This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the ICC 2007 proceedings.

MobiSplit in a Virtualized, Multi-Device Environment

Julien Abeillé, Rui L. Aguiar, Joao Girao, Telemaco Melia, Ignacio Soto and Patrick Stupar

***Abstract*—This paper details a novel architecture, MobiSplit [17], for managing mobility in future IP based networks. The architecture separates mobility management in two levels, local and global, that are managed in completely independent ways. We describe how such a mobility architecture can be used to support a new paradigm in mobility. By combining the user's identity with a multi physical virtual terminal we treat the movement of people rather than their physical manifestations in one device. We conclude by analyzing the concrete system, built from this new architecture and existing protocols, in terms of scalability, flexibility and security.**

I. INTRODUCTION

The increasing complexity being perceived in next generation mobile networks, with multi-mode terminals always best connected, with multiple types of network available, both operator and community supported, has brought mobility issues into a central role for the future networks.

The other trend we've observed is that users are becoming more and more detached from their physical devices. While it is true in IP networks today that a user is comprehended by the network as the device it owns, the pervasive component in current research tells a different story. People will not only own multiple devices, from PDAs to laptops and even powerful body-ware and in-body sensor networks, but also interact with public devices which enhance the user's experience depending on his context and preferences. These devices can be accessed by anyone at anytime, which means that mobility is associated to the person's perceived identity and not to the individual devices he interplays with.

In this context, there are large initiatives, both industry and academia led, that address the multiple aspects of mobility. The EU-funded project Daidalos is one such project, addressing architectures for future networks. Starting from current trends discussed in standardization organizations, the Daidalos architecture is a major breakpoint from traditional approaches in IP networks.

In particular, we make use of a new mobility architecture, called MobiSplit, to support these new trends as well as continuing to support more traditional views on device mobility. The base element of this architecture is the splitting of mobility management in two domains, Local Mobility Domain (LMD) and Global Mobility Domain (GMD), assuming the IP protocol as basic architecture element. This splitting is done according administrative domain considerations. Seamless handovers and multi-technology local domains are also supported.

We not only consider an architecture which allows a device to be mobile and multi-homed, but extend these concepts to the virtual terminal which consists of multiple physical devices. This enables the architecture to treat the movement of a user as person in a seamless way. The user's identifier binds the virtual terminal together and identifies the user, rather than the devices, as the anchor for application end-points. These end-points can move as the physical devices where they are attached or from one interface to another in the virtual terminal. Since these interfaces can correspond to different physical devices, what we propose is an architecture which allows a person to be mobile, independent of his physical manifestations in the network.

The paper is structured as follows: Section II describes previous work in the field of localized mobility architectures and introduces MobiSplit as the basis for this work. Section III describes a new paradigm in mobility and how it can be achieved. Section IV presents a short analysis on scalability, flexibility and security of our proposal. Finally, Section V concludes the paper and possible future directions.

II. RELATED WORK

A. Mobility architectures

Discussions on heterogeneous networks agree on the need of a common protocol for communication, the IP protocol. Mobility is also supported at the IP level, with Mobile IP (MIP) becoming intrinsically supported in IPv6 (or with novel proposals such as HIP [1]). MIPv6 has nevertheless well-known deficiencies both in terms of performance and functionalities. Thus most of the research being done recently has been focused in these aspects, in particular along the lines of localized optimization of mobility behavior, separating the local mobility from the global mobility.

The localized mobility proposals aimed initially to reduce signaling outside the local domain, and improve efficiency by managing part of the mobility closer to the mobile node

This work was supported in part by IST FP6 Integrated Project DAIDALOS. DAIDALOS receives research funding from the European Community's Sixth Framework Program. Apart from this, the European Commission has no responsibility for the content of this paper.

Julien Abeille, Joao Girao and Telemaco Melia are with NEC Network Laboratories, Kurfuersten Anlage 36, Heidelberg, Germany (e-mail: {Julien.Abeille, Joao.Girao, Telemaco.Melia}@netlab.nec.de).

