

# Management Driven Hybrid Multicast Framework for Content Aware Networks

Radu Iorga<sup>\*1</sup>, António Pinto<sup>†‡2#</sup>, Eugen Borcoci<sup>\*3</sup>, Gustavo Carneiro<sup>†</sup>, Radu Miruta<sup>\*4</sup>, Tania Calçada<sup>†§5</sup>

<sup>\*</sup>Telecommunication Department, University Politehnica of Bucharest, Bucharest, Romania

<sup>1</sup>riorga@elcom.pub.ro, <sup>3</sup>eugen.borcoci@elcom.pub.ro, <sup>4</sup>radu.miruta@elcom.pub.ro

<sup>†</sup>INESC TEC (formerly INESC Porto), Porto, Portugal

<sup>§</sup>Faculdade de Engenharia, Universidade do Porto, Portugal

<sup>5</sup>tcalcada@fe.up.pt

<sup>‡</sup>CIICESI, ESTGF, Polytechnic Institute of Porto, Portugal

<sup>2</sup>apinto@estgf.ipp.pt <sup>#</sup>corresponding author

**Abstract**—The need for a better adaptation of networks to the transported flows, conducted research to new approaches such as Content Aware Networks and Network Aware Applications. In parallel, recent developments of multimedia and content oriented services and applications such as IPTV, Video streaming, Video on Demand, and Internet TV, reinforced the interest on multicast technologies. IP multicast has not been largely deployed due inter domain and QoS support problems and therefore alternative solutions have been investigated. This paper proposes a management driven hybrid multicast solution that is multi-domain, media oriented and combines overlay multicast, IP multicast, and P2P. The architecture is developed in a content aware networks and network aware applications environment, based on light network virtualization. The multicast trees can be seen as parallel virtual content aware networks, spanning a single or multiple IP domains, customized to the type of content to be transported while fulfilling the quality of service requirements of the Service Provider.

**Keywords**—Hybrid Multicast, Content Aware Networks, Resource Management

## I. INTRODUCTION

The recent, and strong, orientation of Internet towards services [1], has led to a closer coupling between the transport/network and service/application layers [2] aiming to increase the overall efficiency through a cross layer optimization. This can be achieved by making networks more aware of the transported content [3] - *Content Aware Networks (CAN)*, or by making applications more aware of network conditions - *Network Aware Applications (NAA)*.

In parallel, recent developments of multimedia and content oriented services (e.g. IPTV, Video streaming, Video on Demand, and Internet TV), reinforced the interest on *multicast technologies*. IP multicast has not been globally deployed due to problems related to group management, router capabilities, inter-domain transport and lack of Quality of Service (QoS) support [4]. Overlay multicast, despite its lower efficiency, has emerged as an alternative [5]. In a complex scenario, a hybrid multicast, combining IP multicast with overlay multicast, can be attractive in terms of scalability, efficiency and flexibility [6].

Another trend, aiming to overcome the current Internet ossification by creating customized flexible networks, is to use *network virtualization* [7]. New business entities (see Fig. 1), named Virtual Network Providers, can offer customized virtual networks. In particular, Services Providers (SP) can deploy their services on top of some hired virtual networks without the burden of performing connectivity control. Such virtualized transport service can be deployed by Network Providers (NP), either enhanced to become Virtual Network Providers, or by cooperating with separate new entities which offer network virtualization. However, each NP still manages its own infrastructure. While full network virtualization is challenging in terms of seamless deployment, more “light” solutions can be attractive, by being deployed as parallel data

planes [8], logically separated, but under the coordination of a single management and control plane. In [9], several research challenges, related to management and control plane, are identified. The proposed solution addresses some of them. In particular, the guaranteeing of service availability in accordance to a pre-established Service Level Agreement (SLA), guaranteeing QoS, supporting large scale service provisioning and deployment, enabling of a higher integration between services and networks, and the capability of accepting new, activated on-demand, services.

