Project Number:	**IST-2001-32603**
Project Title:	**6NET**
CEC Deliverable Number:	**32603/ULP/DS/4.1.5v2/A1**
Contractual Date of Delivery to the CEC:	December 31st 2004
Actual Date of Delivery to the CEC:	January 07th 2005
Title of Deliverable:	Multicast with mobile hosts: analysis and performance evaluation
Work package contributing to Deliverable:	WP4
Type of Deliverable*:	R
Deliverable Security Class**:	PU
Editors:	Christophe Jelger (ULP), Thomas Noel (ULP)
Contributors:	Christophe Jelger (ULP), Thomas Noel (ULP)
Reviewers:	Martin Dunmore (Lancaster University)

* Type: P - Prototype, R - Report, D - Demonstrator, O - Other

** Security Class: PU- Public, PP – Restricted to other programme participants (including the Commission), RE – Restricted to a group defined by the consortium (including the Commission), CO – Confidential, only for members of the consortium (including the Commission)

Abstract:

This document aims to provide an analysis of the issues relative to IPv6 multicasting in the presence of IPv6 mobile nodes. This document is an extension of the version 1 of deliverable 4.1.5. This second version mainly focuses on the development and the deployment of a solution that allows a mobile node to receive and send multicast data via its Mobile IPv6 home agent. This solution, developed by ULP, is known as an MLD proxy. Such a service is publicly available via the 6NET network: an MLD proxy is indeed hosted at ULP and is publicly accessible.

Keywords:

Multicast, Mobile IPv6, MLD proxy.

Executive Summary

This document describes a solution that allows a mobile node to receive and send multicast data via its Mobile IPv6 home agent. The motivations for providing multicasting to mobile devices are as follows. Mobile communications are becoming extremely popular and efficient and, in the mean time, multicasting has a lot to offer, with services such as Internet TV and radio, network games, and video-conferencing applications. It is clear that mobile users want to have access to such kind of multicast services. For example, a user who is away from its home network may want to be able to organize a video-conference with some of its colleagues in order to discuss the terms and details of an important contract. If moreover the user is mobile (e.g. sitting in a taxi and using some wireless technology to access the Internet), delivering multicast data to the user is challenging.

The authors want to remind that the issues related to multicast communications in the presence of mobile group members have already been analysed in version 1 of this deliverable [1]. In this document, which is an extension of the version 1 of deliverable 4.1.5, we focus on a solution developed by ULP in order to add multicast forwarding capabilities to a Mobile IPv6 home agent. This solution is known as an MLD proxy. The MLD protocol (Multicast Listener Discovery) [5] is used by IPv6 routers to detect the presence of hosts that wish to subscribe to a given IPv6 multicast group. When the MLD proxy is installed on a Mobile IPv6 home agent, it allows a mobile node to subscribe and unsubscribe to multicast groups, and to send and receive multicast data via the home agent. This solution is currently the only available solution to provide multicasting to mobile nodes using the Mobile IPv6 protocol. ULP has also installed a MLD proxy that is publicly available via the 6NET network. With this public service, any mobile node using Mobile IPv6 can join multicast groups via the MLD proxy hosted by ULP.

The authors would also like to mention that the original goal of this deliverable was to report on the performance of multicast routing with mobile nodes and the Mobile IPv6 protocol. However, in 2001 we expected that multicast routing would be supported in the future releases of the implementations of Mobile IPv6. Unfortunately, this is still not the case at the end of year 2004. The specifications of the Mobile IPv6 protocol [2] indeed specify that multicast support SHOULD be implemented in home agents, and MAY be supported by mobile nodes. It also appears that the main objective of developers of Mobile IPv6 implementations (i.e. MIPL and KAME) has recently been focused on rewriting most of these implementations as user-space applications, with as little kernel code as possible. This has been done in order to minimize the complex task of maintaining kernel-dependent code.

As a result, little effort has been made to support multicast routing in current implementations of Mobile IPv6. The main consequence is that the original goals of this deliverable, i.e. compare and analyse the performance of multicast routing with Mobile IPv6, cannot be fulfilled. As a result, ULP has decided to develop a solution in order to add multicast support in current Mobile IPv6 implementations. This deliverable therefore focuses on this development since the original objectives are unfortunately obsolete.

