Supporting Mobile Streaming Services in Future Publish/Subscribe Networks

Konstantinos Katsaros, Nikolaos Fotiou, George C. Polyzos and George Xylomenos
Mobile Multimedia Laboratory
Department of Informatics
Athens University of Economics and Business
Athens, Greece
e-mail: [ntinos, fotiou, polyzos, xgeorge]@aueb.gr

***Abstract*— The architecture of the current Internet was not originally designed to support either mobility or multicast. In particular, its coupling of host identification and location identification has hindered the provision of effective mobile services. At the same time, its lack of support for multicast distribution causes a multitude of redundant unicast transmissions, leading to an inefficient utilization of network resources. Both these limitations are especially apparent in the case of real-time continuous media distribution. The publish/subscribe paradigm has been proposed as a promising alternative to the current send/receive paradigm for a future Internet architecture. In future publish/subscribe networks, multicast will be the norm, and this change of the end-to-end communication semantics will lead to a networking environment more suitable for mobility. In the context of this paradigm, this paper considers a prototype architecture based on the Scribe overlay multicast scheme. Preliminary simulation results show that our publish/subscribe network implementation achieves better performance during mobility compared to Mobile IPv6 in all relevant metrics, such as hand-off delay (or resume time) and loss of real-time traffic during disconnections, at the cost of a slight increase of the end-to-end delay due to the routing stretch imposed by the overlay.**

I. Introduction

In the early days of the Internet, mobility was not an issue, as hosts were (at best) the size of a large closet. As the Internet evolved and hosts shrunk, the need for mobility appeared, but instead of redesigning the Internet, mobility was supported via “add-on” protocol enhancements such as Mobile IP. Nowadays, in order to address a number of difficulties with the current Internet architecture and realize the Internet’s full potential, a clean slate approach is widely advocated [1]. The intensified traffic requirements, content evolution, prevalence of mobile devices, novel applications, real-time constraints and other requirements create an environment with which the current Internet architecture can barely cope. Such a clean slate approach should revolve around information, not endpoints [2], taking mobility into consideration from the start rather than as an afterthought.

The publish/subscribe communication paradigm has been suggested as a candidate for a future Internet architecture as it focuses on the information rather on the endpoints generating or consuming that information. The publish/subscribe paradigm has received considerable attention and many variants have been proposed, each adapted to different applications or slightly different network models [3]. A publish/subscribe architecture divides the roles of the endpoints in two categories: publishers and subscribers. Publishers are information generators or owners and subscribers are information consumers. Subscribers express their interest in particular pieces of information, in the form of subscription messages to an event notification service. Publishers make information available in the form of publications, which are also handled by the event notification service. This service is responsible for matching subscriptions with publications and notifying subscribers when a publication matches their interest.

In this paradigm, subscribers are not necessarily aware of the publishers, and vice versa, and there is no need for synchronization or coordination between them as the event notification, or rendez-vous, service handles this indirect form of communication. The anonymity and the asynchrony of publish/subscribe systems allows them to adapt quickly to frequent disconnections and reconnections, making them advantageous in a mobile network environment [4]. At the same time, by shifting towards an information-centric paradigm, multicast becomes the main routing mechanism for the efficient delivery of the content to the group comprised by an information item’s subscribers. Hence, multicast assisted mobility [5] re-emerges as a promising research direction, but in a new context.

The *Publish/Subscribe Internet Routing Paradigm* (PSIRP) [6] EU-funded research project is working on creating a clean slate Internet architecture based on the publish/subscribe communication paradigm. However, being pragmatic, PSIRP considers two levels of publish/subscribe implementations: a native one, where network elements directly support publish/subscribe at all stack layers, and an overlay approach, where the publish/subscribe functionality is provided on top of an underlying traditional (e.g., IP) network. In this paper we follow the overlay based approach towards a publish/subscribe Internet architecture, using Scribe [7], an overlay multicast scheme, as a starting point. We show the effectiveness of Scribe in supporting mobility in the context of a video streaming application. Based on this overlay architecture we study the intrinsic characteristics of a publish/subscribe network and explore the potential benefits of an overlay realization of this paradigm, which would undoubtedly be easier to deploy than its native counterpart. Our work is inspired from previous efforts on multicast assisted mobility [5], but it also exploits certain properties of the underlying Pastry *Distributed Hash*

Table (DHT) scheme [8]. To the best of our knowledge, this is the first attempt to address the issue of mobility in the context of overlay multicast.