Patrick Stupar is with Telecom Italia; via Reiss Romoli 274, Torino (e-mail: patrick.stupar@telecomitalia.it).

Ignacio Soto is with Universidad Carlos III de Madrid; Av. Universidad 30; Leganés, Madrid, Spain (e-mail: isoto@it.uc3m.es).

Rui L. Aguiar is with Instituto de Telecomunicações, Campus Universitário de Santiago, Aveiro, Portugal (e-mail: ruilaa@av.it.pt)

1-4244-0353-7/07/$25.00 ©2007 IEEE

MN (reducing handover delays). Recently, operational exploitation considerations are gaining increasing relevance:

1) Host based localized mobility management

Initial localized mobility techniques were host-based, i.e. hosts had to handle signaling, and to be aware of local and global signaling protocols. Most relevant previous protocols were HMIP and Cellular IP.

The basis for HMIPv6 ([11]) is the design of a hierarchy of mobility domains. A MN is anchored to a gateway at each level of the hierarchy. If, due to the MN movement, the gateway of a node at one level changes, mobility is handled in the upper level using MIPv6 extended signaling. But movements inside a gateway domain are transparent to upper levels.

In Cellular IP [2][3] a gateway separates the global and local domain. When moving inside the local domain, the MN exchanges signaling with the network, and host routes are created or updated on the routers between the MN and the gateway. A global mobility protocol such as MIP are HIP can be used to handle mobility between local domains.

2) Network based localized mobility management

Aspects of network control and operation have led to the renewed development of localized mobility solutions, including at standardization level in IETF. Unlike host-based mobility where mobile terminals signal a location change to the network to achieve reachability, network based approaches relocate relevant functionality for mobility management from the mobile terminal to the network.

Several approaches were considered, mainly proxy versions of MIPv6 ([8]) and the NetLMM WG design team proposal ([7]). These proposals share the same architectural design but differ in the signalling exchanged between functional entities. MIPv6 based signalling was considered inappropriate by the NetLMM design team for several reasons. As NetLMM is used in our approach, we describe it in the next section.

3) The NetLMM protocol

The NetLMM approach ([7]) is currently being designed in the IETF NetLMM Working Group. [5] and [6] define the requirements and rational for NetLMM.

Figure 1 shows the entities involved in NetLMM. The Local Mobility Anchor (LMA) is a router defining the edge between the NetLMM domain and the core network. The Mobility Access Gateway (MAG) is the Access Router for the MN. The NetLMM operation is located between MAGs and LMAs. The movement of the MN is perceived by the network through standard L2 or Neighbour Discovery operation.

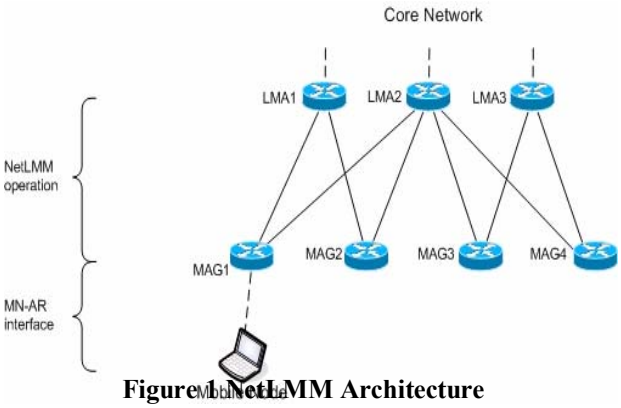

Figure 1. NetLMM Architecture

When the MN first attaches to the network, it obtains an IP address from a prefix owned by the LMA. A route to the MN is installed on LMA and MAG, and a source route is installed on the MAG to route traffic from the MN through the LMA; upon movement of the MN, the routes are updated in the LMA and the MAGs involved (previous MAG and new MAG). As long as the MN remains in the same NetLMM domain, it keeps the same IP address

The forwarding method between the MAG and the LMA can be IPv6 in IPv6 tunnelling, General Routing Encapsulation ([10]) or Multi Protocol Label Switching ([11]). Such forwarding methods allow the use of standard routers on the path between the MAGs and LMAs: this considerably reduces the signaling in the LMD and avoids the extensive use of resources (routing tables) in the intermediate nodes.