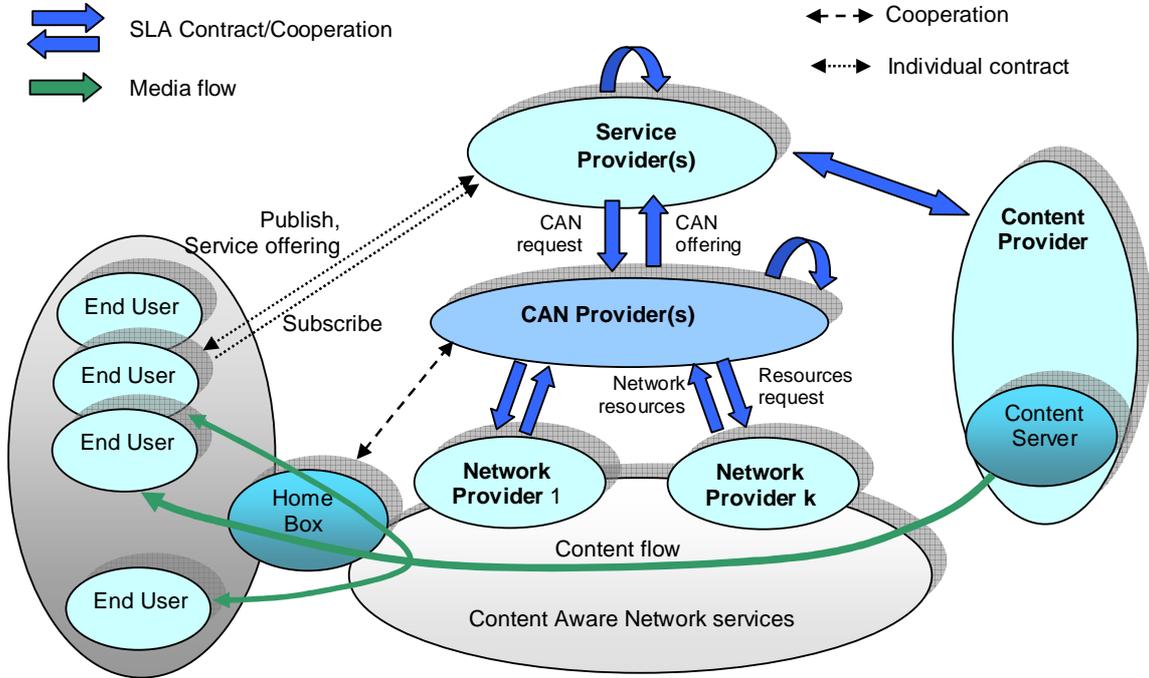


Fig. 1 Interactions between business actors in CAN oriented architecture

This work combines a *light network virtualization* of the data plane, combining IP multicast, overlay multicast and P2P communications in order to create a *management driven hybrid multicast* framework, embedded in a *CAN/NAA* architecture, capable of being deployed over multiple, QoS capable, IP domains. These virtual CAN data planes will be called *VCANs* and are to be constructed and offered by CAN Providers (CANP). The CANP can be seen as an enhanced NP. SP will only offer high level services, and Content Providers (CP) will provide the content. The HomeBox is a residential gateway placed in the end users premises, used by SP for content and service publishing and distribution, and by end users to connect to CANs and consume services [2].

A multicast-capable VCAN can be associated to a customized multicast tree (denoted *mVCAN*). For a given network technology (e.g DiffServ), an *mVCAN* is implemented by using the network technology QoS mechanisms (classification, queuing, scheduling, shaping). The proposed solution coordinates *mVCANs* planning and implementation, upon request from one or several SPs. The infrastructure consists in one or several IP networks, which are managed by distinct business entities, and support some form of content awareness.

Two major problems were identified. The first problem consists in planning and mapping *mVCANs* onto several network domains, while meeting the NP policies and resource availability on one side, and the SP needs in terms of traffic demands, topology information and QoS requirements, on the other. The second problem is the realization of the data plane elements and functionality to support such multicast-enabled transport.

Multi-domain mVCANs are computed by an algorithm that combines QoS constrained routing, admission control (for the SP requests), logical resource reservations (based on network resources made available and published by the supporting NPs). The NP independency is preserved (important business requirement) in terms of managing their own resources; however their cooperation is promoted. NPs are not required to disclose their internal topologies and capacities to third parties. The mapping algorithm works in two phases: inter-domain mapping and intra-domain mapping. Finally, each NP decides its intra-domain mapping. Business entity cooperation is based on dynamically negotiated Service Level Agreements/Specifications (SLA/SLS). The multicast services offer three levels of QoS guarantees: fully managed (strong); partially managed (statistical), and unmanaged services (no guarantees). Therefore, the proposed architecture can support all kinds of applications.

The proposed solution, while not being fully content oriented as the “clean slate” ones in [3], offers seamless deployment without the scalability problems of a full content oriented solution. CPs negotiate with SPs in order to assure content delivery to end users. SPs negotiate with CANPs in order to request inter-domain delivery and intra-domain content aware connectivity services. CANPs negotiate with NPs in order to request intra-domain content delivery. End users negotiate with SPs in order to request content. The routers that construct the VCAN are named *Media Aware Network Elements (MANE)*. Their content awareness (CA) is realized by: 1) configuration - the control plane instructs MANEs on how to treat different types of flows; and 2) metadata - content servers of the CP may insert content description information into the data packets. MANEs are only placed at the edge of network domains has their additional tasks related to content aware treatment of the network flows may be CPU-intensive.

The described CAN architecture was adopted by the ALICANTE European FP7 ICT research project [2]. A preliminary hybrid multicast architecture has been proposed in [6]. This work further refines its functionalities, algorithms, and design. It also outlines some implementation choices and solutions.

The paper is organized in sections. Section II presents an overview of the proposed solution. Section III presents the mVCAN construction and identifies resource management issues. Section IV details the data plane design. Section V presents a summary of the testbed used in validation. Section VI contains some conclusions and future directions.

