A SIGNALING FRAMEWORK FOR HYBRID MULTICAST TREES

Radu IORGA[1], Eugen BORCOCI[2]

This paper proposes a management framework that allows enhanced signaling capabilities offering tight control over the construction and maintenance of multicast hybrid trees in IP communications. Hybrid trees are composed of native IP multicast trees inter-connected by unicast Overlay Trees. Inline with Future Internet architectures, that add powerful management entities, the proposed framework divides the computation and installation of the multicast trees in several layers, increasing the scalability and providing flexibility in the sense that, customized trees can be established on demand, conforming to the needs of the Service Provider which uses these trees for media flow distribution.

Keywords: signaling, multicast, hybrid multicast, overlay

1. Introduction

Nowadays the majority of communications (in terms of traffic volume) on the internet are dedicated to multimedia applications like IPTV and VoD as shown in [1][2][3][4]. These applications require group communications in order to minimize the needed network resources. IP multicast is a mature technology, over two decades old, but it is not largely deployed because of issues related to router capabilities, group management, QoS guarantees and inter-domain spanning [5]. Some of those drawbacks, like inter-domain or needed router capabilities are partially solved by the Application Layer Multicast (ALM) [6] approach that delegates the multicast tree constructions to the end nodes. ALM completely ignores the network resources and capabilities and more, it exposes some resource inefficiency by sending the same packet multiple times on some links. A combination of the two approaches, trying to take the advantages of each one is the concept of hybrid multicast trees [7], [8], [9]. The current hybrid multicast solutions existing in the literature are usually limited to inter-connect the IP islands and do not treat the native IP multicast tree construction. The signaling methods of the current solutions are best emphasized in the IETF initiative to hybrid multicast named Automatic IP Multicast without Explicit Tunnels (AMT)

[1] Ph.D. student, Faculty of Electronics Telecommunications and Information Technology, University POLITEHNICA of Bucharest, Romania, e-mail: riorga@elcom.pub.ro
[2] Prof., Faculty of Electronics Telecommunications and Information Technology, University POLITEHNICA of Bucharest, Romania

[7]. The basic idea is to tunnel the Internet Group Management Protocol IGMP [10] join messages from one IP island to another thus expanding the IP multicast tree. The present paper proposes a framework that can be used by protocols aiming to construct both the overlay trees and the native IP multicast trees. The main advantage of this approach is the end-to-end control that can be obtained over the combined multicast tree allowing also the creation of QoS capable trees.

Multicast trees built under management supervision need a sequence of control messages to be exchanged between various entities. The sequence of actions is triggered by a request for a hybrid tree that can span multiple IP multicast domains. The request comes from an entity generically called Service provider, which is supposed to exploit this tree for media flow distribution. In an abstract view of the framework, the IP capable domains will be named *IP islands* while the inter-domain tree spanning from the traffic source to the end-nodes will be named *Overlay Tree (OT).* The OT is connecting the islands at data plane by tunneling the native multicast packets, or in an alternative evolved architecture presented in [11], by rewriting the IP address. The decision on how to treat the packets is also enforced with the help of control messages.

The proposed framework, named *Multicast Signaling Management Framework* (*MSMF*), is flexible enough to allow the addition of any information that might be needed to construct a custom tree and poses no restrictions to the creation of only one of the type of trees available. The first section of the paper shows a proposed high-level architecture that can take advantage of the framework together with the possible sequence of actions that lead to a complete installation of a hybrid tree in the network. The format of the signaling message used is presented in the third section, followed by an example of how the framework capabilities cab se used to construct and install a tree.

2. Management entities

Fig. 1 shows the high level architecture of the proposed signaling framework. The notations used are: OTM – Overlay Tree Manager; IPM – IP Multicast Manager; Requester – A manager (usually the Service Provider [11]) that aggregates the users' interests and decide to construct a hybrid tree. Because of the hybrid nature of the solution, a hierarchical approach has been chosen to decouple the creation of one type of trees from the other. This also allows for a flexible usage in the sense that one can create only one type of tree if desired. Although OTM and IPM can be physically located on the same computer, the recommendation is that are implemented as two logically separated modules. The two managers are deployed one for each domain. If the existence of an IPM for each IP island (administratively separate domain) comes natural, the usage of one

OTM for each island is based on architectural reasons. Such a deployment ensures that one hybrid tree is easily extendable to multiple native IP capable islands.