The NetLMM protocol can be used in conjunction with a Global mobility protocol, to handle mobility between local domains. It only supports reactive handover and does not consider the support for multiple technologies within the same LMD, which is quite limitative.

4) MobiSplit

The present work makes use of some features of the MobiSplit architecture ([17]). MobiSplit consists in a three-tiered mobility architecture, where the mobility of a MN is handled using 802.21 as a common interface between MN and Access Router, a Network Based mobility protocol in the access network, and a host based mobility protocol in the core. Based on this design, some extensions to the existing protocols are proposed, which provide additional features. The main feature we exploit in this paper is the "multihoming approach in the LMD". We extend this design to support multiple devices and introduce a new paradigm in mobility. To present our solution as a whole, we detail the solution as a whole in section III.

5) Session Mobility and Virtual Terminal

Session mobility has long been understood by the community as a session level issue. In [18], the authors present a session mobility protocol based on the well-known Session Initiation Protocol (SIP). Solutions of this type are bound to the application and require both end-points (or an end-point and a proxy) to support it.

The concept of a virtual terminal composed by several

physical devices is not new. One such instantiation is proposed by Fu, et al. in [19].

Our approach differs from all of these in that it proposes the support of a virtual terminal through network mechanisms. The mobility architecture should also not require modification in the correspondent node and only minimum support from the application side.

B. *Identity architectures*

UMTS networks separate the concept of user and terminal by means of the SIM card. This allows users of these networks to use different terminals by introducing in them their SIM cards, so that the network recognize the particular user (subscription) and deal with her appropriately. Also, it is possible in some terminals to include more than one SIM card so different identities can be used from the terminal. A typical scenario is that the same user has a personal identity and a professional identity.

There are some limitations in this solution. First, the terminals are restricted to UMTS technology. And second, the simultaneous use of the identity is very restricted. Our proposal is for an environment of heterogeneous technologies and a real identity centered solution in which a user can use several terminals simultaneously, and different users can share the same terminal.

HIP [22] introduces a new cryptographic namespace for identification. It also eliminates the dual role of IP addresses, providing added flexibility for solutions which provide location privacy. However, it does not consider the usage of the same identifier across different devices.

III. A New Paradigm in Mobility

In all architectures and mobility management solutions so far, we perceive the end points of the protocols as devices, interfaces or applications. In reality, the concept of user has far been neglected and forgotten, even if it is the user who moves and interacts with the different terminals and who runs applications. We believe that in the future, as the terminals become more aware of their users, as part of context and personalization, and the identity of the users plays the major role in computer communications, we can no longer take the simplified view of the network as devices, interfaces and applications: nodes in a graph.

Our vision is that the user may own some devices but also interact with others which are part of the infra-structure, and for which he may potentially have to pay. All this we believe the user should experience without ever terminating his sessions and without its communication peers knowing it.

As we introduced in the previous section, MobiSplit deals with mobility and multi-homing at IP level and below. It provides an architecture which can handle the separation of flows and mobility management of a terminal's interfaces. In this section we deal with the problem of extending this architecture to deal with virtual terminals as if this virtual terminal was one single device.

A virtual terminal is no more than a set of terminals within the same LMD, as defined in the previous section, bound by the identifier of a user. Although the inherent authentication mechanisms are out of the scope of this paper, it suffices to say that a virtual terminal can be built from any single authentication point of the user and extended to any device which recognises the user's permissions and authentication credentials. Once a virtual terminal is built, for the network and under the architecture proposed, it acts as if it was one single device. Hence, this approach does not require a MobiSplit compliant architecture to undergo any modifications. Also, since we make no restrictions on the number of CoAs per interface nor in the number of users per terminal, it may be that more than one virtual terminal is composed of, at least partially, the same device. How to handle conflicts and application management is outside the scope of this paper. This issue is application dependent and not restricted to our approach[1].