Table of Contents

1 Introduction and problem statement

IP multicasting has been extensively studied in the past twenty years. In contrast and for various reasons not detailed here, there have been very few native deployments of multicast in ISP networks. However, the increasing availability of broadband access technologies like ADSL, along with the transformation of personal computers into interactive multimedia desktops (for digital video, music, Internet TV and radio, network games …), are factors which call for a deployment of native multicast support in the next generation Internet. The transition/migration to IP version 6 can also benefit from the immediate adoption of native multicast support. In particular, services like Internet TV and radio are perfect examples of multicast-like applications, where one source broadcasts the same content to a (possibly) very large set of receivers.

Meanwhile, there has also been an increasing interest for wireless communications, especially since powerful handheld devices are now widely available. It is also becoming clear that mobile Internet users will expect to have access to the services and applications that are available in traditional wired networks and these services will surely include multimedia applications. Consequently, many efforts are being made to provide efficient mobility and multicasting support and to bring the two together in the next generation of IP networks.

As already described in version 1 of this deliverable [1], multicast routing protocols in their current form do not provide any specific support for mobile nodes. Moreover, the current implementations of the Mobile IPv6 (MIP6) protocol do not support the tunnelling of multicast data between a mobile node and its home agent. This feature is however specified in the specifications of the Mobile IPv6 protocol [2]. The main consequence is that a mobile node using the Mobile IPv6 protocol cannot join multicast communications via its home agent.

In order to overcome this situation, ULP has developed an MLD proxy for Mobile IPv6. Such a proxy allows a mobile node using MIP6 to send and receive multicast data via its home agent. This development fulfils the requirements imposed by the MIP6 specifications in order to support the bi-directional tunnelling of multicast data between a mobile node and its home agent. The idea of using an MLD proxy with Mobile IPv6 was already proposed in [3]. A similar architecture [4] is also currently being considered at the IETF within the MAGMA working group. The goal is to achieve the delivery of multicast data without requiring the installation and management of true multicast routing protocols.

2 Summary of MIP6 specifications related to multicasting

The IETF MIP6 (Mobility for IPv6) working group has proposed to use two straightforward solutions which can allow a mobile node to receive and send data to one or more multicast groups. These solutions, are usually known in the literature as *remote subscription* and *bi-directional tunnelling*. Actually, these mechanisms do not depend on any particular version of the IP protocol. The use of these two solutions is explicitly mentioned in the specifications of the Mobile IPv6 protocol [2]. These two solutions are illustrated in Figure 1 (HA = Home Agent, MN = Mobile Network).

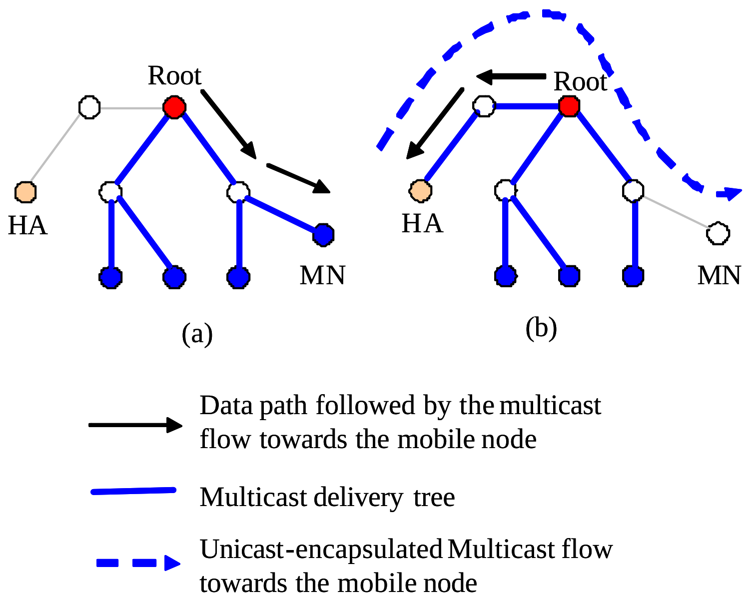


Figure 1. Remote subscription (a) vs. bi-directional tunnelling (b)