The remainder of this paper is organized as follows. Section II briefly presents related work. Section III provides a detailed description of the proposed network architecture, which is then evaluated, via both analysis and simulation, against Mobile IPv6 in Section IV. Finally, Section V presents conclusions and directions for future work.

II. Related work

Mobility and its adverse implications for the current Internet architecture have been widely studied over the past years, resulting in a wide range of mitigating approaches. The Mobile IP (MIP) scheme, whose most recent version was designed for IPv6 [9], was one of the first attempts to add mobility to IP. In MIP whenever a *mobile node* (MN) connects to a foreign network, after being assigned a *care of address* (CoA), it informs its *home agent* (HA), that is, a router in its home network which represents the MN when it is away, about its CoA. Every message towards the MN has as the destination address the MN's initial home address, therefore it has to be tunneled by the HA to the CoA of the MN (see Fig. 1). This leads to an inefficient triangular routing and has triggered the emergence of various optimizations such as Route Optimization in Mobile IPv6. In this scheme, the *corresponding node* (CN), that is, the node currently communicating with the MN, is informed of the MN's CoA, so that traffic can be directly destined there.

Even with route optimization, MIP suffers from various problems. Packet encapsulation, which is required until route optimization starts working, adds significant overhead to each packet, and handover is not always fast enough. MIP handles local mobility of a host the same way it handles global mobility, so the same signaling load is incurred regardless of the user's mobility pattern, making MIP not scalable. In order to tackle this problem various extensions to MIP have been proposed such as Hierarchical MIPv6 [10], an extension to MIP that distinguishes between local and global mobility, and Cellular IP [11] which aims to provide fast local mobility. In order to perform faster handovers, various approaches, such as the Fast Handover for MIPv6 scheme [12], try to predict user movement so as to pre-configure the next access point. The accuracy of these prediction schemes is not very satisfactory [13] and sometimes the burden they impose at the link layer outweighs the improvement in handoff time that they offer.

The alternative approach of multicast assisted mobility has been widely studied in the context of IP multicast [5]. This approach creates multicast trees that distribute data around the area that a mobile user resides in (see Fig. 2). This scope of this distribution can be either statically determined, for example, data are forwarded to all neighboring areas, or based on the expectation of possible movements of the mobile node. Multicast based mobility solutions offer the advantage of fast local handoff as well, as they provide an effective mechanism for simultaneous data delivery to various destinations. However, these approaches still suffer from IP multicast's drawbacks, such as its limited scalability which have hindered its deployment. In this work, we avoid these limitations and revisit multicast assisted mobility by focusing on overlay multicast.

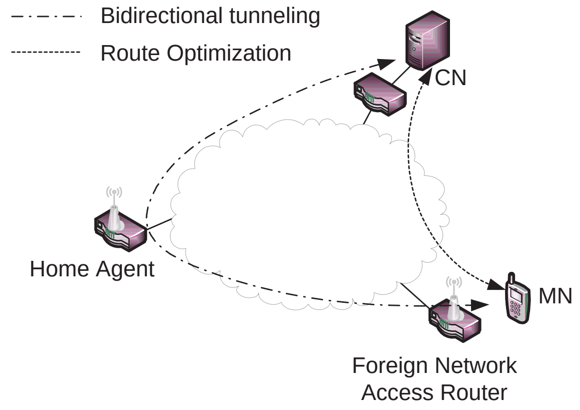

Fig. 1. Routing in Mobile IP.

