sparse Mode PIM over ATM", Cisco Systems, INTERNET DRAFT, draft-ietf-rolc-pim-atm-00.txt, April 1996.