2.1 Remote subscription

With the solution known as remote subscription, a mobile node joins (or leaves) a multicast group via a (local) multicast router located on the foreign link being visited. Thus, the mobile node follows the traditional behaviour of any stationary node that wishes to join (or leave) a multicast group. In particular, the mobile node must use its (visited) link-local address when sending MLD messages (to the "all routers" multicast address). Although the Mobile IPv6 specifications [2] state that the care-of address MUST be used, the MLDv2 protocol [5] requires that a link-local address MUST be used. The former document indeed refers to MLD version 1. We personally favour the use of a link-local address as MLD is a protocol whose scope is restricted to a link. In contrast, the mobile node must use its care-of address if it wants to send data to the group (if the mobile node is also a source) in order to avoid possible problems with ingress filtering. All packets/datagrams are sent directly on the foreign link. The main advantage of this method is that routing is optimal, in the sense that the multicast data follows a shortest path route between the mobile node and the root of the multicast delivery tree. The main drawback of this solution is that it can introduce frequent tree reconstructions when the mobile node is in movement and hand-offs between different subnets. Moreover, if source-rooted multicast is considered, a major problem occurs if the source is mobile (see [1] for details).

This proposal is illustrated in Figure 1(a): multicast data is sent from the root of the tree to the visited network that hosts the mobile node.

2.2 Bi-directional tunnelling

Alternatively, a mobile node can join (or leave) multicast groups via a bi-directional tunnel to its home agent: this method is known as bi-directional tunnelling. The mobile node tunnels all its MLD messages to the home agent, which will then forward data to the mobile node via the tunnel. If the mobile node wants to send data to the group it must also use the tunnel. To do so, the source address of the (inner) multicast datagrams sent by the mobile node must be the home address of the mobile node (in order to avoid possible problems with ingress filtering by a router in the home network). In contrast, the source address of the (outer) unicast packet must be the care-of address of the mobile node, i.e. in order to avoid ingress filtering from a router of the visited network.

In order to support bi-directional tunnelling, the home agent must either have the functionalities of a multicast router (routing and forwarding), or have the functionalities of an MLD proxy. The main advantage of this solution is that the mobility of the mobile node does not have any influence on the multicast tree(s), since the mobility is hidden. Another advantage is that multicast only needs to be supported at the home network. The main drawback of this proposal is that routing becomes sub-optimal because the solution involves triangular routing via the home agent.

Bi-directional tunnelling is illustrated in Figure 1(b): multicast data is forwarded from the root of the tree to the home agent (HA) which forwards it (encapsulated) to the current position of the mobile node.

2.3 Current multicast support in MIP6 implementations

In their current form, none of the Mobile IPv6 implementation supports the bi-directional tunnelling of multicast data between a mobile node and its home agent. We have in particular checked the following popular implementations of Mobile IPv6 :

- MIPL (for the Linux operating system)
- KAME (for BSD-like operating systems)
- Mobile IPv6 home agent functionality in Cisco routers

We have therefore decided to develop a solution that extends the capabilities of the Mobile IPv6 protocol with multicasting support. This additional functionality, which implements bi-directional tunnelling as presented earlier, is known as an MLD proxy [3][4]. In particular, it follows the specifications of the Mobile IPv6 protocol related to multicasting [2].

3 MLD proxy

3.1 Introduction and specifications

In the particular case of bi-directional tunnelling, the home agent must have the functionalities of a multicast router. These functionalities can be separated in three fundamental modules: receivers subscription, multicast tree grafting/pruning, and data forwarding. Receivers subscription is the ability to receive MLD messages, i.e. the ability for the home agent to learn that a receiver wants to subscribe (or unsubscribe) to a given multicast group (or channel for SSM). Multicast tree grafting/pruning is the capability to join (or leave) the multicast delivery tree, i.e. the core operation of multicast routing. Finally, data forwarding is the aptitude to forward multicast data to interested receivers. For example, the PIM-SM multicast routing daemon available on the FreeBSD operating system complies with all these requirements. For IP version 6, this is specifically possible because the kernel of this operating system is capable of forwarding IPv6 multicast datagrams. In other words, FreeBSD implements an IP version 6 multicast forwarding cache (MFC). On the other hand, the current version of the Linux operating system does not support an IPv6 MFC.