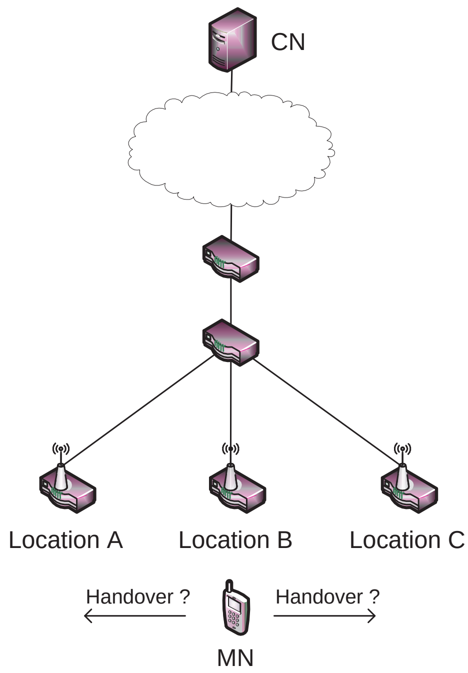

Fig. 2. Multicast assisted mobility.

III. A publish/subscribe Internet architecture

In the envisioned publish/subscribe architecture for the future Internet, the notions of end-host identity and location are decoupled. Endpoints are identified by flat, location independent identifiers. The event notification service is realized by *rendez-vous points*, that is, network elements where subscription and publication matching, as well as forwarding, is taking place. Routing and forwarding will take place as in the routing on flat labels [14] and data-oriented network architecture [15] proposals, where only data identifiers, called *rendezvous identifiers* (RId)[1], are used for routing and forwarding decisions. Each rendez-vous point is responsible for a certain range of RIds, performing the subscription and publication matching for those RIds, with all other rendezvous points providing forwarding services. Whenever a subscriber wants to express its interest for a certain publication, it only needs to know its RId, not who is the publisher or where this publication is located. In a similar manner, publishers do not have to know who is accessing their publications, they only have to set an appropriate RId for it and advertise it to the correct rendez-vous point.

This paradigm favors the use of multicast and caching. Whenever a rendez-vous point receives multiple subscriptions for the same RId, it forwards a single subscription to the next rendez-vous point towards the publisher. In effect, there will be a single data flow from the publisher towards that rendez-vous point. When data reaches a rendez-vous point that merged multiple subscription messages, it is copied and forwarded to all subscribers. Moreover, a rendez-vous point may cache published data for a period of time in order to serve future subscriptions.

A. An overlay approach

In this paper we consider an overlay realization of the proposed architecture, based on Scribe [7], an overlay scheme providing multicast routing. It is anticipated that the deployment of an overlay publish/subscribe architecture will make the transition to the new paradigm smoother, as it will be able to coexist with current technologies. Scribe is based on Pastry [8], an efficient and scalable DHT substrate. Unlike other DHT schemes Pastry employs proximity metrics, such as the number of IP hops or the round trip time towards other nodes, when choosing among the potentially large number of DHT nodes that may be used to relay data. As a result, the average distance a message travels does not exceed 2.2 times the distance between the source and the destination in the underlying IP network [16]. Scribe maps the name of each group to an identifier and makes the node responsible for that identifier the rendez-vous point for that group. A subscriber to a group issues a Scribe JOIN message towards the RV of the multicast group. As the JOIN message propagates via Pastry's overlay forwarding mechanism, reverse path routing state is established until a node already in the tree is found, thus forming a multicast tree rooted at the rendez-vous point. When the rendez-vous point receives a publication it forwards it to all nodes that have expressed their interest for this publication.

One of the main reasons for choosing this particular mechanism is that multicast routing state is maintained in a completely decentralized fashion: each node in a tree is only aware of its immediate ancestors and descendants, eliminating the signaling traffic needed in order to maintain global state information. Hence, in a highly dynamic environment with a multitude of mobile nodes, where group membership (i.e. subscriptions) change not only due to content-related reasons but also due to mobility, state management is simplified. Furthermore, according to the *route convergence* property of Pastry, the distance traveled by two messages sent to the same identifier before their routes converge is approximately equal to the distance between their respective source nodes. This property is of particular importance in mobile environments, as when a mobile node moves from one *access point* (AP) to another, these APs are expected to be close enough, therefore a new subscription message from the mobile node will meet its previous delivery tree in a few hops.

