Building Shared Trees Using a One-to-Many Joining Mechanism

Ken Carlberg <K.Carlberg@cs.ucl.ac.uk>

Jon Crowcroft <J.Crowcroft@cs.ucl.ac.uk>

Department of Computer Science, University College London
London WC1 6BT, United Kingdom

Abstract -- This paper presents a new approach for building shared trees which have the capability of providing multiple routes from the joining node onto an existing tree. The approach follows a design parameter of CBT and PIM in that it operates independently of any unicast routing protocol. However, a paradigm shift is introduced such that trees are built in an on-demand basis through the use of a one-to-many joining mechanism. In addition, the paper presents optimisations of the new mechanism to help constrain its impact in the case where many receivers exist for a given multicast group.

1. Introduction

As it exists today, IP multicasting is centered on a receiver initiated model for building a delivery tree. This approach allows the one-to-many distribution model of data packets to scale well in the presence of many receivers that have joined the same group. In expanding this model to the case of a many-to-many distribution, wherein many senders are sending data to the same group destination, shared trees have become a standard approach in minimizing the amount of state that needs to be maintained by the network for a given group address.

[1, 2] are two proposals that specify designs to build and maintain a shared tree. These proposals are being advanced in the Inter-domain Multicast Routing working group of the IETF and are designed to operate independent of any underlying unicast routing protocol. A key feature inherent in both PIM and CBT is that they use the unicast routing tables to forward traffic along the branches of a tree. This eliminates duplicate routing tables, for both unicast and multicast, and further contributes to the minimalization of state maintained by the routers of the network. The approach presented in this paper follows this design principle of operating independently of any unicast routing protocol. However, unlike CBT or PIM, it introduces a new mechanism to build a shared tree.

1.1 Background

By default, most unicast routing protocols calculate a single "best" path to a destination. In certain cases, protocols like OSPF [3] provide a choice of equal-cost paths, but these paths are commonly constructed using a single metric, such as the shortest hop count from source to destination. And while work is being proposed in the IETF to provide QoS based routes in [4] and [5], the effort is targeted towards a specific unicast routing protocol or architecture and is not a ubiquitous capability among all routing protocols.

As a consequence, existing efforts like PIM and CBT are constrained to only a single path from a receiver to the core/RP of a tree. In the context of best effort service, this is not a problem. However, the integration of resource reservation protocols like RSVP [6] with CBT or PIM can be problematic when reservation requests exceed the available resources for the single "shortest" path. [7] attempts to address this problem by defining a more malleable mechanism that involves a one-pass-with-advertising (OPWA) reservation scheme. But this effort focuses on altering the reservation request of a single path and does not address the issue of finding other paths that can support the original reservation parameters. The responsibility of this latter aspect, of course, is left to a unicast routing protocol that provides multiple paths between source and destination.

In this paper, we propose a new mechanism for building shared trees that provides (potentially) multiple routes from the joining node onto an existing tree. The approach follows the design parameters of CBT and PIM in that it operates independently of any unicast routing protocol. In addition, the paper presents optimisations of the new mechanism to help constrain its impact on an internet in the case where many receivers exist for a given multicast group.

Section 2 of this paper provides an overview of the architecture and a detailed presentation of a new joining mechanism used to provide multiple routes to an existing tree. Section 3 provides an evaluation of the design approach. Sections 4 and 5 presents new areas of research made available by the new joining mechanism as well as related work in the establishment of delivery trees in an internetwork. Finally, section 6 presents a summary of our design.

1.2 Terminology

A Core and a Rendezvous Point, as described in detail in [1] and [2], are essentially the same entity: a node that acts as the shared root of a distribution tree. For the sake of simplicity, this paper will follow this generalised description. In addition, this paper introduces the notion of egress node and active/inactive root cores. An egress node is an endpoint of an intra-domain branch that goes through the process of discovering on-tree nodes residing in other domains. An active root core is the node that acts as the shared root of a tree. The inactive classification is used for nodes that will become the root core when the active root fails or becomes unreachable. More detailed descriptions will be provided in the following.