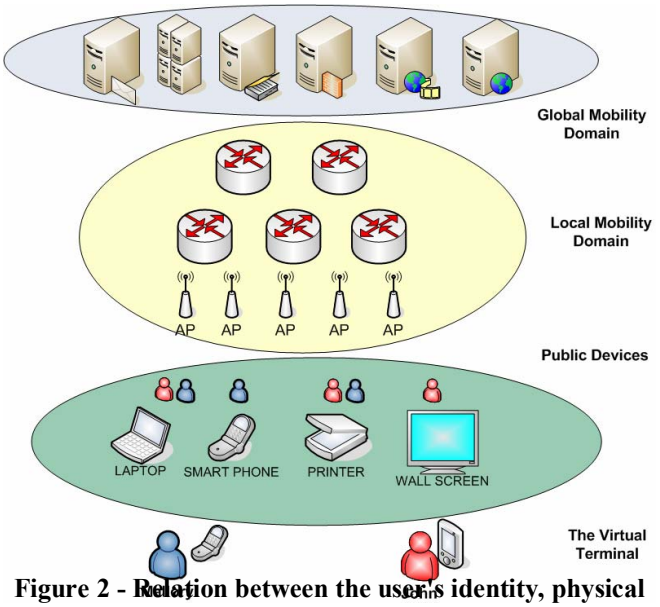

Figure 2 - Relation between the user's identity, physical devices and network.

In Figure 2 we can see how two users can share devices and combine devices to enhance their interaction with the network and services. Furthermore, the figure also depicts how the different levels can be managed by different operators, including the physical world and user movement in the virtual terminal domain. As each user or application and no more the interface is selecting the network to attach to, there may be different networks selected for one interface. How to handle this is not in the scope of this paper, but note that recently, some hardware/software allows one interface to attach to several networks in the same time (e.g. WLAN atheros cards with madwifi driver).

[1] In fact there is no overlap in ports or addresses since the CoA of the user in the LMD is the address to which packets are addressed and this address is shared by all devices which are part of the virtual terminal. Also, different users use different CoAs even when sharing a physical terminal.

A. Mobility and multihoming

This section describes an integration of NetLMM with multihoming, as presented in MobiSplit architecture ([17]). MobiSplit defines an architecture for mobility management, and proposes extensions to the NetLMM protocol to support proactive handover, heterogeneous LMDs, multihoming, and improve the scalability of NetLMM. Multihoming is handled in the NetLMM domain, and is transparent to the core network. One can even envision that an enterprise or home form an LMD where mobility and multihoming is handled.

We are assuming a MN can have more than one interface active at the same time. Managing multihoming in the LMD offers various advantages. It allows access operators to manage the mobility of visiting MNs inside their domain without depending on an external operator. In particular, an access provider can optimize the use of its resources by using the best interface/technology to provide connectivity to the MN, according to the MN's preferences, but also to the general situation of the network. The network can for example decide to move a flow to a different interface on the MN in order to perform load balancing. We discuss in this section the addressing scheme we use, the signaling required to map flows to interfaces, and the way we handle the routing inside the LMD.

To hide the multihoming from the core network, all the interfaces on the MN will use the same IP address. We define a new NetLMM identifier (the interface identifier, in addition to the MN-ID), to make this possible. When a MN activates a new interface, the LMA knows the NetLMM prefix assigned to this interface should be the same as for all the interfaces of the MN[2]. The MN will use the same suffix as on the other interfaces and therefore configure the same IP address on all active interfaces.

The MN and the LMA must agree on the mapping between flows and interfaces, so the uplink and downlink traffic belonging to the same flow follow the same path. An easy solution is that the MN chooses the interface for the traffic it initiates, and the LMA chooses the interface for the traffic initiated in the core network. This is not very restrictive, as the normal procedure for intra-LMD handovers can be afterwards used to move a flow from one interface to another.