In our architecture every router participates in Pastry and can serve Scribe multicast trees as well. Every router is assigned a unique ID and hence, a unique position in the identifier space. On the other hand, (mobile) end nodes neither participate in Pastry nor carry an IP address. This clearly reflects our target of breaking the end-to-end semantics of today's communication. Every node is directly connected to an *overlay access router* (OAR), which provides access to the overlay network and the multicast communication substrate. Mobile nodes are connected to OARs through their currently associated AP. APs may act as simple bridges to the wired part of the network, may form groups connected to a single OAR so that link layer mobility (roaming) is provided in certain parts of the network, or even act as OARs themselves. For simplicity, we will consider the first option in the remainder of this paper.

Whenever a (mobile) node wishes to act as a publisher, it simply delivers its publication to its OAR, which is then responsible to deliver it to the proper rendez-vous point. Whenever a (mobile) node wishes to subscribe to a publication, it sends a subscribe(RId) message to its OAR. When an OAR receives a subscribe(RId) message for the first time, it issues a Scribe JOIN message towards the RV of the multicast group with id equal to RId requested, so as to become a member of that group and its multicast tree.

B. Mobility support

For mobility to operate properly in the above described publication delivery mechanism, we must take into account that as *mobile nodes* (MNs) move from one AP to another it is possible that they will change their OAR as well. In this case, they must inform their new OAR about the publications they are interesting in so that the OAR may join the appropriate trees. Of course, it is possible for an OAR to already belong to the corresponding multicast tree, either because another end node attached to that OAR had already subscribed to the same publication, or because this OAR had become a part of this multicast tree in order to serve another OAR as an intermediate, forwarding node. In these cases, the OAR only needs to forward the publication towards the MN.

In order for an OAR to leave a multicast group, it must send a LEAVE Scribe message after a period of time from the moment it anticipates the need to leave the multicast tree, that is, the OAR schedules a delayed transmission of a Scribe LEAVE message for a specific group when the last MN for

[1]Statistically unique identifiers, for instance, the result of a hash function over the data.

that group has disassociated from the APs attached to the OAR. The reason why an OAR does not immediately leave the multicast group is two-fold. First, whenever an OAR leaves a multicast group, the multicast tree for this publication is (partially) destroyed, thus it may not be possible for a MN to take advantage of the existing Scribe trees and the route convergence property of Pastry, unless if another MN in the same area has subscribed to the same tree. Second, a MN may soon return to the original OAR[2], so by refraining from immediately leaving the multicast tree, signaling is reduced and potential caching/buffering techniques may be enabled.

This architecture has significant advantages in supporting mobility. First, by employing multicast as the main routing mechanism, routing updates can be localized. This means that when a MN moves from an OAR to another, traffic can be diverted to its new point of attachment at the lowest *common ancestor* (CA) of the two OARs in the tree. Second, neighboring OARs can proactively join the established multicast tree in order to prepare for a potential attachment of the MN. This proactive mechanism could be based either on a deterministic selection of nearby OARs or a probabilistic analysis of a MN's movement (e.g. [13]). This enhancement lies beyond the scope of this paper, but constitutes and important area for future work. Third, due to the route convergence property of Pastry, this CA is expected to be close enough to the new point of attachment for the routing update to be performed fast. The intuition is that when a MN moves from one AP to another, these APs are expected to be close enough in the proximity space, so the re-subscribe message of the MN will meet the CA after traveling a short distance in the proximity space. On the other hand, the *stretch*[3] imposed by overlay routing is expected to negatively affect the performance of the proposed architecture.

IV. Performance evaluation

In this section we evaluate the performance of the proposed architecture in the context of streaming services such as Mobile TV. Our target is to investigate the impact of node movement on the experienced service quality in the context of the proposed mobility support mechanism. We compare the performance of this mechanism to that of Mobile IPv6, a standardized solution that can also serve as a basis for indirect comparisons with other mobility schemes. In this comparison the Route Optimization feature of MIPv6 is enabled through the standardized Return Routability procedure [9]. We did not consider multicast assisted mobility schemes based on IP multicast due its limited deployment. The following performance metrics are investigated:

Resume time: The elapsed time between the association of the MN with a new AP and the reception of the first packet by the MN in its new position. It indicates the efficiency of the signaling mechanism used to divert traffic towards the new position of a MN.