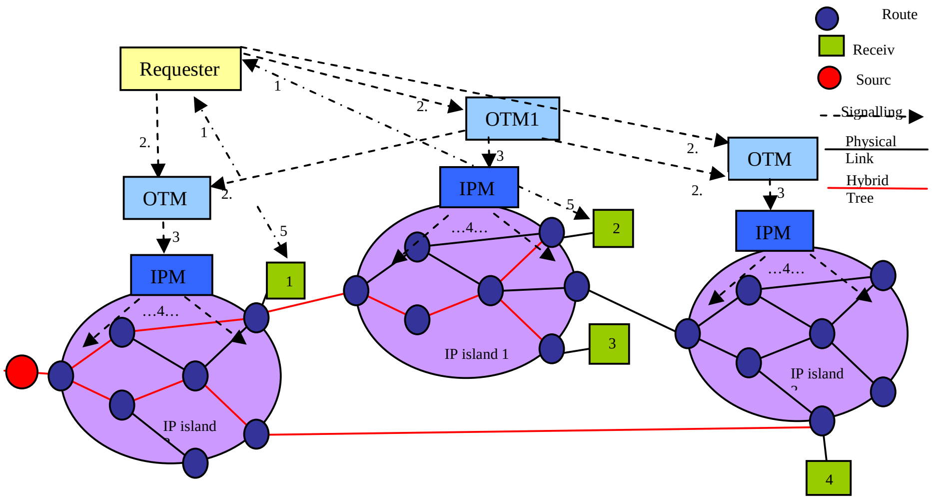

Fig. 1. High-Level view of the signaling framework

Action 1 in the above figure is used by the end clients to express their interest in some content and it is outside the *MSMF* scope as it is more of a business operation between users and different providers. It is presented here only to show the complete sequence of actions.

Based on an aggregated decision actually collecting the end-users interest depicted by actions like 1 (or based on some other forecast information- not specified in this work), the requester can take the initiative to construct a hybrid tree connecting multiple IP islands. In a simpler scenario, not presented in this paper, all the clients are located inside the same IP island. The subsequent actions taken by the requester of the tree are marked with 2.x in Fig. 1. If the requester has knowledge on multi-domain topology of the underlying network, than it can compute the tree itself and enforce the results (actions 2.1) on the involved OTM. However knowing the inter-domain topology information is an unusual and high burden for the requester. It is most likely that requester will to delegate the tree computation to other entities (more “network-aware”) that are capable to discover the topology. Since a distributed computation is very difficult, one OTM, usually the one associated with the IP island where the source is connected, is chosen to be the initiator of the tree. The Action 2.1 but this time is directed and used to request a multicast tree from the *initiator OTM*. The latter will compute the tree and signal the extension of the tree to its pairing OTMs (actions 2.2). In the given architectural example the forwarding model used for the signaling action 2.2 is not explicitly presented, but *MSMF* is able to accommodate both the following: a) the hub model, which implies that the tree overlay tree is entirely computed at the initiator and then the initiator communicates with all involved OTM and signal them the result of the computation; b) the cascade model, in which it is supposed that the initiator does not have the full topology so it computes the tree only in relations with its connected neighbors, and delegates the rest of the tree computation and construction to the neighboring OTMs. Actions 2.x can be bi-directional if certain information needs to be sent back to the requester (i.e., the group IP address to be used for the requester tree, which will be later used by end users to JOIN the IP multicast tree)

Once the overlay tree has been computed and all OTMs know the new tree configuration, they will determine the local terminations and translate them to local receivers (or source) for the IP multicast tree and the request is transmitted to IPM (action 3). This manager than computes the local tree and sends the configuration commands to the routers (actions 4). The last action (action 5) is again out of *MSMF* scope and it is used by the requester to inform the end-users about the newly created tree (i.e., the group address).