As regards routing, in the downlink case, the traffic is always destined to the same IP address. The LMA has to check the flow identifier (the flow could be identified by the source and destination addresses and source and destination ports in the packet), and it forwards the traffic to the Access Router (AR) which the MN's interface is attached to. In the uplink case, for each flow, the MN just selects the correct interface to send the packet on.

The important issue to notice is that in the MobiSplit solution, that address is not seen by the nodes between the LMA and AR, as the packets are routed through a tunnel. For this reason, it is not a problem if the address is duplicated. The LMA can choose the interface in which the MN will receive the traffic just by using the appropriate tunnel.

Note that some problems appear in a situation in which two interfaces of the MN are attached to the same AR. To overcome this problem, we could have on the AR one routing table and neighbour cache per tunnel.

B. Virtual terminal and mobility

To simplify the process, we divide it in three stages: Discovery, Configuration and Network Signalling. We now detail these phases and provide an instantiation of the framework as well as a few considerations on the way the terminal functionality is distributed.

1) Discovery

In this phase we already assume the user is operating a terminal. He has already authenticated and proven his identity and is registered to the network under this one device. While operating this terminal, or under the user's personalization rules, the user makes use of service discovery protocols to find other devices in the same LMD. The services provided by these devices (e.g. video display, speakers) are then matched for compatibility with the applications the user is currently running (or will run in the future).

Once this process is completed the terminal will have a list of devices which the user is entitled to access and also their compatibility with the user's needs.

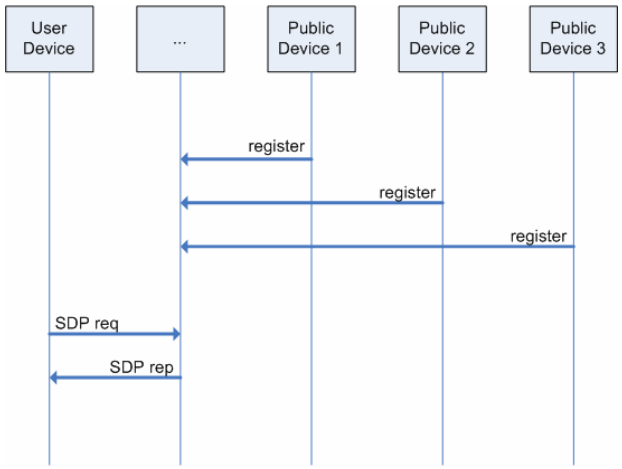

Figure 3 - High-lever message sequence chart of the discovery procedure.

In Figure 3 we represent the actual message flow during this phase. The protocol assumes the devices pre-register with a common point in the network which is used by the protocol stack to query about surrounding devices. A distributed approach could also be used.

2) Configuration

Once the user is authenticated, he will proceed to the configuration of the devices which he now wants to become

[2] If we did not have this new identifier, when a MN activates a new interface the LMA would think the MN is performing a handover

part of his virtual terminal. By either using the network, or in a peer-to-peer fashion, one mobile terminal configures another to receive part or all of the flows it is currently receiving.

We begin by setting up a new CoA in this terminal which matches the CoA of the user in the LMD. This process is inline with the multi-homing process described in the previous section. Once this is done and the tunnel established to the LMA, the transport and application levels are configured. Depending on the actual application and transport protocols involved different parameters must be passed from one terminal to another. While it is trivial to configure the receiving end of an RTP stream, where the transport level parameters are reduced to a port number and possibly the current sequence number, the process to transfer a connection oriented transport protocol, such as TCP, is fairly more complex. To validate our work we'll focus on the simple case since it is obvious that this step is highly dependent on the transport and application used.