Packet loss: The percentage of transmitted packets not delivered to the MN due to mobility. It is a coarse grained indication of the service disruption incurred.

End-to-end packet delay: The elapsed time between the transmission of a packet by its source and the reception of the packet by the MN. It indicates the impact of stretch on the delivered service.

A. Signaling analysis

In this section we provide an analytical comparison of the proposed architecture and Mobile IPv6 in terms of signaling overhead. We chose Resume time as the primary performance metric as it directly reflects the efficiency of a mobility support scheme in terms of communication disruption. Obviously, this time is heavily affected by the signaling required in order for the network entities involved to be informed about the MN's change of position, that is, the time required for the routing substrate to adapt to the movement of the MN. Figure 3 shows a generic network topology in which a *mobile node* (MN), initially residing at its home network, moves around the network while communicating with a *corresponding node* (CN). Table I outlines the signaling messages of the Return Routability procedure which is required for Route Optimization to work[4].

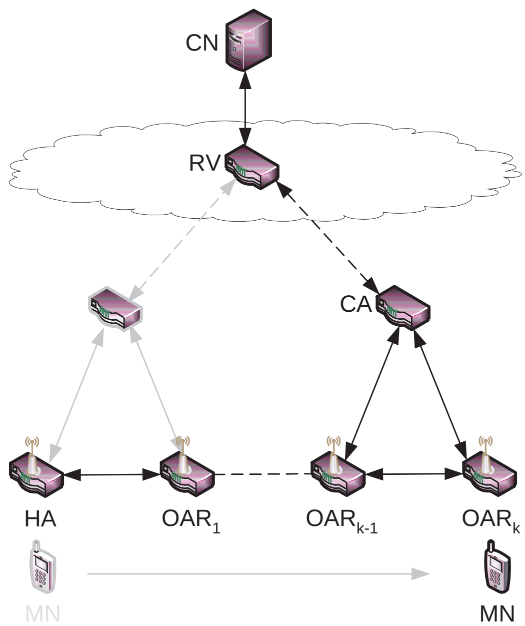

Fig. 3. Example topology.

In the following we use $d_{x \to y}$ to denote the distance in number of hops between the network entities x and y. Since steps 3 and 4 and steps 5 and 6 of the Return Routability procedure take place in parallel, we can express the Resume

[2]This could also be due to a *ping-pong* effect between neighboring APs.

[3]The ratio between an overlay path length (or delay) and the length (or delay) achieved by IP routing.

[4]A limited description is provided here due to length limitations. For further details please refer to [9].

#	**Message Type**	**Source**	**Destination**
1	Binding Update (BU)	MN	HA
2	Binding Ack. (BA)	HA	MN
3	Home Test init (HoTi)	MN	HA → CN
4	Care-of-Test init (CoTi)	MN	CN
5	Home Test (HT)	CN	HA → MN
6	Care-of Test (CT)	CN	MN
7	Binding Update (BU)	CN	MN
8	Binding Ack. (BA)	MN	CN

TABLE I

RETURN ROUTABILITY PROCEDURE.

Time of Mobile IPv6 as follows:

$$RT_{MIPv6} = 4d_{MN\rightarrow HA} + 2d_{MN\rightarrow CN} + 2d_{HA\rightarrow CN} \quad (1)$$

In our *overlay multicast assisted mobility* (OMAM) scheme, a re-subscribe message is sent over the wireless medium by the MN towards the newly visited OAR which in turn generates a Scribe JOIN message towards the lowest *common ancestor* (CA). Hence:

$$RT_{OMAM} = d_{MN\rightarrow OAR_k} + d_{OAR_k\rightarrow CA} \quad (2)$$

The following equation expresses Pastry's route convergence property.

$$d_{OAR_k\rightarrow CA} = a \times d_{OAR_{k-1}\rightarrow OAR_k}, a \rightarrow 1 \quad (3)$$

