Mobility Management in ULOOP

Sebastian Peters¹, Denis Pozo Pardo¹, and Qing Zhou²

¹ DAI-Labor, Technische Universität Berlin, Berlin, Germany
sebastian.peters, denis.pozo@dai-labor.de
² Huawei Technologies Düsseldorf GmbH, Düsseldorf, Germany
zhouqing@huawei.com

Abstract. This paper provides an overview of the mobility management solution designed and developed in the context of the ULOOP Project.

1 Introduction

The concept of user centric networking empowers the end-users as new Internet stakeholders with their own network resources, rather than considering them as just consumers and producers of content [1]. User Centric Networks (UCN) allow end-users to cooperate by sharing their network resources and services. It is crucial for the success of UCN vision to provide a certain level of quality of experience to the members of the UCN community, comparable to those expected from legacy network architectures. The dynamic environment of UCNs and the diversity of user entities (UE), acting either as micro-providers or as clients, are certain factors responsible for the increased complexity in designing such enabling mechanisms for UCNs. The EU IST FP7 ULOOP (User-centric Wireless Local Loop) Project [2] tries to address those complexities, in particular by designing and implementing the foundations for trust and incentive mechanisms, resource management, and mobility management in UCNs. This paper aims to provide a detailed overview of the mobility management aspects in the context of ULOOP Project.

The structure of the paper is as follows: Section 2 provides a general introduction on the Mobility Management (MM) challenges tackled in ULOOP, which includes the background and overall objectives. Section 3 describes the mobility coordination aspects in ULOOP for a single mobility domain, including the technical challenges, how they were tackled, and provides operational examples in form of use cases. In Section 4 we extend the perspective to multiple MM domains and explain how the mobility coordination works in this scope. Finally, Section 5 concludes the paper.

2 ULOOP Environment and Challenges from a Mobility Management Perspective

As mentioned earlier, the ULOOP project is based on the notion of User-centric Networks (UCNs), in which users cooperate by sharing wireless resources as well
as Internet services. The UCN environment envisioned in ULOOP is a highly dynamic one where user-provided entities, possibly providing various network support functions, may frequently appear and disappear. To deal with this dynamism, the ULOOP concept handles mobility of users in a way that is graceful towards disappearing mobility management entities such as mobility anchor points (MAPs).

Any ULOOP user, whether providing a part of the UCN infrastructure or just taking advantage of it, is seen as a member of the ULOOP community. Conversely, ULOOP communities are based on nodes made available based on users willingness, which means they are not provided and controlled by network operators. Within a ULOOP community, any ULOOP node assuming the role of gateway, can also be a potential MAP. Therefore it is relevant also to assist in an adequate coordination of the MAPs made available. Instead of defining a new MM solution, the definition of a signaling architecture that can assist in a better coordination of available MAPs in the dynamic UCN environment is the design methodology adopted in ULOOP. In other words, ULOOP networking architectures are expected to rely on existing mobility solutions (e.g. Proxy Mobile IPv6 - PMIPv6 [3], Mobile IPv6 - MIPv6 [4], Session Initiation Protocol - SIP [5]) and to address aspects that are crucial to sustain the least disruption when nodes roam across the dynamic infrastructure. PMIPv6 fits well into the design concept of UCNs as it does not involve the mobile node in the MM methodology. However, it is necessary to extend PMIPv6 to overcome the challenges posed by the UCN environment (i.e. an environment where the infrastructure is not provided by a network operator as it is usually the case with PMIPv6). In the remainder of this paper we will show novel MM concepts for the scope of UCNs, using PMIPv6 as a baseline solution.

In a common PMIPv6 environment the Local Mobility Domain (LMD) only supports a single mobility anchor. In UCNs this is problematic since a single-MAP environment poses a single point of failure. On account of this multi-MAP environments have been discussed, most notably in the PMIPv6 extension by Korhonen et al. called “Runtime Local Mobility Anchor (LMA) Assignment Support for Proxy Mobile IPv6” [6]. The runtime LMA assignment of [6] is designed for the purpose of load balancing in a multi-LMA environment with a static number of mobility anchors. However, in a UCN environment the number of available LMAs is not known at the time of deployment, as is implicitly assumed in [6]. In this paper the MAPs that are currently offering their anchoring service to MNs, and which are thus available for a dynamic selection are coordinated in the PMIPv6 domain by a broker-like entity that is described in detail in Section 3. Furthermore, PMIPv6 only provides mobility management for MNs of a single LMD, lacking support for roaming across multiple LMDs. A concept for inter-domain mobility based on P2P is detailed in Section 4.

To summarize, we identify the following set of technical challenges to be overcome with regard to mobility management aspects in ULOOP:

- ULOOP communities are based on nodes owned by Internet users and are out of the control of the operator. Their dynamic behavior which may even
be erratic, can cause unreliabilities from a mobility management perspective. The MCF is expected to assist in a better coordination of MAPs, in the face of this dynamic behavior.