## II. HYBRID MULTICAST OVERVIEW

The hybrid multicast architecture has two key functional entities: the CAN Manager (CANMgr), and the Intra-domain Network Resource Manager (IntraNRM). A one-to-one mapping between CANMgrs and IntraNRMs has been adopted. The distributed management and control plane [9] enables large scale provisioning capabilities and supports the integration of independent NPs in multi-domain scenarios. IntraNRM can either be evolved to include CAN management functionalities or a separated entity can be deployed. A business entity CANP could own one or more CAN Managers.

An SP can request, using a CANMgr (initiator CAN Manager or *iCANMgr*), the creation of a mVCAN, supplying the tree topology and QoS requirements. The *iCANMgr* then computes (phase 1) an overlay tree spanning multiple domains, delegating the intra-domain multicast tree computation (phase 2) to the IntraNRM of each domain. The *iCANMgr* may negotiate with other CAN Managers if required (multi-domain VCANs). The *iCANMgr* requires inter-domain topology knowledge; mechanisms for discovery of such topology exist [2], but they are not within the scope the current work. If the replies of all involved CANMgrs are satisfactory, the SP is informed and the SLS is concluded. Later, the SLS is enforced at SP request and the resources are allocated in the network equipment (MANEs and core routers) using specific signaling: SP → *iCANMgr* → (other CANMgrs if multi-domain) → IntraNRM → routers (MANE and core routers).

Each mVCAN has a given class of service. For instance, in multi-layer Scalable Video Coding (SVC) flows, each SVC layer can be associated with a mVCAN whose leaves are the MANEs connected to the receivers subscribing that layer. Receiver heterogeneity is supported and adaptation accordingly to

network conditions, by dropping packets from higher layers when and where necessary, may be performed. SVC flow adaptation, while possible [2], is also not within the scope of this work.

### III. MULTICAST VCAN

#### A. Inter and Intra-domain Multicast VCAN Construction

Fig. 2 shows an example of two inter-domain mVCANs, each one requested by a different SP (1 and 2). Each CANMgr receiving the request becomes the iCANMgr of the respective mVCAN. The IntraNRMs of each Core Network Domain (CND) have not been represented; they are assumed to be deployed in the CANMgrs. The SPs provide a summary of the desired tree (the root and the leaf nodes) and its QoS requirements; the rest is computed by the iCANMgr. The VCAN1 spans  $CND_k$ ,  $CND_m$  and  $CND_n$ . The equivalent tree has the root in  $CND_k$  (MANE connected to AN1) with the MANE egress routers of  $CND_k$  (AN2), of  $CND_m$  (AN3 and AN4) and of  $CND_n$  (AN6) as leaves. Another example is VCAN2 which has the root in  $CND_n$  (AN6) and as leaves the MANEs in  $CND_n$  (AN5),  $CND_m$  (AN4) and  $CND_k$  (AN2). The iCANMgrs are usually associated with the domains where VCAN1 and VCAN2 roots are placed,  $CND_k$  and  $CND_n$ , respectively.

Fig. 2 also includes a summary of the management actions required to establish VCAN1 (represented by black, dashed and numbered arrows). SP1 requests VCAN1 to an iCANMgr<sub>k</sub> (action 1). Then, in *phase 1*, the iCANMgr<sub>k</sub> determines which domains could participate in the tree and computes the inter-domain multicast tree using a QoS constrained routing algorithm that respects the SP constraints (see Section III.B). A first level mapping onto the multi-domain topology is performed and resources are logically reserved. Next, iCANMgr<sub>k</sub> negotiates, in a hub model, with the other CANMgrs (actions 2.1, and 2.2) in order to finalize the requested mVCAN.

The hub model is a simpler solution (versus chain/cascade model) appropriate for the establishment of multicast trees spanning multiple domains. The management advantage is that iCANMgr has complete control over the multi-domain mVCAN construction, but on the other hand it requires knowledge of the inter-domain topology. The number of domains that may be involved in a mVCAN is rather low (less than 10, due to the Internet tier-oriented architecture) and tend to be localized in an Internet region. The related scalability issues, due to management signaling, are not so stringent. Thus, other logical chaining of signaling, such as the cascade model, where deemed not appropriate for mVCAN establishment (it is more adapted to 1-to-1 pipes).