By neglecting the distance cost of wireless medium transmissions in both scenarios and assuming, for simplicity, that OAR_{k-1} belongs to the path between the HA and OAR_k, our scheme results in a smaller RT value when:

$$RT_{OMAM} < RT_{MIPv6} \Leftrightarrow$$

$$a < 4 + 2\frac{2d_{OAR_{k-1}\rightarrow HA} + d_{OAR_k\rightarrow CN} + d_{HA\rightarrow CN}}{d_{OAR_{k-1}\rightarrow OAR_k}} \quad (4)$$

According to the route convergence property $a \rightarrow 1$, thus it is evident that our architecture results in a reduced RT. Moreover, $RT \rightarrow 0$ when OAR_k is already a member of the publication's multicast tree. This may happen due to another MN attached to OAR_k that has expressed interest for the same publication, or because OAR_k is acting as a forwarding node, that is, an intermediate tree node, for another OAR_j.

B. Simulation results

To evaluate the performance of our scheme under dynamic conditions, we used the OMNeT++ simulation framework [17] enhanced with xMIPv6 [18] and OverSim [19]. In our simulations we considered a simple network topology comprising multiple OARs deployed in a grid-like topology. All OARs run the full TCP/IP protocol stack, as well as Pastry and Scribe in the case of the proposed architecture. All wired connections are implemented with Ethernet links. One IEEE802.11b AP is directly connected to each OAR, with neighboring APs operating in disjoint channels. We were restricted by the xMIPv6 model structure which necessitates the deployment of a single home network with a single MN in any topology.

Both the HA and the CN are connected to randomly chosen, disjoint OARs of the topology. The AP in the Home Network of a MN is directly connected to the HA rather than to the respective OAR. In the case of the proposed architecture and in order to produce comparable scenarios, we placed an extra OAR between the AP and the randomly chosen OAR acting as the initial point of attachment of the MN to the network. Furthermore, an extra OAR was also connected to a randomly chosen OAR in the grid topology to serve as the originator of the data flow. Figure 4 shows an example topology. The circles around each networking entity denote the transmission range of its wireless interface. The square bounded area denotes the part of the topology actually accessible by the MN. We have chosen a topology providing full wireless coverage in order to restrict the anticipated service disruption to each protocols' operation.

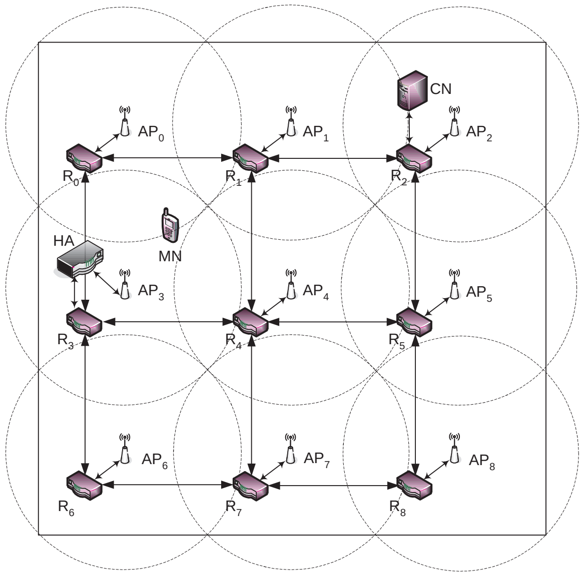

Fig. 4. Grid-like example topology.

Our scenarios consider a single MN initially connected to its home network. Upon initialization, the MN starts to move following the mobility model described in [20]. In this model a MN moves in a straight line, makes a turn and starts over. Its speed and direction are updated every x seconds, with x following a normal distribution with a mean of 10 seconds and a standard deviation of 0.1 seconds. The speed of the MN is normally distributed with a mean of 1.39 meters/sec (approximate walking speed of 5 Km/h) and a standard deviation of 0.01 meters/second. The change in direction also follows a normal distribution with an average of 0 degrees (no turn) and a standard deviation of 5 degrees. Whenever the MN reaches the limits of the simulated area it bounces with the same angle and speed.