- **ULOOP communities interoperability** to other systems requires a way to deal with the dynamic behavior of UCNs. A MCF domain shall coordinate with other MCF domains to provide for roaming of MNs across multiple MCF domains.

### 3 Mobility Coordination in ULOOP

As described in Section 2, UCNs create the need for a MM mechanism that is able to cope with the dynamic behaviour in order to still provide the best possible continuous network connection to a ULOOP Node. Another characteristic apart from the dynamic infrastructure is that the user provided infrastructures only have limited resources in terms of the available network bandwidth and computational potency. The mechanism that has been developed within ULOOP to meet these requirements has the following major components:

- Mobility Coordination Function (MCF).
- Mobility Anchor Points (MAPs).
- Mobility Access Gateway (MAG).

The MCF component serves as a broker of MAPs that helps in managing the limited resources of the ULOOP Gateways that offer MAP services by selecting MAPs on behalf of the users. The MAP component is located in a ULOOP Gateway and provides mobility anchor functionality to ULOOP Nodes that are associated with another ULOOP Gateway. This way ULOOP Nodes are less affected if the associated ULOOP Gateway disappears. It should be noted that the MAP component is based on the LMA functionality of PMIPv6 which is modified to provide a loose coupling between MAG and LMA. We refer to this modified version of the LMA as MAP. The MAG component also corresponds to the entity in PMIPv6, however, it is modified to be able to request an anchor node (re-)selection from the MCF.

Figure 1 provides an example of how these functional entities can be distributed within the user provided Gateways, thus making up the user provided MM infrastructure. It can be seen that each ULOOP Gateway runs a selection of the three ULOOP MM components. While the MAG is obligatory, users can choose whether they would like to provide a MAP or offer their Gateway to make the MCF service available. If multiple MCFs are available in a MCF domain, coordinator election algorithms could be used to retrieve the most suitable Gateway to provide the MCF functionality for the ULOOP community, however, this is not within the scope of this paper.

In Fig. 1 we have depicted three ULOOP communities as an example. ULOOP community 1 covers area A; ULOOP community 2 covers area B, and ULOOP community 3 covers area C. In area A the ULOOP node associates to ULOOP Gateway 1 (UGW1) in ULOOP community 1, which also provides a MAP and a
MCF service. When the ULOOP node moves to area B, UGW2 in ULOOP community 2 is selected as the ULOOP gateway and the MAG component obtains the mobility context and updates the MAP in UGW1 with a Proxy Binding Update (PBU). That way the ongoing session of the user continues, however, since the MAP is from a remote ULOOP community selecting a new MAP from the current community for newly started sessions is beneficial to the performance. As UGW2 does not provide a MAP functionality itself, a query is sent to the local MCF to select a new MAP for this user. The MAP in UGW3 is selected, being currently the only active MAP in this community. Finally, in area C the MCF has multiple MAPs registered and can select the one with the most free resources in terms of available network bandwidth or based on stability metrics (uptime).

While the ULOOP MM infrastructure is intended to be mainly provided by users, the MCF entity and MAPs can also be located in the access network of an operator to support the mobility of ULOOP users (as indicated by the 3GPP coverage area).

Fig. 1. Overview on Mobility Coordination concept
3.1 Mobility Coordination Function Architecture

On the highest abstraction level the Mobility Coordination Function Architecture is composed of a Mobility Decision (MD) sub-block and a Mobility Cooperation (MC) sub-block, as depicted in Fig. 2. The MD sub-block groups the functionality to decide on a suitable MAP for a ULOOP node based on certain triggers. The MC sub-block groups the functionality to provide context on available MAPs to the local, as well as remote instances of the MD sub-block. As we proceed to detail the design specification of the sub-blocks we now further decompose them into the smaller components of the respective blocks in Fig. 2. Inside the MD sub-block the Mobility Trigger & Information Collection is responsible for receiving triggers from other blocks and triggers the mobility decision. Subsequently the Mobility Mapping performs the mobility decision and selects a suitable MAP for the ULOOP node that caused the trigger. Within the MC sub-block the Mobility Context Register is responsible for providing a constantly updated registry of MAPs that is sorted either by the id or the current score of the MAP. Finally, the Inter-MCF Coordination component indicates the inclusion of information from other MCF domains, which is described in Sec. 4 of this paper. Situated at the left side of the figure the interface to the Mobility Tracking component (located in a ULOOP Node, optional) provides further input to help selecting a suitable MAP. It does so by tracking some properties of user mobility, and by providing an estimate of potential target gateways and time to move. For more information on the Mobility Tracking please refer to [].

On the top left side of Fig. 2 the interfaces to other ULOOP blocks are indicated. In ULOOP device-internal socket communication is used to share information that is relevant to MM functionality. Most notably the Resource Management block provides information on the current load and the overall available bandwidth, which the MAP then provides to the MCF to allow for load-balancing mechanisms.