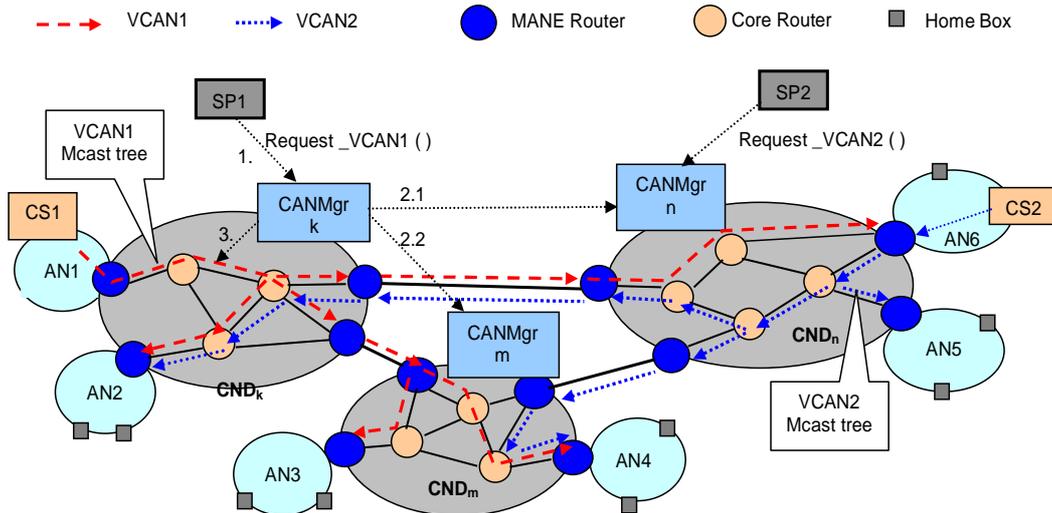


Fig. 2 Hybrid multi-domain multicast trees examples (VCAN1 and VCAN2) and summary of management actions

In *phase 2*, after computing the first level tree, the iCANMgr asks each other CANMgr to solve its part of the tree. The intra-domain sub-trees of the mVCAN are computed by the IntraNRM of each domain. The algorithm used to compute the intra-domain multicast tree is the same as for inter-domain case. The mVCAN is finalized when its configuration is sent to the MANEs and core routers (action 3.). Core routers are not required to be media aware, but need to support IP multicast with QoS constraints. A low frequency of mVCAN requests is expected.

### B. Multicast VCAN Mapping Algorithm

The algorithm for computing the multicast trees is a modified version of the Dijkstra *Shortest Path Tree (SPT)* algorithm, *applied twice*, in the two phases described above, with a special additive metric to allow for QoS constrained routing. It also performs SP requests admission control and logical reservation of resources. The adopted metric is  $1/A_k$ , where  $A_k$  is the available bandwidth of the link  $k$ . Other, more complex, metrics could be adopted. An alternative metric, which minimizes the utilization of a path, is the cost of a link  $k$ , defined as  $1/(1-U_k)$  where  $U_k = R_k/A_k$  ( $R_k$  representing the required bandwidth for the link  $k$ ). Higher link utilization implies a higher link cost. The system is flexible since it allows each SP to specify a customized metric for its mVCANs. The algorithm will minimize the overall network utilization for a given request.

The algorithm is summarized below. Traffic Matrix (TM) is supposed to be requested by SP as  $TM(\text{root}, \text{leaves\_list}, \text{Breq})$ . The notation *//text* represents comments and DJ means Dijkstra algorithm.

1. Compute the DJ\_SPT (root) where; //Routing metric  $\sim 1/B_{ij}$
2. Select the TM branches that can be satisfied (i.e  $B_{ij} > B_{req}$ ); // Mapping and Admission Control
3. Reserve capacities for these branches (e.g. by subtraction); // reduced graph
4. Compute the overall utilization for each path reserved:  $U_{\text{path}} = \text{Sum\_links} (B_{req}/B_{\text{avail}})$
5. List the unsatisfied branches;
6. Compute VCAN utilization (sum over all paths mapped onto the real graph);

The solution proposed is scalable. The Dijkstra's original algorithm runs in  $O(|V|^2)$  complexity where  $V$  is the number of vertices. If the algorithm is changed to be based on a min-priority queue, implemented by a Fibonacci heap, a complexity of  $O(|E| + |V| \log |V|)$ , with  $E$  being the number of edges, is achieved. The Dijkstra is the fastest SPT algorithm for arbitrary directed graphs with nonnegative weights. The algorithm now proposed will have  $n \cdot O(\text{Dijkstra})$  complexity for inter-domain tree establishment, where  $n$  is the number of the requested mVCANs. The domains are represented as vertices in the inter-domain graph. Remember that the mVCANs are not frequently requested. For intra-domain, the algorithm has the same complexity as link-state routing protocols based on Dijkstra algorithm.

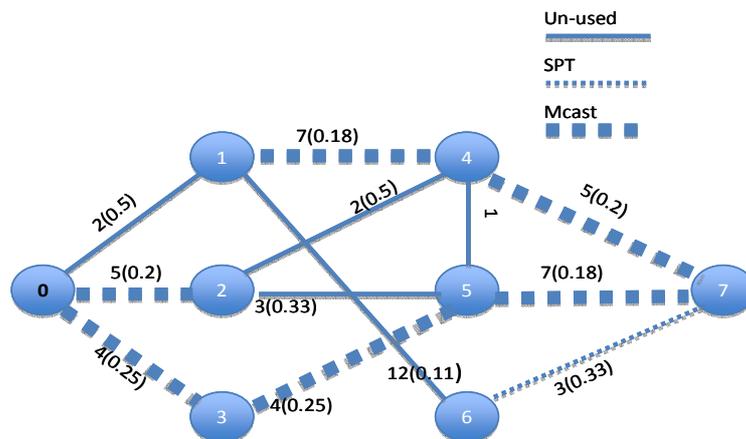


Fig. 3 An example topology and its resulting multicast tree

Fig. 3 presents an example of a network graph. Given the same approach used for inter and intra-domain, each node can represent a domain (when in phase 1) or a router (when in phase 2). The number on each link represents the available bandwidth. The additive metric is  $1/A_k$  (shown in parentheses on each link). The SPT is represented as dotted lines. Branches with no foreseen receivers, or that do not belong to a path to a receiver (e.g. node 6), can be pruned. Future SP requests for new mVCANs will be treated using the reduced graph (after reservation). When a mVCAN is deleted (or pruned) the released bandwidth is added to the respective branches of the graph.