3. Signaling message format

The format of the message that is exchanged as part of the signaling actions is presented in Fig. 2. The name, Multicast SLS, is derived from the well-known Service Level Specification [12] concept and extended to multicast environment. The source and the leaves of the tree are mandatory and they are specified in the form of IP addresses. The “any” parameter means that any of piece of information (or a combination of them) could be used to specify the source or the receivers. More, the leaf nodes, which most likely will be routers connecting Access Networks (AN) to the core domains, can be instructed about the IP multicast capabilities of the ANs using the “distribution method” parameter. Because of the hybrid nature of the trees, where traffic needs to be translated from IP multicast to unicast environments, a traffic identification method needs to be communicated from different level managers all the way to the routers. Several pieces of information are transmitted inside MSLS which can be used to uniquely identify a packet: Source IP, Group IP, Protocol, Source Port and Destination Port. One special note should be made related to the Group IP that should be used to identify the tree. This parameter might be blank at the first step, and its allocation can be delegated to the OMT that initiates the tree. Optionally, QoS parameters can be transported by MSLS. The parameters, like bandwidth, delay, packet loss or jitter, can be used to construct and install QoS aware hybrid multicast trees. More, there is also the possibility, if desired, to impose the treatment that should be applied to the non-conformant traffic (i.e., excess traffic can be dropped with a given priority or can be marked). The format of the signaling message has been designed to be open to any desired modifications.

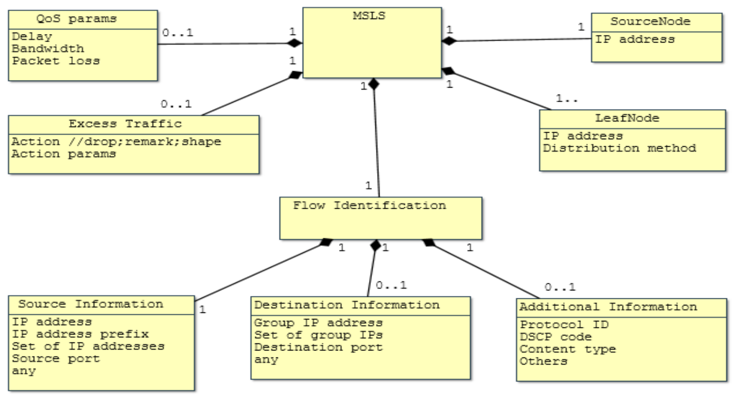

Fig. 2. The signaling message general format

4. An example of end-to-end signaling

The Message Sequence Chart in Fig. 3 exemplifies the actions presented in the above section. The numbers in parenthesis at the end of the MSLS primitives represent the action number as presented in Fig. 1.