In a strict sense, a home agent should be able to satisfy all of the three functionalities of a multicast router. However in practice, and especially in the particular case of bi-directional tunnelling, it is possible to separate these functionalities. The home agent does not indeed strictly require to perform the multicast routing operation. Removing this functionality from the home agent is of high interest for two reasons. First, the home agent can focus on its main task: forwarding (unicast and multicast) data to the current locations of the mobile nodes it serves. The second reason is mainly a deployment issue: in the first phase of the deployment of Mobile IPv6 services, network operators may be reluctant to integrate the multicast routing operation in their home agents. The main motivation for that would be the ease of administration.

This possibility leaves the home agent with two tasks: receivers subscription and data forwarding. In the literature, these functionalities are known as *MLD proxying* [3][4]. An MLD proxy must of course be associated with a router capable of multicast routing, but the MLD proxy itself is simply seen as a multicast receiver by the multicast router. In short, the MLD proxy subscribes to multicast groups on behalf of the mobile nodes it serves. To do so, it simply sends MLD messages on the link of the network to which it belongs. The multicast router of the link subsequently joins the corresponding multicast trees and forwards the data on the network where the MLD proxy resides. The final task of the MLD proxy is then to forward the multicast flows to the mobile nodes. To do so, it sends the multicast datagrams in the tunnels towards the mobile nodes. A mobile node can also wish to send datagrams to a given group. To do so, it uses its bi-directional tunnel to send multicast datagrams to the home agent, which removes the unicast header and forwards the multicast datagrams on the home network. It is then the task of the multicast router to forward this multicast flow onto the multicast tree. The operation of an MDL proxy is illustrated by Figure 2.

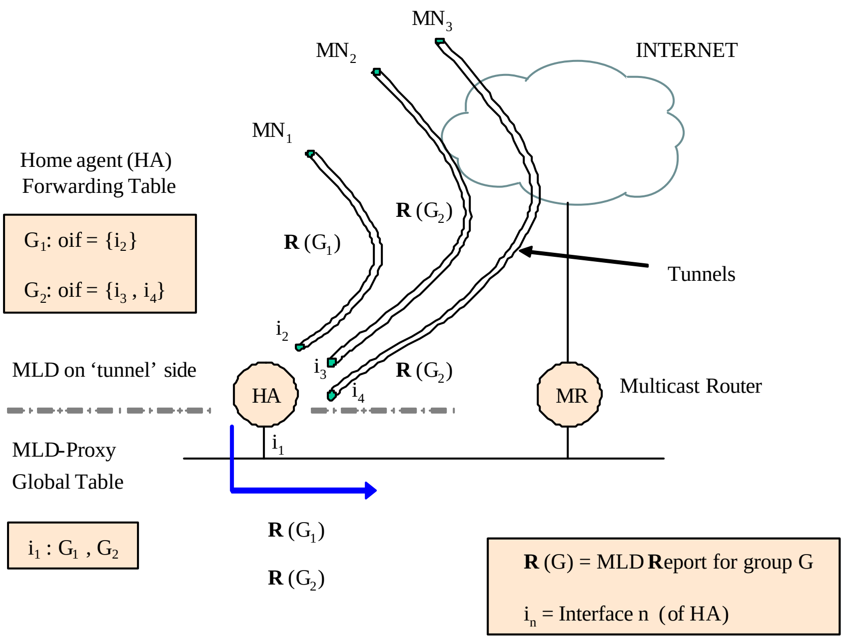


Figure 2. MLD proxy operation

In Figure 2, the home agent is also an MLD proxy. Three mobile nodes have registered with the home agent. One mobile node (MN_1) has sent an *MLD report* message for the multicast group G_1. The two other mobile nodes (MN_2 and MN_3) have sent *MLD report* messages for the multicast group G_2. The home agent, when receiving the reports, subscribes to the corresponding groups. To do so, it sends its own *MLD report* messages onto the network link. The multicast router subsequently joins the two groups, and starts forwarding the multicast flows onto the network. When the home agent receives a datagram destined for a multicast group for which at least one of the mobile it serves is subscribed, it forwards the multicast datagram to all the mobile nodes which are subscribed to this group. To do so, a copy of the datagram is sent to each interested mobile node via the corresponding Mobile IPv6 tunnel. The MLD proxy must therefore maintain a list of multicast groups to which it is subscribed, and a forwarding table that keeps a record of the outgoing interfaces (oif on Figure 2) to which datagrams must be forwarded. Moreover, and in order to be compatible with the multicast model, the home agent must forward multicast datagrams sent by the mobile nodes. It is the task of the multicast router to forward (or not) these datagrams onto the appropriate multicast trees.