#### IV. DATA PLANE

##### A. Multicast bridge

The MANE data plane multicast module is named Multicast bridge (see Fig. 4) and, in short, re-transmits the received multicast packets multiple times. Multicast bridges receive their configuration from the IntraNRM. Upon receiving its configuration, the Multicast bridge builds a multicast forwarding table. It can perform three types of output functions of the received multicast packets: Inter-domain output; Intra-domain output; P2P output.

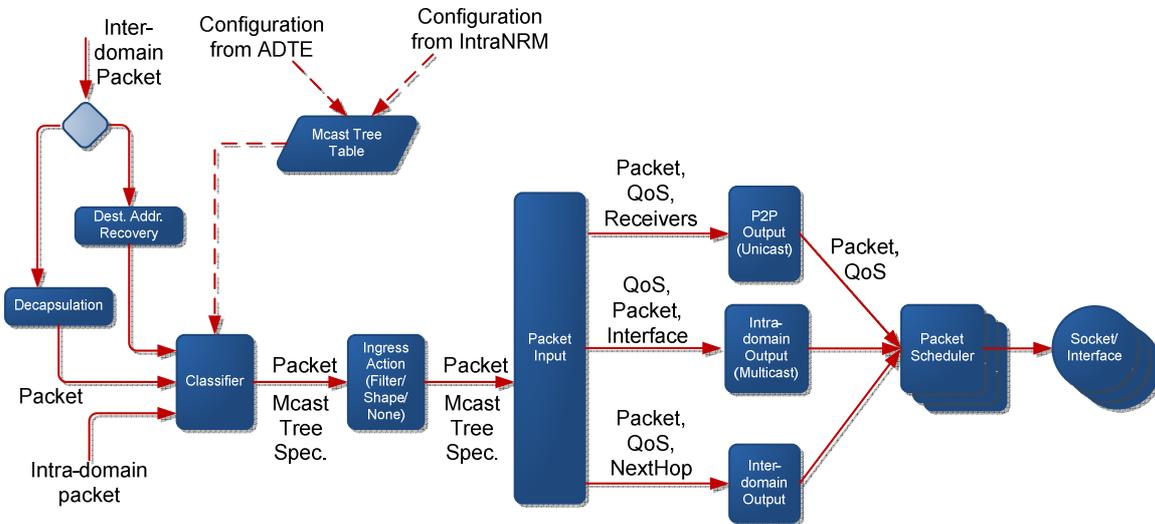


Fig. 4 Multicast Bridge architecture

Intra-domain output is used if the domain supports IP multicast and once the required negotiation between the CANP and IntraNRM is concluded. Information like *source IP address*, *group IP address*, *input interface*, *output interfaces*, and *QoS information* is obtained in this negotiation. Afterwards, MANE's are able to process and forward IP multicast packets correctly. If the domain does not support IP multicast, the received unicast packets will be send as is.

##### B. Inter-domain multicast

Inter-domain output is used by MANE to transmit packets across inter-domain links, to a peering MANE, as unicast. Two forms can be used: UDP encapsulation or address rewrite. In UDP encapsulation, the original multicast IP packet is encapsulated, including the full IP header and payload, in a new UDP packet with the next hop MANE as destination. In the address rewrite mode, the destination IP address of the packet is rewritten as the IP unicast address of the next hop MANE. The original address will be recovered when received at the destination MANE, based on content-related metadata present in the packet.

The two approaches have different tradeoffs. In the first case, the packet encapsulation incurs in some packet size overhead, MTU reduction and extra processing time. The address rewriting erases the original

multicast address and the next hop MANE will require some form of recovering the original IP multicast destination address. The decision on which case to use is performed on per-packet and based on the perceived ability of the next hop MANE to recover the original IP multicast destination address. Multicast address recovery is possible when the MANE is able to identify the flow to which the packet belongs, based on flow information metadata inserted, by the Content Server, in the data packets.

Support for both SVC media flows and in network adaptation, increases the complexity of the flow identification. For instance, all SVC layers of a given content belong to the same VCAN but the tree used for each layer differs. Traditionally, the base layer reaches all interested end users, while enhancement layer reach subsets of those, either because some tree branches were pruned due to local congestion, or because of lesser end user capabilities. Thus, the VCAN identifier (which is found inside the flow information) alone is not enough to uniquely identify the multicast tree. In such case, it is required to look into the information present in the SVC NAL units to determine the layer.