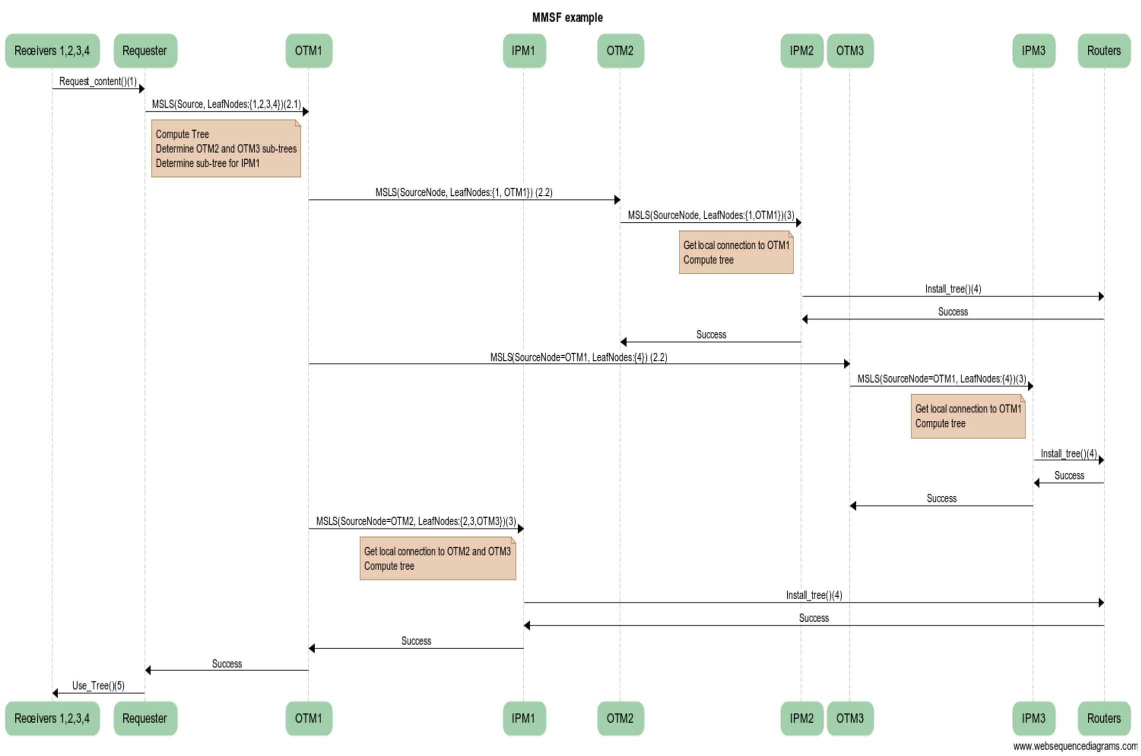

Fig. 3. Example of signaling

After aggregating all the requests for content, the Requester start to construct a hybrid multicast tree by sending a request to an OMT entity (in this example it is OMT1), using the MSLS format in which the source and the four receivers are specified. Assuming that the hub model has been chosen, OTM1 will compute the overall tree and will determine the sub-tree to send to each of the pairing overlay manager. OTM2 will receive the MSLS with SourceNode unchanged, but with LeafNodes changed to nodes that have local significance to OTM2. Receiver 1 is already interested in the content, but the router connected to OTM1 is also considered a receiver so that the hybrid tree can be extended to the next IP island. The determination of the local router inside island 2, connected to IP island 1 can be realized either by OTM2 or by IPM2 as depicted in our scenario. Regardless of this design choice, once IPM2 has the needed information it will compute the tree and install it in the routers. Action number 4 can use any well known management protocol like SNMP [13], web-services [14] or even a proprietary protocol. The same logic applies to OTM3: it will receive a MSLS with SourceNode changed to OTM1, as this will be the local source of the IP tree and with LeafNode 4 as this is the only receiver in IP island 3. Again, the local router connected to OTM1 needs to be determined and the tree is computed by IPM3 and installed in the routers.

One last thing done by OTM1 is to send the MSLS to local IPM1. In the case of this IP multicast tree, the traffic will come from IP island 2 and the receivers are both local (2 and 3) and remote in island 3. IPM1 will know determine the two routers connecting to islands 2 and 3, compute the tree and install the tree.

Our measurements in Fig. 4 showed that the number of routers does not influence the total installation time. The measurements were made using a prototype code implemented in C under Linux and used on a physical topology similar to the one presented in Fig. 1. We have chosen to use a physical and not a simulated topology use the installation time is, in our view, is significantly influenced by the network or node loads. The testing procedure involved measuring the time that our implementation took between the moment the request has been received and the moment when a acknowledgment is received that the tree has been installed in the routers. Several points of measurements have been established: 1) OTM1 which is the initiator of the tree; 2) IPM1; 3) IPM2; 4) IPM3. The time is measured in microseconds, but the point of interest is the tendency shown by the time needed to install the tree. We have implemented a multi-threading environment allowing parallel communication with each router and we obtained the expected result of having roughly the same time.

It can be seen in Fig. 4 that, if the tree is not installed on any router, the duration is very small. Once we begin to install the tree on increasing number of routers it can be seen that the overall time becomes saturated and does no increase anymore. This could be explained by the parallel processing we have implemented. An interesting observation was made at IPM1 and IPM2. While increasing the number of routers, it has been noticed that the duration decreased slightly. This is due to the network and computers load which decreased exactly when we performed the measurement, and proves that the decision to use a physical setup to try to observe the qualitative behavior of the solution, was correct.

However, the installation of the tree can become very time expensive in a scenario with an IP island having many routers inside. Even if a parallel approach is followed, like in our prototype, a large number of routers can pose scalability problems related to IPMx CPU and memory limitations and, additionally, it can lead to an increase of the total time, foreseen because of the shared structures that need to be protected by semaphores. And the use of semaphores, in an environment with multiple threads, highly increases the probability of one thread waiting for others to complete. The above can be solved by increasing the resources inside the management entities and by using a lock free programming method. Another possible idea is to divide the IP islands in smaller areas that are easier manageable, but this implies both architectural problems (who is taking care of the smaller areas) and may bring out other scalability issues by adding a new level of signaling between IPM and this new manager.