3) Network Signaling

Finally, now that the terminal is ready to receive the flow, we signal the network to divert traffic of a certain flow in that particular tunnel which culminates at the new device. Please note that since we have imposed no restrictions on these devices, other than they should be able to communicate with each other and belong to the same LMD, nothing impedes any of the devices to be mobile.

All further mobility and multi-homing is handled in the exact same way as MobiSplit but under the assumption of a virtual terminal. Figure 4 represents the message flow diagram corresponding to the insertion of a new flow handover policy and subsequent enforcement of this policy. This mechanism is used to transfer flows associated to session or application level data from one interface in the virtual device to another.

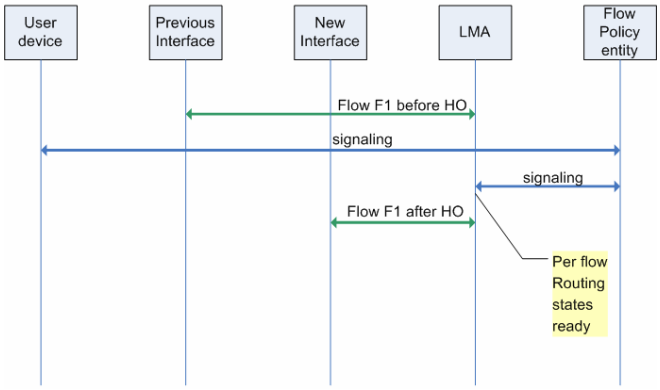

Figure 4 - High-lever message sequence chart of the flow handover procedure.

4) Instantiation

The additional mechanisms to MobiSplit can be implemented by using existing protocols. The first phase, *discovery*, can be an application of the IETF Simple Service Discovery Protocol (SSDP) [20]. This protocol is SIP based and deals with both the discovery of new services, in this case devices, and their capabilities. By describing these capabilities in terms of applications the terminal can run and transport protocols it supports, we can easily build the list of possible candidates to extend the virtual terminal.

The simplest instance of the configuration phase can be achieved by applying a Remote Procedure Call (RPC) protocol. An example of a well-known protocol which provides the necessary functionality is CORBA [21]. This type of protocol allows the user to run remote commands, such as the configuration of the interface and tunnel establishment as well as the actual running of the application in a remote terminal. We can use RPC to perform the configuration phase in the device we have decided to add to the virtual terminal to. This approach is naturally restricted to the parameters the applications allow and, in some cases, may prove insufficient. For a seamless integration of any application, a protocol designed for this effect might be required.

The final stage of this instantiation has already been described in the section above. MobiSplit requires no modifications to support this extension.

One of the disadvantages of this instantiation appears due to its distributed nature. Since the state, applications and protocols run concurrently on different physical devices we must have a set of rules on how to handle the loss of communication between them or a protocol which keeps the state synchronized across the different devices. We propose a simple rule based on a keep-alive mechanism in the tunneling protocol which connects the interface to the LMA. Should the keep-alive mechanism fail, the flow should be returned to the original interface and subsequent tunnel. Should this tunnel no longer exist, the flow should be dropped.

Please also note that this approach can also be combined with the protocols and architectures in Section II.B.

IV. Analysis

In this section we provide a short analysis of the architecture and framework for mobility presented in this paper.

A. Scalability and Flexibility

In terms of scalability, the size of the LMD determines the amount of state required to keep in the intermediate nodes. We believe such considerations are not impeditive in home or even small to medium enterprises but may play an important metric in large enterprises or operator networks. Since the virtual terminal only impacts the number of registered interfaces at the LMA it should only affect scalability in the case multiple users share a physical device. The rest of the communication and state is handled by the terminals themselves.

The administrator of the local domain has full control of the mobility and multihoming protocols. This allows for tailored solutions. Since the support for the virtual terminal

construction comes only from the terminal's support of the protocols, several different instantiations of these mechanisms are feasible under the same LMD.

B. Security

1) Attacker Model

We base our security analysis on the information an attacker obtains on a user and his devices when the attacker is outside and inside the LMD. We further consider the case where the attacker can share a device with the user, i.e. if the VT of the attacker includes a terminal which is also part of the VT of the user.