### C. Peer-to-peer Multicast

P2P output will be used by MANE to stream contents in unicast to a subset of all HomeBoxes (HBs) in a P2P enabled domain. This subset will become the P2P seeding HBs, while the remaining HBs will request contents from the subset HBs in P2P. The HBs are explicitly informed of the availability of P2P in their domain, done via signaling procedure between HB and SP upon its initial negotiation with CANP. CAN Managers interact with HBs to accept requests for the P2P content, for which one of these two possible roles (P2P seeder, P2P client) is selected and replied to the requesting HB. From the Multicast bridge point of view there is no distinction between unicast transmissions to a group of HBs, and unicast transmission to the subset of HBs that are P2P seeders.

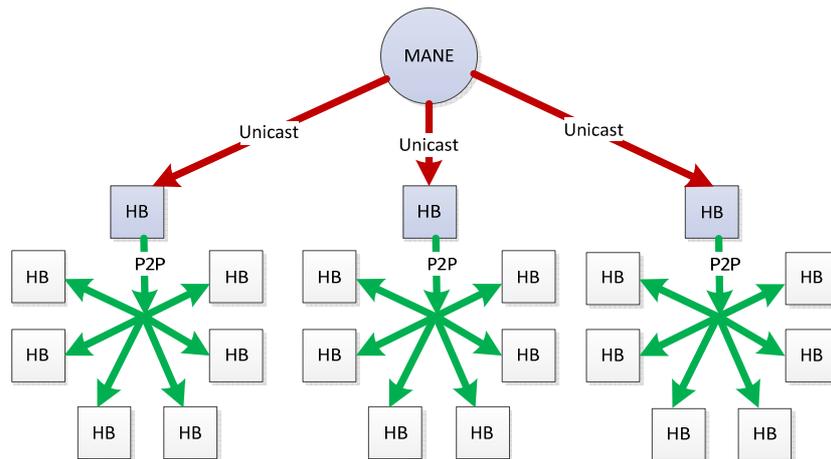


Fig. 5 Peer-to-peer multicast distribution forming clusters.

When the SP negotiates with a CANP, it may request P2P assistance instead of IP multicast. In this case, for any future HBs requests in that domain, the reply will indicate P2P as the distribution method. As the number of HBs increases, the topology that is formed by this method is a cluster-based one, as shown in Fig. 5. One out of N HBs receives the stream directly from the MANE, and then seed a copy of the content to each cluster. MANEs only provide a fraction of bandwidth. The algorithm to determine the ideal value of N is an open research issue.

## V. VALIDATION

The proposed solution was functionally validated and implemented. Prototype modules were setup in a testbed similar, in topology, to that of Fig. 2, and tested in order to demonstrate the main concepts. The Linux implementation of the described modules runs in user space and uses multithreading. The

implementation avoids copying packet payloads as much as possible. Although processing packets in user space inevitably implies two packet copies - one when the packet is received, and another when the packet is transmitted - additional copies of the packet were avoided. Given that management entities involved in the proposed multicast architecture are remote software entities, placed in different core domains, and their communication is not a real-time one, web services were used for communication.

#### A. Multicast VCAN Validation

An experimental testbed composed of the elements required for the installation of the VCAN1, shown in Fig. 2, was setup. The testbed contained three core network domains, each one having a CAN Manager and an IntraNRM. All nodes were Linux routers with IP multicast and QoS support enabled. The managers were collocated in the same physical machines as the routers. The implementation was made using C under Linux. MANE routers have been placed at the edges of the core domains, in order to classify the packets conforming to their content-types, and to forward them to the appropriate QoS enabled trees, which were established in advance by the management framework.

Additional information on implementation and validation results can be found at <http://www.ict-alicante.eu/validation/use-cases/>. There, a *Multicast Live TV* use case, based on the proposed solution, is described. Several parallel trees (seen as distinct mVCANs) have been constructed to support SVC media flows [2].

#### B. Multicast VCAN Efficiency Validation

The hybrid multicast data plane efficiency was evaluated using the metric defined in [10]:  $\delta = 1 - (\text{multicast hops} / \text{unicast hops})$  and the scenario depicted in Fig. 2, where AN3 and AN4 support P2P multicast, AN6 only supports unicast, and AN2 supports IP multicast. Each AN is assumed to be a 5-ary balanced tree (with  $\text{height}=2$ ) comprising 30 clients ( $c=30$ ). The number of packet forwarding operations that occur in a AN, when using unicast transport, is obtained by  $h_{uni} = \sum_{k=1}^{\text{height}} kc^k = 55$ , whereas for the P2P multicast transport is calculated by  $h_{P2P} = (rh_{uni}) + ((1-r)c) = 35$ , (assuming an  $r=15\%$ ). Finally, IP multicast transport is obtained by  $h_{mcast} = c = 30$ .