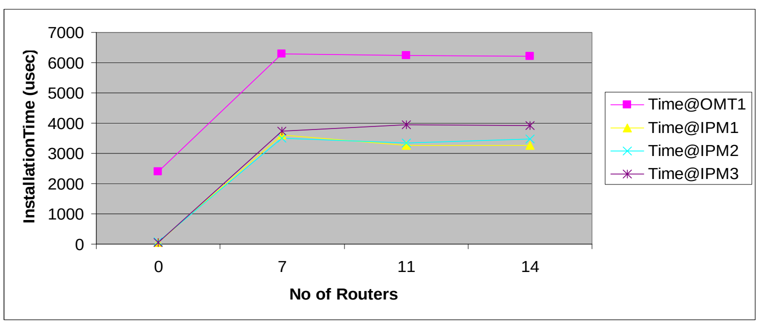

Fig. 4. Installation time related to number of routers

After the requester gets the status of the tree (successful in the presented scenario) it will inform the end-users if and how to use the installed tree in order to receive the desired content.

5. Conclusions

A *Multicast Signaling Management Framework* has been developed in this paper aiming to offer a viable solution for media distribution over multiple domains in multicast mode, with QoS guarantees. Note that the available solutions are lacking the end-to-end control that is necessary in multimedia environment. The architecture presented in section 2 shows that multiple domains, capable to support native IP multicast, can be interconnected using OT with the help of the *MSMF*.

The format of the signaling message allows for an inter-domain extension of a tree by offering the possibility to explicitly announce who the source and receivers are without any restriction related to their global position. The uniqueness of the tree in an inter-domain environment is achieved by the flow identification fields as shown in section 3.

By proposing a hierarchical approach of signaling, *MSMF* is able to improve the scalability of a multi-domain solution and also solves the possible political issue of the rights to modify the configuration of the internal routers of a domain by a third external party.

Future work needs to be done in order to determine if another hierarchical level can be added beneath IPM level, thus reducing the number of messages sent to routers, and also increase the response time in case of network failures.

R E F E R E N C E S

[1] *Ning Wang; Yan Zhang; Serrat, J.; Gorricho, J.L.; Tao Guo; Zheng Hu; Ping Zhang*, "A two-dimensional architecture for end-to-end resource management in virtual network environments," Network, IEEE, vol.26, no.5, pp.8-14, September 2012.

[2] *N.M. Mosharaf Kabir Chowdhury, Raouf Boutaba*, A survey of network virtualization, Computer Networks, Volume 54, Issue 5, 8 April 2010, Pages 862-876, ISSN 1389-1286, 10.1016/j.comnet.2009.10.017.

[3] *J. Pan, S. Paul, R. Jain*, “A survey of the research on future internet architectures”, IEEE Communications Magazine, vol.49, no.7, pp.26-36, July 2011.

[4] J.*Choi, J.Han, E.Cho, T. Kwon, and Y.Choi*, A Survey on Content-Oriented Networking for Efficient Content Delivery, IEEE Comm. Magazine, March 2011.

[5] *H. Asaeda et.al.*, "Architecture for IP Multicast Deployment: Challenges and Practice", IEICE Transactions on Communications, Vol. E89-B, No. 4, 2006, pp. 1044-1051.

[6] *M. Hosseini, D. T.Ahmed, S. Shirmohammadi, and N. D. Georganas*; “Survey of Application-Layer Multicast Protocols” - http://www.discover.uottawa.ca/publications/files/CST.pdf

[7] *D. Thaler, M. Talwar, A. Aggarwal, L. Vicisano, T. Pusateri;* “Automatic IP Multicast Without Explicit Tunnels (AMT)” – internet draft 2010 - http://tools.ietf.org/html/draft-ietf-mboned-auto-multicast-10

[8] *B. Zhang, S. Jamin, L. Zhang*, “Host Multicast: A Framework for Delivering Multicast To End Users”, *IEEE INFOCOM*, 2002

[9] *H. Luo and K. Harfoush;* “Adaptive Hybrid Multicast with Partial Network Support” – http://www4.ncsu.edu/~kaharfou/Papers/hybrid_multicast.pdf

[10] RFC 3376 - Internet Group Management Protocol, Version 3 – 2002 – http://tools.ietf.org/html/rfc2236

[11] ALICANTE, Deliverable D2.1 ALICANTE Overall System and Components Definition and Specifications, http://www.ict-ALICANTE.eu/, Sept 2011

[12] *D. Goderis et all*; “ Service Level Specification Semantics, Parameters and negotiation requirements” - http://www.ist-tequila.org/presentations/dg-ietf48.pdf

[13] http://en.wikipedia.org/wiki/Simple_Network_Management_Protocol

[14] http://en.wikipedia.org/wiki/Web_service