2) Location Privacy

Both NetLMM and subsequently MobiSplit, provide an inherent protection to the location of the user. The LMD shadows the current location of the terminals within it since the CoA is mapped to the LMA and the end points of the tunnels established between the LMA and the terminals are never revealed to the outside. We can further extend this reasoning to include the case in which the user owns a virtual terminal. The attacker has no form of knowing whether the user in the LMD has more than one device, how these devices are moving and how they relate to each other. The location protection given by the LMD depends, therefore, in its size.

If the attacker is inside the LMD it can try to find to which AR one of the user's terminals is attached. Depending on how the *discovery* and *configuration* protocols are used, these might reveal information about the user's virtual terminal. However, since all communication within the LMD is also based on the CoA and travels via the LMA, the attacker cannot learn any more than the outside attacker unless he happens to be under the same AR or shares the same link.

3) Network security

The network authentication problem can be mitigated by a strong notion of identity where the user can be authenticated, and not only the device. Although the data path can be protected with, for example, IPSec, the signaling which allows the user to transfer flows in his VT is much more critical. The attacker could perform both denial of service (DoS) attacks, hijacking and impersonation attacks. Since the user is not bound to a terminal, his credentials must be distributed over several devices, which increases the chances for an attacker to obtain them. In order to prevent these cases, the signaling must be protected with a key bound to the user under one common identity for all his devices. This opens the door to attacks on the user's privacy from the access network provider, where he can easily see which users are associated to which devices.

4) From the User to the Device

The boundary between different users in the same device is well defined in operating systems such as Linux. However, since the credentials from the user have to be passed from one device to another, not to depend on weak secrets, more points of failure exist. There are several approaches on how to protect this last meter, such as MANA [23], which focus on how to first authenticate the channel between a device the user owns and one he wishes to communicate with. Other possibilities, such as basing the initial authentication on biometrics are also viable.

There are further security issues which stem from the fact that several users share the same device but these are Operating System specific and will not be addressed in this context.

V. Protocol Comparison

Many protocols have been developed for handling mobility, as we have seen. Over these protocols, many variants have been developed. For instance, CIP principles could be also used in a network based localized mobility solution. Without being exhaustive, Table 1 shortly summarizes some characteristics of different protocols.

Table 1 – Protocol Comparison

	CIP	H MIP	MIP	Net LM M	SIP	Mobi spilt
Local/global	L	L	G	L+G	G	L+G
Multihomin g	N	N	N	N	N	Y
Network overhead	Hig h	Low	Lo w	Low	Hig h	Low
Seamless handover	Y	Y	N	Y	N	Y
Terminal modification	Y	N	N	N	Y	Mini mal
Application Specific	N	N	N	N	Y	N
Support Virtual Terminal	N	N	N	N	Y	Y

The advantages of handling mobility at the lower layers are clear: one protocol can suit different applications. In this aspect, the combination we propose of MobiSplit and the virtual terminal allows for the combination with the most benefits. Our solution does not require modifications on the terminal, other than for installing rules in the LMA and RPC support, provides a solution which fits all types of applications and allows the user to move in the new paradigm previously introduced.

VI. Conclusion

In this article we have presented a new paradigm for future mobile operators. The architecture recognizes the current trend in networks to a heterogeneous landscape of access providers. In this environment it is important to give the access providers the flexibility of managing the mobility inside their domains according to their needs, technologies, and requirements, without being conditioned by how mobility is managed in other domains.

To cope with this concept, the architecture proposed in this paper splits the mobility management in two levels: the local domain and the global domain; and the management of the mobility in these two levels is kept completely independent.

The architecture proposal also supports mobility as seen from the user's perspective across multiple networks but also multiple devices. We support flow handovers also between different physical devices which are bound by the user's identifier.

Further research in this area should account for the generic protocol which is required in the configuration phase, as described, and also at how the user's identity is bound to that of a device in order to perform the operation securely.