The number of total unicast and multicast hops for each access network are given respectively by  $H_{uni} = ((d+1)c) + h_{uni}$  and  $H_{mcast} = (2d+1) + h_{type}$ , where  $d$  is the number of standard routers interconnecting the domain to CS1, and  $type$  is the transport of the domain. For AN2  $d=4$ ,  $H_{uni}=145$ ,  $type=mcast$ ,  $H_{mcast}=33$ ; for AN3  $d=7$ ,  $H_{uni}=235$ ,  $type=P2P$ ,  $H_{mcast}=41$ ; for AN4  $d=8$ ,  $H_{uni}=265$ ,  $type=P2P$ ,  $H_{mcast}=42$ ; for AN6  $d=8$ ,  $H_{uni}=265$ ,  $type=uni$ ,  $H_{mcast}=62$ . The multicast efficiency obtained in this example was  $\delta = 1 - (173/910) = 80.4\%$ .

Fig. 6 shows the multicast efficiency of the proposed hybrid multicast architecture in a broader set of scenarios, assumed closer to real world implementations. This results assume that each domain has an AN represented by a 10-ary balanced tree, that all ANs in a given domain support the same transport type, and that in each domain there are 2 core routers interconnecting the ingress and the egress MANEs. The considered scenarios are threefold: 1) all ANs support IP multicast; 2) all ANs support P2P multicast but not IP multicast; and 3) all ANs used only pure unicast transport. In any case, hybrid multicast was always considered for inter-domain connections (VCAN). The number of domains was not specified, instead, the mean number of transit domains of the VCAN was adopted.

Fig. 6(a) shows how the efficiency evolves while increasing the number of clients per domain. Two sets of curves are presented for VCANs with either 4 or 30 transit domains, on average. Within each set, when multicast is supported by ANs (either IP or P2P), the efficiency tends to improve with the increase of the number of clients per domain. When using only unicast, the increase in clients results in a small efficiency

improvement while the AN tree height is 2; for higher tree heights, the efficiency in the core network is not enough to overcome the inefficiency in the ANs, leading to a decrease in the overall multicast efficiency. The rate of this decrease depends on how populated the last level of the tree is, as can be seen by the wavy shape of the unicast lines.

Fig. 6(b) shows how the efficiency evolves while increasing the mean number of transit domains. Two sets of curves are presented for ANs with either 20 or 20000 clients. Within each set, VCANs with more transit domains have a higher multicast efficiency since a higher number of packet forwarding operations are avoided when compared with the pure unicast transport.

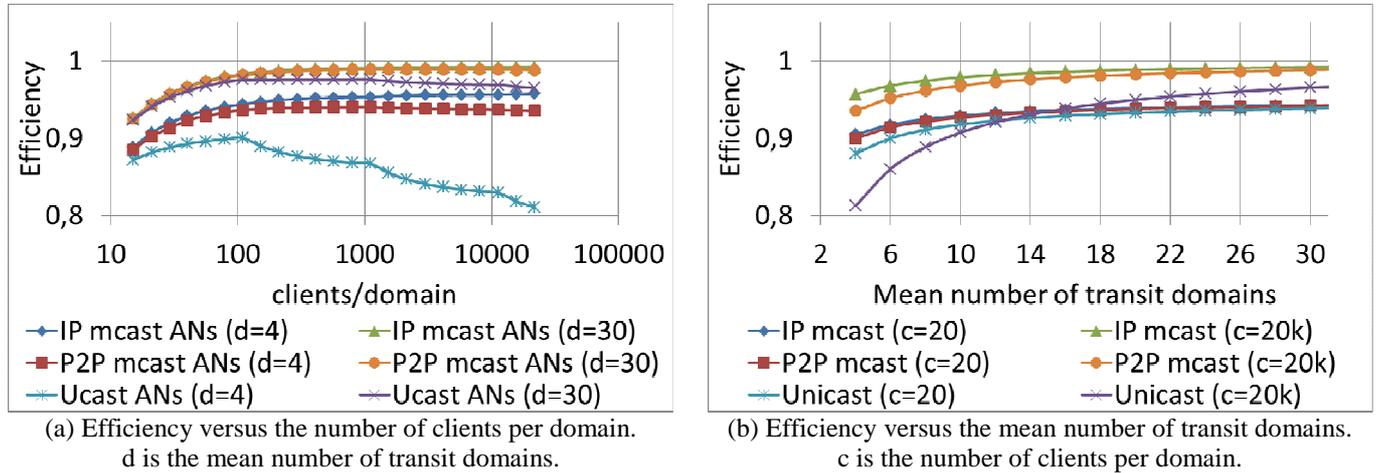


Fig. 6 Efficiency of the proposed hybrid multicast architecture.

## VI. SUMMARY

The paper continued and further developed a management driven hybrid multicast framework based on the light network virtualization of the data plane, that combines IP multicast, overlay multicast and P2P, embedded in a CAN/NAA architecture, capable of being deployed over multiple, QoS capable, IP domains. A signaling system for multi-domain VCAN management is proposed. Optimized inter-domain and intra-domain mapping of VCANs, onto several domain network resources is realized, by using a combined algorithm for QoS constrained routing, mapping and resource reservations for multicast trees. Data plane solutions are developed by introducing a novel element, the Multicast bridge, to solve intra and inter-domain transport. Implementation aspects and achievements of the IP multicast and overlay multicast are discussed and validation results are presented.