The major advantage of using an MLD proxy with Mobile IPv6 is that the movements of the mobile nodes do not affect the MLD proxy in any way. When a mobile node moves and get attached to a new subnetwork, the tunnel end-point is updated by the Mobile IPv6 protocol and therefore the transmission and reception of multicast data via the tunnel can be resumed very quickly. From the point of view of the MLD proxy, nothing has changed. This is a very attractive feature.

3.2 Implementation issues

In the context of the 6NET project, ULP had implemented a first version of an MLD proxy at the time we wrote the version 1 of this deliverable. This first version was mainly a prototype designed to validate the operation and the feasibility of MLD proxying. This first version was developed on the Linux operating system, with the MIPL 1.0 implementation of the Mobile IPv6 protocol.

Since then, we have completely rewritten the code of the original prototype of the MLD proxy. This new version has also been developed for MIPL (see http://www.mobile-ipv6.org) and the Linux operating system. It works with both MIPL 1.0 and MIPL 1.1, but it is preferable to use MIPL 1.1 since it requires a Linux kernel (i.e. 2.4.26) that has a more stable implementation of the MLD protocol. Moreover, as the MLD proxy has been developed as a user-space application, it should also work with the recent release of MIPL 2.0. Due to time and man-power constraints, we unfortunately could not yet test our MLD proxy with MIPL 2.0. The current version of the MLD proxy is publicly available for download at ULP at the following URL :

http://www-r2.u-strasbg.fr/~jelger/MLD/MLD_Proxy_Main.htm

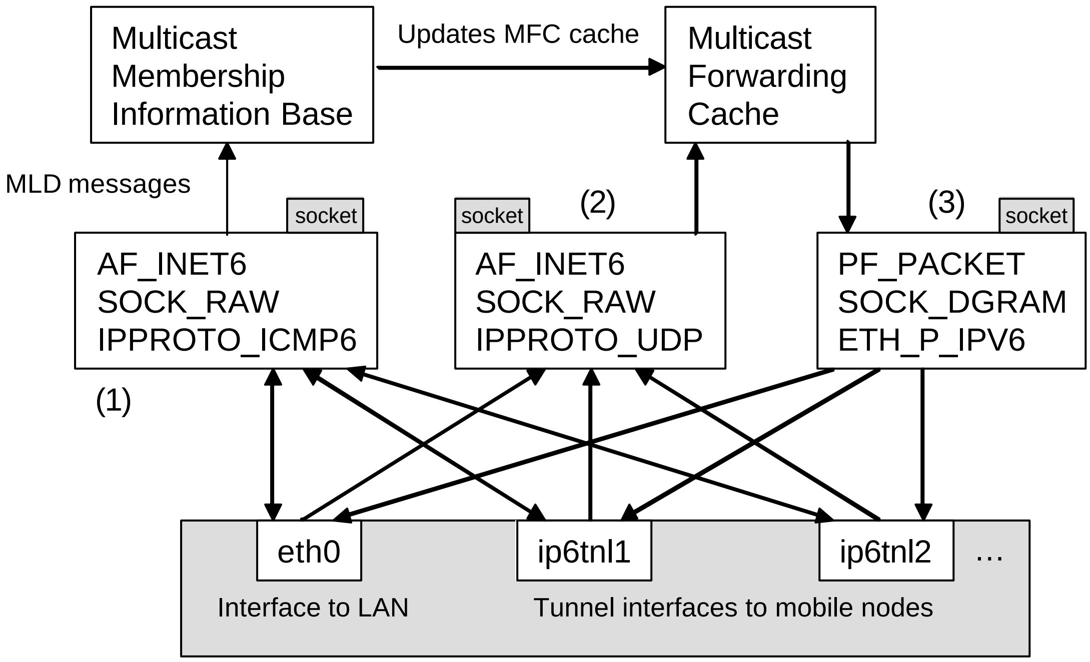
The particularity of this MLD proxy is that it forwards datagrams in *user land*, i.e. the MLD proxy itself is responsible for forwarding the multicast datagrams. The main reason for that is that the current version of the Linux operating system does not implement an IPv6 multicast forwarding cache (MFC). We also planed to do a porting on FreeBSD, but with the current state of the KAME implementation of Mobile IPv6, it is unfortunately not possible. With the KAME implementation, a tunnel towards a mobile node is indeed not represented by an active interface by the operating system. This therefore prevents the MLD proxy from being able to use the tunnels. For this reason, the implementation of the MLD proxy is currently limited to the Linux operating system.