2. Architecture

The architecture of the one-to-many joining scheme is split into two tiers; thereby aligning itself with the current intra-inter-domain split of unicast routing. The first tier involves the establishment of the initial intra-domain branch of the shared tree. We say initial because the actions taken in Tier 1 only occur within the domain that has a joining receiver or a non-member sender. Subsequent extensions of the initial intra-domain branch to an inter-domain branch is dealt with in Tier 2 of the architecture[1].

2.1 Tier 1: Initial Intra-Domain Branch

There are two scenarios that trigger the formation of the initial intra-domain branch of a shared tree. One scenario is where an initial receiver joins a group via IGMP [8]. When a leaf router receives the IGMP-join, it uses an out-of-band mechanism to provide a mapping of <egress node, multicast group>. Any out-of-band mechanism can be used, but in our example, we choose DNS [9] because of the query/response nature of its architecture, which obtains information in an on-demand basis[2]. In addition, DNS allows us to take advantage of an existing resolution mechanism within a localised (i.e. intra-domain) environment. It should be stressed that this approach of using DNS is restricted to only building an intra-domain branch. It is not used at any other point in the continued construction of the branch among domains.

Upon receiving the mapping, the leaf router issues a join towards the appropriate egress node, thereby installing state and building an intra-domain branch from the leaf router to the egress point of the domain. When the receiver leaves the group, the branch between the leaf router and the egress node is torn down. Figure 1 presents a visualisation of the cascading join; wherein the receiver in Domain C issues an IGMP join to the leaf router, which in turn issues an intradomain-join to one of the two egress nodes.

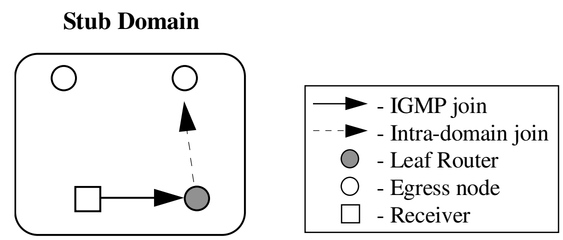


Figure 1: Formation of Intra-Domain Branch

Another scenario that triggers the formation of an intra-domain branch is when a non-member sender within the domain sends traffic to the destination group address. In this case the leaf router receiving the traffic obtains the <egress node, group> mapping and issues a join-request to the egress node. However, because the join-request is generated by a non-member sender, the leaf router associates a timer with the instantiated branch. When data stops flowing to the leaf router from the sender, the branch is timed out and torn down by the leaf router.

In the case where DNS is used for intradomain <egress node, group> mappings, information such as the association of blocks of multicast addresses to specific egress node(s) can be stored in an a priori manner or through dynamic DNS updates. In the former case, configuration files can be set in the authoritative server and periodically downloaded to other servers within the domain. In the latter approach, dynamic DNS updates, as specified in [11], can be used to update DNS servers and clients in near real-time fashion.

2.2 Tier 2: Building an Inter-Domain Branch

The previous section focused on instantiating a branch within the domain that contains a receiver or non-member sender. This section describes the use of a one-to-many inter-domain join issued by a egress node to other nodes residing in other domains. The purpose of this action is to locate other on-tree nodes; which in turn can provide multiple paths from the joining node to the existing tree. For the sake of simplicity, we start by assuming that at a minimum, there exists an inter-domain root core for the target group address. Later, we describe the series of actions taken when there is no pre-existing tree/root.

---

[1] Future research is expected to extend the initial two tier design to that of N levels.

[2] Other out-of-band schemes, like the Session Directory (SD) protocol [10] can be used for providing <core,group> mappings.