Our measurements were based on a simple application scenario in which a stationary node sends a sequence of UDP datagrams (CBR traffic) towards the MN. The UDP stream resembles a H.264, Level 1 SQCIF video stream with 30.9 frames per second [21]. In the case of MIPv6, the CN is initially only aware of the MN's home address and uses it as the destination of its data. While moving, the MN is responsible for updating its bindings with its HA and the CN in the case of Route Optimization. In the case of the proposed architecture, the CN simply sends its UDP packets towards a rendez-vous point whose position in the networks is determined by a randomly generated key and Pastry's functionality. The complete set of parameter values used in our simulation environment are provided in Table II.

Parameter	**Value**
Grid size	30 x 30
Number of MNs	1
Number of CNs	1
Wired connections type	100 Mps Ethernet
Propagation delay (ms)	0.5
Data rate (Kbps)	64
Packet size (bytes)	26
Total number of packets sent	556200

TABLE II
SIMULATION SCENARIO PARAMETERS.

We now present our initial results derived from the simulation scenarios described above. These results constitute the first step towards a thorough evaluation of the proposed architecture with respect to mobility. Table III shows the values derived for the above described metrics.

	Mobile IPv6	**Overlay Multicast**
Resume time	1.208 sec	0.007 sec
Packet loss	2.002%	1.059%
End-to-end delay	12ms	17ms

TABLE III
SIMULATION RESULTS.

As expected, our scheme yields a significantly lower Resume time than Mobile IPv6. This is justified by the localized handling of mobility. Upon each change of network position Mobile IPv6 launches the Return Routability procedure in order to apply Route Optimization. As shown in the previous section, this causes an exchange of signaling packets both with the HA and the CN. Considering the fact that these nodes may be located in distant parts of the network, this signaling may considerably delay the establishment of new routes for the traffic destined to the MN. On the other hand, in our architecture this signaling overhead is reduced since only the CA node in the multicast tree is notified about the MN's change of position. The significance of the reduced Resume time is depicted in the Packet loss metric where we see a noticeable difference between the two considered approaches. Furthermore, it must be noted that by localizing routing updates our architecture enables the actual delivery of packets in transit in the scenarios where the CN resides in a distant area of the network. However, these improvements come at the cost of end-to-end delay. Indeed, as we see in Table III, our approach results in a higher end-to-end delay. As explained earlier, this is due to stretch imposed on the routing due to the reliance on a DHT substrate. We must note however, that this increase could be acceptable for non-interactive streaming applications as the ones considered in this paper.

V. CONCLUSIONS AND FUTURE WORK

In this paper we show how an overlay multicast architecture can assist mobility, especially in the context of multimedia streaming applications. Our reference model is the information centric publish/subscribe paradigm, a promising approach to address many of the deficiencies of the current Internet's architecture and protocols. Initial measurements show that overlay multicast routing can greatly assist mobility in terms of reduced hand-off delay (or Resume time) and the reduced data loss if buffering and re-forwarding of traffic is not implemented, which would be natural for real-time streaming applications, at the cost of slightly increased end-to-end delay due to the overlay imposed path stretch.

Future work will explore several different promising directions. First, caching, which we have not yet considered but is natural for publish/subscribe, is expected to accelerate data dissemination, improve efficiency and better support, in particular, reliable communication. Second, mobility prediction schemes in conjunction with delayed LEAVE messages and proactive multicast group joins must be considered as they can certainly lead to faster handoffs. While mobility prediction is the first step, also taking advantage of the local nature of mobility, and the multicast nature of all communication in a publish/subscribe network, mobility prediction can become very effective, particularly since information can be economically distributed to a few neighboring locations. Finally we need to investigate how mobility support will be affected when multicast will have to cope with reliable data transfer, as well as two-way, or multipoint-to-multipoint, communication.

ACKNOWLEDGMENTS

This work was supported by the EU FP7/ICT PSIRP project under contract ICT-2007-216173.