3.3 Implementation details

The MLD proxy has been implemented as a user-space application. It is therefore very portable and should thus work with the recent release of MIPL 2.0. It was extensively tested with MIPL 1.1. Moreover it supports multiple mobile nodes and multiple multicast groups. We believe that implementing the MLD proxy in user space was a very good choice as it greatly simplifies the maintenance and the portability of the code. For example a similar approach was followed for the new version of MIPL (i.e. MIPL 2.0) whose main functionalities were moved from kernel to user-space (when compared to MIPL 1.1).

The particularity that made it possible to implement the MLD proxy in user-space is that recent versions of the Linux kernel use by default the version 2 of the MLD protocol [5]. With this version, MLD report messages are sent to the “all MLDv2 routers” address (ff02::16). Therefore the MLD proxy can join this group and receive the report messages sent my the mobile nodes via the tunnel interfaces. With MLD version 1, this would have not been possible as report messages are sent to the group address contained in the MLD message.

The design of the MLD proxy is illustrated by Figure 3.



(1) Used to send/receive MLD messages to/from LAN

(2) Used to receive multicast data for groups joined on behalf of the mobile nodes. Sock_raw is used as the UDP port numbers are not known by the MLD proxy.

(3) Used to forward multicast data to appropriate interfaces. PF_PACKET is used as original IPv6 header must remain unchanged.

Figure 3. MLD proxy block diagram

Following the specifications in [4], the MLD proxy executes the router part of the MLD protocol on the tunnel interfaces, and the host part of the MLD protocol on the LAN interface. In particular, it means that the MLD proxy sends MLD query messages towards the tunnel interfaces. If a mobile user wishes to use the MIP6 tunnel for multicasting, it must therefore specify that in the routing table of its system. With Linux, this can be specified with the command (as root) :

```
route -A inet6 add ff0e::/16 dev mip6mnha1
```

where mip6mnha1 is the name of the tunnel interface towards the MIP6 home agent. Note that in this case, all multicast groups with global scope (ff0e::/16) will be joined via the tunnel interface.

3.4 Configuration

The operational parameters of the MLD proxy are specified in a configuration file whose format is shown below. Note that the MLD proxy can send a MLD version 1 report on the LAN interface when it receives an MLD report from a mobile node. This is only used if the multicast router of the LAN is by default in MLD version 1 mode. In some situations, this option might be needed.

```
# debug level, 0 means "no log", 1 is the standard level and 2 means verbose
# the default value is 1
# if DEBUG_LVL == 2, see log file for maximum log
DEBUG_LVL = 2

# multicast model used, values can be ASM, SSM or BOTH is the two are available
# default is ASM
MC_MODE = ASM

# the name of the LAN interface which will be used to join the groups
# default is eth0
INT_LAN = eth0

# the common part (i.e. prefix) of the MIPL tunnels' names towards mobile nodes
# MIPL default is ip6tnl
PREFIX_TUNNEL = ip6tnl

# The index of the first tunnel
# i.e. the final name becomes PREFIX_TUNNEL followed by FIRST_TUNNEL
# by default it becomes ip6tnl1
FIRST_TUNNEL = 1

# maximum number of tunnels to MN : put as many as expected number of MNs
# default is 1
NB_TUNNELS = 1

# the multicast scopes we will forward
# GLOBAL means only global scopes (default)
# SITE means site and global scopes
# LINK means link, site and global scopes
FWD_SCOPE = GLOBAL

# flag that allows or forbids the MN to be multicast sources
# default is 1 (allowed), 0 means forbiden
MN_SRC = 1

# we can send MLDv1 REPORTs right after a MLDv2 REPORT is received if the
# designated router is a MLDv1 router
# 1 means yes, 0 means no (default)
IMMEDIATE_REPORT_V1 = 0

# the value of the timer for an interface in the forwarding table
# default is the MLD default, 260 seconds
ENTRY_VAL = 300
```