2.2.1 Spanning Joins

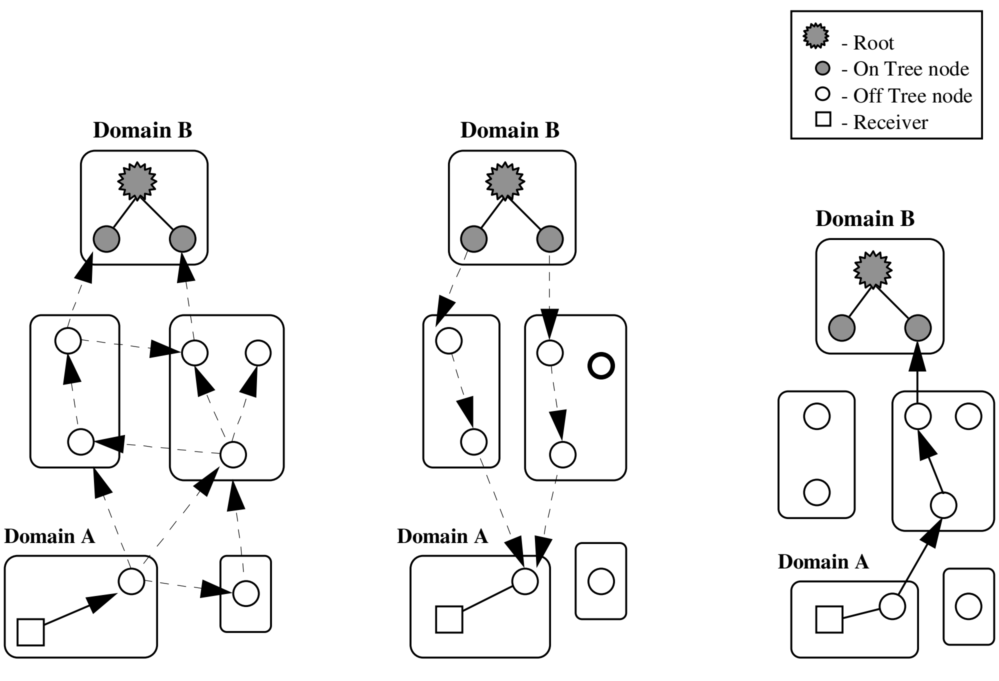
A spanning join is a one-to-many joining mechanism that creates a source based spanning tree emanating from the joining node to any on-tree nodes. The spanning tree used for inter-domain joins is built by an algorithm that uses broadcast with Reverse Path Forwarding (RPF) [13, 14, 15]. This combination of algorithms allows optimal trees to be built from other nodes towards the source and filters out duplicate packets without requiring state to be maintained for each source of the join. In the case of spanning joins, it is important to note that pruning is not used to because only a single stateless control message is being sent.

Since this join is issued indiscriminately for the purpose of discovery, there is no need to select a target root core. This approach represents a paradigm shift in comparison to PIM and CBT, which rely on unicast requests being sent to a single core/RP coupled with another mechanism to advertise and select the target in an a priori fashion. In a sense, a source-based tree in the control plane is used to build a shared tree for the data plane.

Inter-domain (one-to-many) joins are generated only by egress nodes and are sent to a well-known multicast address. As described above in subsection 2.1, intra-domain join-requests are issued by leaf routers to a egress node. The absence of state in this node for the target group address triggers the generation of an inter-domain spanning join.

If a node that receives an inter-domain join has no state for the target group address, it broadcasts the message out to its other interfaces via RPF. When the message is received by an on-tree node that has state for the group, the message is terminated and the on-tree node responds with a unicast join-request towards the egress node that initiated the one-to-many join. This join-request installs temporary state along the path towards the initiator of the one-to-many join. In the case where several on-tree nodes receive the broadcast message, several join-requests will be sent to the originating egress node. Thus, the originator can have several inter-domain paths to choose from in grafting a branch onto an existing tree. Once the path is chosen, the initiator of the one-to-many join acknowledges a join-request and the other potential paths are either timed out or explicitly torn down. This ability to discover multiple routes is accentuated with wide topological distributions of group membership.

Figure 2 provides a three part illustration of discovering an existing tree and instantiating a branch onto the tree. In part (a), the leaf router in Domain A sends a join-request to the egress node, which in turn issues a spanning join to all the other nodes. Part (b) shows that when the one-to-many join reaches on-tree nodes in Domain B, join-requests are unicast back to egress node that originated the flood. Finally, in part (c) egress node chooses one of the paths and sends a confirm message towards one of the responding on-tree nodes.



Part (A) *Part (B)* *Part (C)*

Figure 2: Example of One-to-Many Join

2.3 Constraining the Impact of Spanning Joins

In section 2.1, we described how spanning joins are only issued by an egress node, which allows us to aggregate these control messages on a domain basis. The following introduces other measures that constrain the impact of one-to-many joins.

2.3.1 Expanding Rings

An expanding ring is the default mechanism in issuing one-to-many joins. Its purpose is to constrain the impact of source based spanning joins and to employ an incremental discovery scheme to locate any existing shared tree. The Time To Live (TTL) field is used to define the diameter of the ring. The value for this field is set in blocks of N, as opposed to single incremental values so as to minimise the delay of the discovery process.