The current work contributed to finalize the integration of the modules that support the multicast architecture within ALICANTE system, followed by the performance evaluation of the overall system.

## ACKNOWLEDGMENT

This work was supported, in part, by the EC in the context of the ALICANTE project (FP7-ICT-248652) and, in part, by the national Romanian project POSDRU/88/1.5/S/61178.

## REFERENCES

- [1] J. Schönwälder, et. al., "Future Internet = Content + Services + Management," *IEEE Commun. Mag.*, vol.47, no.7, pp.27-33, Jul. 2009.
- [2] H. Koumaras, et al., "Media Ecosystems: A Novel Approach for Content-Awareness in Future Networks," in *Future Internet*, Springer, 2011, pp.369-380.
- [3] J. Choi et al., "A Survey on Content-Oriented Networking for Efficient Content Delivery," *IEEE Commun. Mag.*, vol.49, no.3, pp.121-127, Mar. 2011.
- [4] H. Asaeda et al., "Architecture for IP Multicast Deployment: Challenges and Practice," *IEICE Trans. Commun.*, vol.89, no.4, pp.1044-1051, Apr. 2006.

- [5] L. Lao et al., "A Comparative Study of Multicast Protocols: Top, Bottom, or In the Middle?," Proc. IEEE Int. Conf. Computer Communications (INFOCOM), USA, 2005.
- [6] E. Borcoci et al., "Hybrid Multicast Management in a Content Aware Multidomain Network," Proc. Int. Conf. Advances in Future Internet (AFIN), France, 2011.
- [7] N. Wang et al., "A two-dimensional architecture for end-to-end resource management in virtual network environments," IEEE Network, vol.26, no.5, pp.8-14, Sep. 2012.
- [8] M. Boucadair et al., "A Framework for End-to-End Service Differentiation: Network Planes and Parallel Internets," IEEE Commun. Mag., vol.45, no.9, pp.134-143, Sep. 2007.
- [9] A. Galis et al., "Management and Service-aware Networking Architectures (MANA) for Future Internet - Position Paper: System Functions, Capabilities and Requirements," Proc. Int. Conf. Communications and Networking (ChinaCOM), China, 2009.
- [10] R. Chalmers and K. Almeroth, "Developing a multicast metric," Proc. IEEE Global Telecommunications Conf. (GLOBECOM), USA, 2000.

Radu Iorga has a Ph.D degree in Telecommunications from University "Politehnic" of Bucharest, Romania. He participated in three European Research Projects as a research scientist. His current research interests include Future Internet architecture, networks and services management, communication protocols, QoS, networked media and CON/CCN. He authored or co-authored several scientific publications and research reports. Currently, he is a software engineer working for LUXOFT professional Romania, deployed as a contractor for IBM STG.

António Pinto received his Licenciatura (2000) in computer science from Polytechnic of Porto, MSc (2005) in communication networks and services from Porto University, and PhD (2010) degree in Electrical and Computers Engineering from Porto University. Currently, he is an assistant professor at Escola Superior de Tecnologia e Gestão de Felgueiras (ESTGF) of the Polytechnic of Porto, where he gives courses in computer networks and operating systems. He is also a researcher of the Wireless Networks Group of the Telecommunications and Multimedia Unit, at INESC Porto.

Eugen Borcoci has a Ph.D degree in Telecommunications from University "Politehnica" of Bucharest, Romania. He participated as team leader to several national research projects and European Research Projects, including FP5, FP6, and FP7. He authored or co-authored over 130 scientific publications and research reports. His current research interests include Future Internet architecture, networks and services management, communication protocols, QoS, networked media and CON/CCN. Currently, he is a professor at the Electronics, Telecommunications and Information Technology Faculty of the University "Politehnica" of Bucharest (UPB).

Gustavo Carneiro received a Diploma degree in electrical and computer engineering from Porto University, Portugal, in 2001, and since then has been a researcher at INESC Porto. He actively participated in the European research projects IST ARROWS, IST DAIDALOS (1 and 2) and ICT ALICANTE. He received an MSc diploma in 2006, and a Ph.D. in 2012, both at Porto University, Portugal.

Radu Miruta graduated at Electronics, Telecommunications and Information Technology of the University "Politehnica" Bucharest (UPB) and received his Ph. D in Telecommunications in 2013 at UPB. He participated in the Alicante FP7 research project and is author or co-author of 14 research papers and scientific reports. His work and interest are in content aware networks, routing algorithms with QoS constraints, content aware packet classification, SDN. He is currently employed by ERICSSON Romania.

Tânia Caçada received a Licenciatura (1999) and a PhD (2013) degree in Electrical and Computer Engineering from Porto University. Currently, Tânia Caçada is a research assistant at Universidade do Porto, working for the Future Cities EU project in the area of urban scale sensor networks. Previously, was research engineer at INESC Porto in the area of wireless networks related to wireless mesh networks and vehicular networks. Her research interests include wireless networks, delay tolerant networks, intelligent transport systems and data gathering.