4 Public MLD proxy at ULP

ULP has installed an MLD proxy that is publicly available via the 6NET network. This public service allows any mobile node using MIPL to subscribe to multicast groups via 6NET and the MLD proxy hosted at ULP. The name and address of the MLD proxy home agent are :

```
prox6net.u-strasbg.fr
2001:660:4701:1001:ffff::3
```

This machine is a MIPL 1.1 home agent. To use this service, a mobile user using MIPL must put the address of the MLD proxy in its MIPL configuration file. The user must also configure the routing table of its system as explained at the end of section 3.3. Any existing IPv6 multicast group can be joined via the MLD proxy.

5 Conclusions

At the time of writing, the integration of multicast and mobility is clearly still in an experimental stage. In particular, the KAME and MIPL implementations of the latest version of Mobile IPv6 do not natively support bi-directional tunnelling of multicast data.

However, ULP expects to share its experience by the development and the distribution of our MLD proxy, which can promote the use of multicast communications with mobile devices. To fulfil this objective, we are having contacts with the MIPL development team in order to distribute our MLD proxy as an official extension of MIPL. Our aim is to provide a flexible tool that is easy to install and configure, and which exhibits a satisfactory performance in terms of data forwarding.

Hopefully, the integration of multicast and mobility could be accelerated with the future deployment of native multicast routing. Such deployment is already being accelerated in order to provide multimedia services efficiently. For example, some ADSL network operators in France start deploying IPv4 multicast in order to propose an Internet TV service to their customers, i.e. traditional TV channels will be available via the ADSL network. In the mean time, the transition to IP version 6 is an adequate moment to deploy multicast routing at a very large scale, thus turning the future next generation Internet into a next generation *multimedia-ready* Internet. This would allow the rapid development of multicast services for mobile nodes, such as Internet TV and radio. One can imagine a next generation mobile phone that is also a TV and radio receiver. Such a revolution could trigger a large attraction towards the next generation of mobile phone technologies. Such services could be the "killer applications" that would accelerate the deployment of IP version 6 along with the next generation of mobile phone networks.

References

[1] C. Jelger, and T. Noel, "**Multicast with mobile hosts: analysis and performance evaluation**," 6NET deliverable 4.1.5 version 1, December 2003.

[2] D. Johnson, C. Perkins, and J. Arkko, "**Mobility Support in IPv6**," IETF RFC 3775, June 2004.

[3] C. Jelger, and T. Noel, "**Multicast for Mobile Hosts in IP Networks: Progress and Challenges,**" *IEEE Wireless Communications Magazine*, vol. 9, no. 5, pp. 58-64, October 2002.

[4] B. Fenner, H. He, B. Haberman, and H. Sandick, "**IGMP/MLD-based Multicast Forwarding ("IGMP/MLD Proxying")**," IETF Internet Draft, draft-ietf-magma-igmp-proxy-06.txt, April 2004, work in progress.

[5] R. Vida, and L. Costa (Editors), "**Multicast Listener Discovery Version 2 (MLDv2) for IPv6**," IETF RFC 3810, June 2004.

Abbreviations / Glossary

ADSL	Asymmetric Digital Subscriber Line
BU	Binding Update
CoA	Care-of Address
HA	Home Agent
IETF	Internet Engineering Task Force
ISP	Internet Service Provider
LAN	Local Area Network
MAC	Medium Access Control
MFC	Multicast Forwarding Cache
MIP6	Mobile IP version 6
MLD	Multicast Listener Discovery
MN	Mobile Node
ULP	Université Louis Pasteur