2.3.2 Directed Spanning Joins

A derivation of the one-to-many join mechanism is referred to as Directed Spanning Joins (DSJ). Its goal is to further reduce the impact of source initiated spanning trees and to "direct" it towards a target destination. At the same time, these joins are also designed to discover potential multiple paths between the initiator and the target destination.

As was mentioned earlier, broadcast using RPF is the basic tenet of building a source based spanning tree that distributes an inter-domain join. The inclusion of TTL is an added measure that bounds the inter-domain join into a series of expanding ring searches.

DSJ, as outlined in this paper, takes a different direction in minimising impact of one-to-many joins. In this approach, a unicast address and a routing metric is included in the join message, together with the broadcast and RPF algorithm, as a means of constraining branches of the spanning tree. The destination unicast address denotes the target of the spanning tree; for example, a well known root. The routing metric measures the 'current' distance of a parent node to the destination unicast address. This criteria is used by the child node to determine if it should continue the broadcast & RPF algorithm, or terminate the message. In its basic form, the Directed Spanning Join algorithm is as follows:

> Step 1: The initiator floods the join message to all directly connected downstream neighbours (i.e., children). This message contains the unicast address of the destination and the routing metric that represents the distance between itself and the destination.
>
> Step 2: Upon receipt of a broadcasted join message, each receiving node checks if it has state for the multicast group. If it does, then the one-to-many join message is terminated and a join-request is sent to the initiator. Otherwise, proceed to step 3.
>
> Step 3: Use RPF to determine if the message was received on an upstream interface. If it wasn't then the join message is terminated with no additional action. Otherwise, continue to step 4.
>
> Step 4: Compare the routing metric in the join message with the 'current' routing metric to the target destination. If the 'current' metric is less than or equal, to the metric stored in the message then: a) update the metric in the join message with the 'current' metric, and b) broadcast the message to downstream interfaces. Otherwise, terminate the join message and take no further action.

Other than adding information into the join message, Steps 1, 2, and 3 involve the same responsibilities of the one-to-many joining mechanism presented earlier in section 2.2.1. However, Step 4 expands the criteria by adding a unicast destination, such as a known on-tree node or the root of the inter-domain tree, and 'current' metric to that destination. This additional information helps constrain the topological width of the spanning tree. In addition, as the directed spanning tree gets closer to the destination unicast address, the edges become pruned at a greater rate.

Figure 3 presents an abstract view of a directed spanning tree. The oval represents the affected region of the directed join. All nodes within the region receive the join, while all nodes outside the region never receive the broadcasted message.

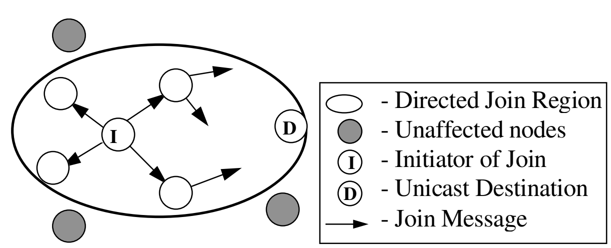


Figure 3: Directed Flood

It should be noted that directed joins do not use TTL values since the above algorithm is already designed to limit the scope of the flood. In addition, this mechanism requires a priori information (e.g., the target unicast address) in order to provide a reference point for the directed join.

2.4. Establishing a Root Core

Root cores are established indirectly when the very first receiver joins a group. As mentioned previously, when a receiver joins a group, an intra-domain branch is established from the receiver to the egress node. In turn, this node uses a one-to-many join in an attempt to graft its intra-domain branch onto an existing tree. However, in the absence of an existing tree, the egress node that initiates the inter-domain join becomes, by default, the root core for that multicast group. Subsequent spanning joins from egress nodes in other domains graft their intra-domain branches to the root core set up by the initial receiver of the multicast group. This on-demand establishment of a root is a departure from the scheme used by both PIM and CBT, which rely on a priori configurations or delegations.

2.4.1 Election of Non-Active Root Core(s)

To protect against failure, non-active root cores are elected by the current root core. When the current root core fails or is no longer reachable, branches on the tree are grafted to one of the non-active root cores, which in turn elects a new set of non-active root cores.

