A Generalized Group Communication Network Stack and its Application to Hybrid Multicast

Matthias Wählisch†
Freie Universität Berlin
Institut für Informatik
Email: waehlisch@ieee.org

Thomas C. Schmidt
HAW Hamburg
Dept. Informatik
Email: t.schmidt@ieee.org

Georg Wittenburg
Freie Universität Berlin
Institut für Informatik
Email: wittenbu@inf.fu-berlin.de

***Abstract*—Group communication services are most efficiently implemented on the lowest layer available. Network layer multicast transparently delegates group distribution to the link layer wherever possible. Native multicast deployment, though, has been mainly limited to 'walled gardens' within provider domains. Overlay multicast overcomes these deployment restrictions on the price of a performance penalty. Current activities focus on hybrid approaches which dynamically combine multicast in overlay and underlay, and adaptively optimize group communication. The basic requirement for such a flexibly deployable architecture is a layer-transparent group communication stack that integrates variable multicast protocols by a common API. In this paper, we present a common group communication stack which serves the requirements of data distribution and maintenance for multicast and broadcast on a middleware abstraction layer, suitable for underlay and overlay communication. We discuss its application in the context of hybrid multicast schemes.**

***Index Terms*—Key-based Routing, Hybrid Multicast, Dabek Model, Adaptive Protocol Stack**

I. Introduction

A structured overlay network consists of three functional groups: a routing scheme, a set of services and the applications. The routing, based on a decentralized key approach, is responsible for locating peers associated with specific key ranges. Such routing algorithm is independent of the applications built upon it. Services like group communication, failover redundancy, etc. supplement the structured overlay and can be developed independently of both, the underlying overlay routing and the application. A well designed protocol architecture should separately account for these components and offer pluggable, modular building blocks that jointly serve as a rich communication fundament.

Structured peer-to-peer systems offer multicast services in an infrastructure-agnostic fashion. They are reasonably efficient and scale over a wide range of group sizes. However, they do not allow for layer 2 interactions and thus do not facilitate unrestricted scaling in shared end system domains. Stability issues for tree-based overlay multicast under churn arise as well, as the departure of branching nodes close to the root may have disastrous effects on data distribution. These drawbacks may be mitigated by hybrid approaches, where overlay multicast routing only takes place among selected nodes, which are particularly stable and form a virtual infrastructure. Hybrid multicast schemes inherit major efficiency from the IP layer, while sustaining ease in deployment and infrastructure-transparency from selected group distribution in overlay networks.

The fundamental idea towards a layered architecture and a common API for structured overlays has been presented by Dabek *et al.*, proposing a key-based routing (KBR) interface to locate peers independently of the overlay protocol in use [1]. But its group service model has only been worked out to *join* and *leave* calls, and does not allow for hybrid multicast.

We give an architectural overview of our proposed group communication stack in section II, and present its application to merge native and overlay multicast (OLM) in section III.

II. A Group Communication Network Stack

The design goals of an application layer multicast service for structured overlay networks are twofold. On the one hand, the *architecture* for the OLM component itself needs to be defined along with its placement in the global system. On the other hand, a generic *API* for each of the interchangeable modules must be identified. In the following, we describe a generic architecture and its main building blocks. These components can also consist of sub-components, which depend on implementation details.

Architectural Overview:
Overlay multicast supplements nodes without a global multicast connectivity with a wide-area group communication service. Thus it is important to provide a transparent (virtual) network stack to application developers beyond the P2P context. This may include enhanced group communication services like group aggregation in namespaces. The group communication stack (see figure 1) consists of a middleware, underlay and overlay multicast modules. The middleware manages the data exchange between applications and group services. Depending on service availability, it creates a transparent overlay or a native network communication channel. In addition to common multicast applications interfaces, the middleware provides a service API to reflect group communication states.

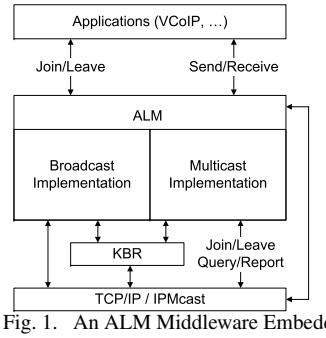

Fig. 1. An ALM Middleware Embedded in a P2P Stack

†The author is also with HAW Hamburg, Dept. Informatik.