REFERENCES

[1] R. Moskowitz, et al, "Host Identity Protocol", draft-ietf-hip-base-06, June 2006.

[2] A. T. Campbell, et al, "Design, Implementation, and Evaluation of Cellular IP"; IEEE Personal Communications, August 2000.

[3] A. G. Valko, et al, "Cellular IP", IETF Internet Draft, draft-valko-cellularip-00.txt, November 1998.

[4] "Draft IEEE Standard for Local and Metropolitan Area Networks: Media Independent Handover Services (Draft 01.00)." IEEE

[5] Kempf, J., et al, "Problem Statement for Network-based Localized Mobility Management", draft-ietf-netlmm-nohost-ps-05, September 2006.

[6] Kempf, J., et al, "Goals for Network-based Localized Mobility Management (NETLMM)", draft-ietf-netlmm-nohost-req-04, August 2006.

[7] Levkowetz, H., et al, "NetLMM Protocol", draft-giaretta-netlmm-dt-protocol-02, October 2006.

[8] S. Gundavelli, et al., Proxy Mobile IPv6, "draft-sgundave-mip6-proxymip6-01", January 2007

[9] Laganier, J., et al, "Network-based Localized Mobility Management Interface between Mobile Node and Access Router", draft-laganier-netlmm-mn-ar-if-00, March 2006.

[10] Farinacci, D., et al, "Generic Routing Encapsulation (GRE)", RFC 2784, March 2000.

[11] Rosen, E., et al, Viswanathan, A., and R. Callon, "Multiprotocol Label Switching Architecture", RFC 3031, January 2001.

[12] Soliman, H., et al, "Hierarchical Mobile IPv6 Mobility Management (HMIPv6)", RFC 4140, August, 2005.

[13] Hillebrand, J., et al, "Quality-of-Service Signalling for Next-Generation IP-Based Mobile Networks", IEEE Communications Magazine, June 2004, pp 72-79.

[14] Marques, V., et al, Evaluation of a Mobile IPv6-based Architecture Supporting User Mobility, Quality of Service and AAAC in Heterogeneous Networks, IEEE Journal on Selected Areas in Communications, Special Issue "Wireless Overlay Networks Based on Mobile IPv6", Vol. 23, no. 11, Nov 2005.

[15] Perez-Costa, X., et al, "A performance comparison of Mobile IPv6, Hierarchical Mobile IPv6, fast handovers for Mobile IPv6 and their combination", ACM SIGMOBILE Mobile Computing and Communications Review, Volume 7, Issue 4, October 2003

[16] Wang, H., et al, NETLMM Protocol, draft-wanghui-netlmm-protocol-00, April 2006

[17] Abeille, J., et al, "MobiSplit, : a scalable approach to emerging mobility networks", MobiArch'06, December 2006

[18] Schulzrinne, H. and Wedlund, E. "Application-layer mobility using SIP", SIGMOBILE Mob. Comput. Commun. Rev. 4, 3 (Jul. 2000), 47-57.

[19] R. Y. Fu, H. Su, J. C. Fletcher, W. Li, X. X. Liu, S. W. Zhao, and C. Y. Chi, "A framework for device capability on demand and virtual device user experience", IBM Journal of Research and Development, volume 48, number 5/6, 2004.

[20] Y. Goland, et al, "Simple Service Discovery Protocol/1.0 Operating without an Arbiter", draft-cai-ssdp-v1-03.txt, October 28, 1999, work in progress.

[21] The Object Management Group, "Common Object Request Broker Architecture Specification 2.2", http://www.omg.org.

[22] R. Moskowitz and P. Nikander, "Host Identity Protocol (HIP) Architecture", IETF RFC 4423, May, 2006.

[23] C. Gehrmann, C. Mitchell, K. Nyberg, "Manual Authentication for Wireless Devices", RSA Cryptobytes, vol. 7, pp. 29-37, 2004.