The election process involves the discovery of other on-tree nodes by the root core. This is accomplished by a scoped Hello sent down the tree and an accompanying response sent back to the root core. The root then sends a list of non-active root cores to all the nodes of the tree. This list is sent on an infrequent basis.

3. Evaluation of Design Approach

One of the distinguishing characteristics of a one-to-many join mechanism is that a joining node has the potential of discovering alternate paths from itself to the shared root of the tree. This allows an inter-domain multicast routing protocol to select the "best" route from itself to the tree, regardless of whether the underlying unicast routing protocol only calculates a single route from source to destination.

As part of this discovery process, information can be exchanged between the existing tree and the joining node. This information can consist of typical unicast routing metrics like delay or hop count to the root core from the edge of the existing tree. But it can also be expanded to include multicast related QoS, such as current fan-in/fan-out values or average delay to the edges of the tree.

In the case of advertising existing network resources, mechanisms like OPWA, as presented in [7], can be extended from the tree to the joining node. And more significantly, OPWA can be accomplished independent of any unicast routing protocol that may or may not support multipath routing. Thus, the "best" route from the joining node to the tree is influenced by factors other than a single shortest hop unicast route.

Another characteristic of the one-to-many joining mechanism is that a separate <root core, group> mapping mechanism is not needed for inter-domain branches. The mapping is inherent in the constrained broadcasting scheme. In addition, the system is capable of supporting extremely large numbers of on-tree nodes in a system, thus minimising the amount of load concentrations to a specific node as well as providing a potentially high degree of optimal branches stemming from the root.

Part of the rationale for using scoped one-to-many joins, instead of Hierarchical PIM (HPIM) [12], centers on the ability of an egress node to join the nearest or farthest on-tree node. Thus, a egress router can directly join the root core or any other node on the tree. In addition, there is no need for deciding in an a priori fashion which hierarchical level a given core/RP is associated with. Finally, spanning joins are generated in an on-demand basis as opposed to periodic flood & prune transmissions sent by HPIM routers.

However, the trade-off of using a one-to-many join mechanism vs. an HPIM scheme is that large number of receivers may adversely impact the system. This impact is reduced to some extent by restricting the source of one-to-many joins to just egress nodes, as opposed to leaf routers directly connected to receivers. In addition, the use of scoped floods, via incremental TTL values, constrains the flooding scheme to that of an expanding ring discovery.

Another means of reducing the system wide impact of constrained broadcasting is through the use of directed spanning joins. By directing the source initiated spanning tree towards a known target, the joining node retains the on-demand discovery capabilities of receiver initiated broadcasting. In addition, the impact of one-to-many joins is limited to those nodes that are more or less topologically between the joining node and the target node.

4. Topics for Future research

The ability to exchange information between a joining node and an existing tree opens a number of areas for future research. One example would be the inclusion of policies, both rational and irrational, that determine how multicast QoS/ToS is used to select the "best" branch of the shared tree. We define rational policies as those that describe network related characteristics like the number of fan-in or fan-out links of a node on the tree. Irrational policies are defined as those that are not directly related to the capability of a network -- such as pricing, inclusionary/exclusionary access lists, etc.

Another aspect that requires future investigation is the ability to efficiently support non-member senders. As has been presented, the one-to-many joining mechanism can be used to instantiate a branch from the non-member sender to an existing node on the tree. However, the delay generated by the discovery process may be intolerable for applications that may send infrequent and short streams of data. Therefore, it would seem apparent that optimisations of the one-to-many architecture are needed to support a wide range of characteristics stemming from different applications.

Finally, migration of the two-tier architecture to one that is aligned with N-levels of hierarchy will be beneficial in terms of aggregation of multicast groups. This is particularly acute in the case where an internet supports tens of thousands of multicast groups; many containing few members which are widely dispersed topologically.

5. Related Work

The construction of shared trees over any unicast routing protocol has been a subject of design for several years in the IETF community and is evidenced in the on-going work of [1] and [2]. The primary objective of these approaches is to build a shared tree and reduce the state information maintained by on-tree nodes to just <multicast group>. A common attribute in both approaches is the use of unicast joins to instantiate a branch on the shared tree. This is coupled with out-of-band mechanism(s) that require a priori configurations to either directly or indirectly dictate the topological placement of the inter-domain shared root of a tree.