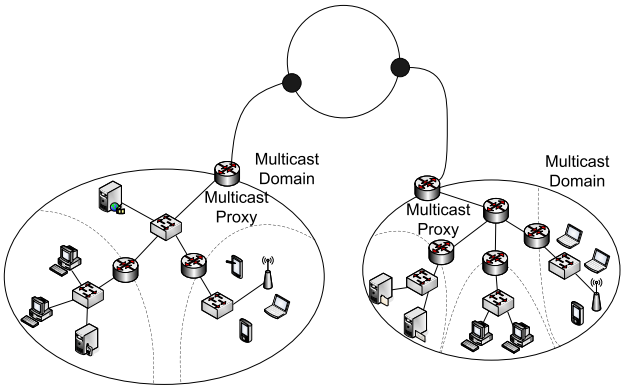

Fig. 2. Two Multicast Domains Covering Multiple Networks (Dashed Lines)

Overlay data will be handled by the broadcast or multicast implementation, depending on the destination address in use. Since broadcast will be delivered without explicit subscription, it is only the multicast implementation that internally provides join and leave calls. The OLM protocols operate by overlay unicast communication. For this reason they interact with the key-based routing layer via the common API [1]. The use of KBR is twofold: On the one hand, it may operate as a transmission layer delivering data to overlay peers. On the other hand, it provides group protocol implementations with unicast routing states. The latter is not needed for overlay multicast and broadcast transmission, as data can be sent natively to peers. For this purpose, the multicast and broadcast components require an interface to the IP layer, as well.

API: The proposed interface divides into two functions: (a) to send or receive multicast data, and (b) to monitor multicast service. The latter includes a *neighborSet* and *groupSet* call, returning all routing neighbors and maintained groups, respectively. Additional upcalls, *update*, inform the application about source and listener state changes.

III. Merging Native and Overlay Multicast

Native and overlay multicast can be interconnected by a specific Inter-domain Multicast Gateway (IMG) as architecturally proposed in [2]. IMGs transparently forward multicast data between overlay and underlay without tunneling. We present an implementation based on the group network stack next.

1) Connecting Small Size Domains: Small size multicast domains consist of one IP network with group management obtained from IP IGMP/MLD, optionally interconnected without a multicast routing protocol as specified in RFC 4605. In this case, the IMG operates in the IGMP/MLD router part. Based on the *groupSet* call, the IMG requests the MLD state table, which provides information about active listeners. In combination with the *update* call, the IMG then initiates join and leave calls towards the overlay for the first and last receiver.

2) Connecting Large Domains: Most larger networks have established a local, domain-wide host-group routing without global multicast connectivity. In such cases, an IMG should be integrated into the local routing infrastructure to interconnect larger native multicast islands (cf. figure 2).

A hybrid multicast gateway must be aware of all groups inside a multicast domain to initiate corresponding states in the overlay. Hence, an IMG requires an interface to the routing infrastructure, where subscriptions occur. In general, this depends on the multicast routing protocol deployed. In rendezvous point (RP) schemes like PIM-SM, all receiver subscriptions and source data will be registered at the RP. Flooding schemes like DVMRP, however, distribute the information across all neighboring routers.

In the following, we sketch methods to integrate the IMG in the two most interesting multicast routing architectures.

PIM-SM: The Protocol Independent Multicast Sparse Mode (PIM-SM) [3] establishes rendezvous points. These entities receive listener and source subscriptions of a domain. To be continuously updated, an IMG has to be co-located with a RP. Whenever PIM register messages are received, the PIM routing instance must signal a new multicast source to the stack. Subsequently, the IMG joins the group and a shared branch between the RP and the sources will be established, which PIM may shortcut to a source specific tree. Source traffic will be forwarded to the RP based on the IMG join, even if there are no further receivers in the native multicast domain. Designated routers of a PIM-domain send receiver subscriptions towards the PIM-SM RP. The reception of such messages invokes an update call at the IMG, which initiates a join towards the overlay routing protocol. Overlay multicast data arriving at the IMG will then transparently be forwarded in the underlay network and distributed through the RP instance.

BIDIR-PIM: Bidirectional PIM [4] is a variant of PIM-SM. In contrast to PIM-SM, the protocol pre-establishes bidirectional shared trees per group, spanning multicast sources and receivers. The rendezvous points are virtualized in BIDIR-PIM as an address to identify on-tree directions (up and down). However, routers with the best link towards the (virtualized) rendezvous point address are selected as designated forwarders for a link-local domain and represent the actual distribution tree. The IMG needs to be placed on the RP-link, where the rendezvous point address is located. As source data will always be transmitted to the rendezvous point address, the BIDIR-PIM instance of the IMG receives the data and can signal new senders towards the stack. The first receiver subscription for a new group within a BIDIR-PIM domain needs to be transmitted to the RP to establish the first branching point. Using the *update* invocation, an IMG will thereby be informed about group requests from its domain, which are then delegated to the overlay.

References

[1] F. Dabek, B. Y. Zhao, P. Druschel, J. Kubiatowicz, and I. Stoica, "Towards a Common API for Structured Peer-to-Peer Overlays," in *Proc. of 2nd IPTPS Workshop*, LNCS, vol. 2735. Springer–Verlag, 2003, pp. 33–44.

[2] M. Wählisch and T. C. Schmidt, "Between Underlay and Overlay: On Deployable, Efficient, Mobility-agnostic Group Communication Services," *Internet Research*, vol. 17, no. 5, pp. 519–534, 2007.

[3] B. Fenner, M. Handley, H. Holbrook, and I. Kouvelas, "Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)," IETF, RFC 4601, August 2006.

[4] M. Handley, I. Kouvelas, T. Speakman, and L. Vicisano, "Bidirectional Protocol Independent Multicast (BIDIR-PIM)," IETF, RFC 5015, 2007.