REFERENCES

[1] A. Feldmann, "Internet clean-slate design: what and why?" *SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 3, pp. 59–64, 2007.
[2] V. Jacobson, *If a Clean Slate is the solution, what was the problem?*, http://cleanslate.stanford.edu/seminars/jacobson.pdf.
[3] P. T. Eugster, P. A. Felber, R. Guerraoui, and A.-M. Kermarrec, "The many faces of publish/subscribe," *ACM Comput. Surv.*, vol. 35, no. 2, pp. 114–131, 2003.
[4] Y. Huang and H. Garcia-Molina, "Publish/subscribe in a mobile environment," *Wirel. Netw.*, vol. 10, no. 6, pp. 643–652, 2004.
[5] A. Festag, H. Karl, and A. Wolisz, "Investigation of multicast-based mobility support in all-IP cellular networks: Research articles," *Wirel. Commun. Mob. Comput.*, vol. 7, no. 3, pp. 319–339, 2007.
[6] PSIRP Project Team, "PSIRP project home page," http://www.psirp.org.
[7] M. Castro, P. Druschel, A.-M. Kermarrec, and A. Rowstron, "SCRIBE: A large-scale and decentralized application-level multicast infrastructure," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 8, pp. 100–110, 2002.
[8] A. Rowstron and P. Druschel, "Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems," in *In Proc. of the 18th IFIP/ACM International Conference on Distributed Systems Platforms (Middleware)*, ser. Lecture Notes In Computer Science, vol. 2218. Springer-Verlag, November 2001, pp. 329–350. [Online]. Available: *http://www.research.microsoft.com/˜antr/PAST/pastry.pdf*

[9] D. Johnson, C. Perkins, and J. Arkko, "Mobility support in IPv6," RFC 3775 (Proposed Standard), June 2004. [Online]. Available: *http://www.ietf.org/rfc/rfc3775.txt*

[10] C. Castelluccia, "HMIPv6: A hierarchical mobile IPv6 proposal," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 4, no. 1, pp. 48–59, 2000.

[11] A. T. Campbell, J. Gomez, S. Kim, A. G. Valko, C.-Y. Wan, and Z. R. Turanyi, "Design, implementation, and evaluation of cellular IP," *IEEE Personal Communications*, vol. 7, pp. 42–49, 2000.

[12] R. Koodli, "Fast Handovers for Mobile IPv6," RFC 4068 (Experimental), July 2005, obsoleted by RFC 5268. [Online]. Available: *http://www.ietf.org/rfc/rfc4068.txt*

[13] L. Song, D. Kotz, R. Jain, and X. He, "Evaluating next-cell predictors with extensive Wi-Fi mobility data," *IEEE Transactions on Mobile Computing*, vol. 5, no. 12, pp. 1633–1649, 2006.

[14] M. Caesar, T. Condie, J. Kannan, K. Lakshminarayanan, and I. Stoica, "ROFL: routing on flat labels," *SIGCOMM Comput. Commun. Rev.*, vol. 36, no. 4, pp. 363–374, 2006.

[15] T. Koponen, M. Chawla, B.-G. Chun, A. Ermolinskiy, K. H. Kim, S. Shenker, and I. Stoica, "A data-oriented (and beyond) network architecture," *SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 4, pp. 181–192, 2007.

[16] M. Castro, P. Druschel, Y. Charlie, and H. A. Rowstron, "Exploiting network proximity in peer-to-peer overlay networks," Microsoft Research, Tech. Rep., 2002. [Online]. Available: *http://research.microsoft.com/users/Cambridge/antr/OLD/PAST/location.pdf*

[17] A. Varga, "OMNeT++ network simulator home page," http://www.omnetpp.org.

[18] F. Z. Yousaf, C. Bauer, and C. Wietfeld, "An accurate and extensible mobile IPv6 (xMIPV6) simulation model for OMNeT++," in *Proc. of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops (Simutools '08)*. Brussels, Belgium: ICST, 2008, pp. 1–8.

[19] I. Baumgart, B. Heep, and S. Krause, "OverSim: A flexible overlay network simulation framework," in *Proc. of the IEEE Global Internet Symposium*, 2007, pp. 79–84.

[20] C. Perkins and K.-Y. Wang, "Optimized smooth handoffs in mobile IP," *Computers and Communications, 1999. Proceedings. IEEE International Symposium on*, pp. 340–346, 1999.

[21] ITU, "H.264 : Advanced video coding for generic audiovisual services," http://www.itu.int/rec/T-REC-H.264, 2007.