However, both of these approaches have a number of design characteristics that distinguish one from the other. One example is the use of soft-state vs. hard-state to provide some measure of fault tolerance of the tree. Another example is the placement of receivers onto a tree. HPIM connects receivers to the lowest depth of a shared tree, while CBT can connect receivers to any level of the tree -- including directly onto its shared root.

Earlier work in the construction of multicast trees for IP networks came in the form of source-based trees, as presented in [13].. A foundation of this effort centers on the use of a modified Reverse Path Forwarding (RPF) algorithm[3] to build a separate tree for each source sending traffic to the multicast group. To avoid loops, <source, multicast group> state is maintained by each node on the tree. A subsequent modification to reduce the impact of source-based trees came in the form of explicit pruning from nodes on the tree that had no members of the multicast group. Thus, streams of data would only be sent down branches of the tree that contained group receivers [15].

6. Summary

This paper has presented a new approach that builds shared multicast trees using a one-to-many joining mechanism. A fundamental goal of this approach is to provide multiple paths from a joining node to an existing tree.

The design of this new mechanism is divided into a two-tier approach, which is meant to constrain the impact of one-to-many joins in the presence of many receivers as well as easily align itself with the current inter- and intra-domain split in unicast routing. In this approach a single node within a domain acts as a proxy for propagating receiver initiated joins, via constrained broadcasting, throughout other domains. Hence, a certain amount of aggregation of control messages is achieved when several nodes within a domain join the same destination group address.

An attractive feature derived from a one-to-many join approach is that the selection of the shared root is accomplished in an on-demand basis. For nodes joining the group, there is no need for a priori configuration and advertisement of cores/RPs nor is there a need for an out-of-band mechanism to map <root core, group> pairings.

Another design feature is the ability of a joining node to exchange information with on-tree node(s). This allows the joining node to select the branch that best satisfies its desired QoS requirements.

[3] Developed by Dalal and Metcalf in [14]

Acknowledgements

The authors would like to thank Tony Ballardie as well as Mark Handley and Nadia Kausar, of UCL for general discussions on the ideas presented in this paper.

References

[1] Deering, D. Estrin, D. Farinacci, M. Handley, A. Helmy, “Protocol Independent Multicast - Sparse Mode (PIM-SM)”, Internet Draft -- Work in Progress, July, 1996.

[2] Ballardie, “Core Based Trees (CBT) Multicast Architecture”, Internet Draft -- Work in Progress, February, 1996.

[3] Moy, “OSPF Version 2”, RFC 1584, March 1994.

[4] Crawley. Presentation at Montreal IETF, June 1996.

[5] Castineyra, J. Chiapa, M. Steenstrup, “The Nimrod Routing Architecture”, Internet Draft - Work in Progress, February, 1996.

[6] Braden, L. Zhang, S. Berson, S. Herzog, S. Jamin “Resource Reservation Protocol -- Version 1 Functional Specification”, Internet Draft - Work in Progress, May 1996

[7] Shenker, L. Breslau, “Two Issues in Reservation Establishment”, Proceedings of ACM Sigcomm ‘95, August, 1995.

[8] Deering, “Host Extensions for IP Multicasting”, RFC1112, August 1989.

[9] Mockapetris, “Domain Names - Concept and Specification”, RFC 1035, Nov, 1987

[10] Handley, V. Jacobson, “SDP: Session Directory Protocol (draft 2.1)”, Internet Draft -- Work in Progress, February 1996.

[11] Vixie, S. Thomson, Y. Rekhter, “Dynamic Updates in the Domain Name System”, March, 1996.

[12] Handley, J. Crowcroft, I. Wakeman, “Hierarchical Protocol Independent Multicast (HPIM)”, white paper, October, 1995.

[13] Deering, “Multicast Routing in a Datagram Internetwork.” PhD thesis, Stanford University, California, USA, 1991.

[14] Dalal, R. Metcalf, “Reverse Path Forwarding of Broadcast Packets”, Communications of the ACM, December 1978.

[15] Pusateri, “Distance Vector Multicast Routing Protocol”, Internet Draft -- Work in Progress, June 1996.