Summing up, all the described components are working together to form what we call the Mobility Coordination Function (MCF) to coordinate the mobility of ULOOP nodes. Having described the architectural perspective we now explore how the MAG, MAP and MCF entities interoperate to achieve this architecture.

3.2 The Mobility Coordination Function (MCF)

Figure 3 provides an overview of the basic MCF functionality and the messages other MM entities exchange with the MCF. The ULOOP Gateway with MAG is depicted by the router symbol to the left, the MAP to the right and MCF in between them. The functionality of each component is implied by boxes, beginning with the initialization of the respective component on top to operational parts further down the column. The events at the MCF are as follows:

1. The MCF service initializes itself and prepares to receive incoming messages.
2. MAPs register at the MCF when they are ready to accept ULOOP Nodes.
3. The information obtained from a MAP registration message is inserted into the Mobility Context database.
4. MAG sends a message to trigger the MCF when a new user in need of a MAP connects to it.

5. The MCF then performs Mobility Mapping to select a suitable MAP for this user.

6. The decision taken in this process is then sent to the MAG.

The message protocol that is used between MCF and MAG and between MCF and MAP is described in more detail in the following paragraph. In addition to this the figure also shows how the Mobility Execution part relates to the Handover Support architecture.

The MCF Protocol. This part of the paper describes the messages that are exchanged among the MAG, MAP and MCF entities. The purpose of these messages is the exchange of information between MCF–MAP and MCF–MAG to enable decision making at the MCF. The specification of these messages is tailored to the specific needs of ULOOP mobility aspects and therefore implements a custom protocol. In Figure 3 the respective messages are indicated by blue arrows. The transmission of these messages is handled by the TCP protocol.

Messages from MAP to MCF. Messages from MAP to the MCF are sent on three different occasions:
MAP registration
MAP update / keep-alive
MAP deregistration

The involved parameters in the message are:

- MAP Message Reason - The reason code for MCF sending this message.
- MAP ID - The IP address of the MAP (IPv6).
- MAP Upload Bandwidth - Overall upload bandwidth of the MAP specified in 100kbit/s unit.
- MAP User Provided Flag - Indicates whether the MAP is provided by a user or by a network operator.
- MAP Registration Flag - Indicates whether the message is a registration or a deregistration message
- MAP Congestion status - Indicates the congestion status of the MAP, interpretable as a percentage value of used resources.

Message from MAG to MCF. A Request MAP for MN trigger message is sent by the MAG whenever a MN requires a new MAP.

The involved parameters in the message are:

- MN Message Reason - The reason code for sending this message
- MN ID - The IP address of the MN (IPv6)
- MN Registration Flag - Indicates whether the message is a registration or a deregistration message
- MAG ID - The IP address of the MAG (IPv6)

Message from MCF to MAG. The Selected MAP for MN response message is sent by the MCF to the MAG to answer the request message with a suitable MAP for this user. The involved parameters in the message are:

- MAP ID - The IP address of the MAP
- MN ID - The IP address of the MN
- MCF Message Reason - The reason code for sending this message

This protocol description concludes the single MCF domain perspective and we will now shift our attention to the interworking of MCF domains. In Sec. 3 we have already hinted at this in the top left part of Fig. 1 showing the P2P interconnection of MCF domains, however, so far we just explained that the mobility context information for a specific user is retrieved in some way. The following section explains the involved concepts in detail.

4 Distributing the Mobility Coordination Function

The MCF entity that has been introduced in the previous chapter of this paper is responsible for the coordination of anchor nodes within a so called MCF
Such a MCF domain corresponds to a PMIPv6 domain in which the MCF manages anchor nodes.

Since the MCF is playing the decision maker role, the number of addressed requests depends directly on two factors: the number of units to be managed and the proper movement of them. So, we may foresee the first constraint that this element introduces in the scenario; i.e. the amount of users managed by one MCF has to be limited. Therefore, in terms of scalability, the extension from one to multiple domains seems to be the logical solution to overcome this issue. Apart from scalability, the MCF entity itself introduces the well-known drawbacks associated to centralized systems because of its way of working. The proposed system deals with these inconveniences, such as bottlenecks, congestion and single point of failure. On the other hand, there is the stronger constraint of meeting the design concept of UCNs. Mobility management is one potential service provided within a UCN, and it is of utter importance as it greatly enhances the user experience of UCN users. Having said that, the challenge of Mobility Management (MM) in UCNs is that the user who offers the connectivity shall not be burdened with complicated administrative or technical tasks in order to provide this service to other users. While the MCF solves the problem of anchor node discovery, selection and assignment to users within the boundaries of a single MCF domain, the challenge of a hassle-free, reliable MM service that spawns across multiple UPN communities (i.e. multiple MCF domains) is tackled here.
In the following we will introduce UDMM, a unified MM system for UCNs that is based on P2P mechanisms, PMIPv6 and the MCF. Before going into the technical details we will first illustrate why a P2P approach was chosen over other solutions, present related papers from the area of P2P based MM, and conclude with a description of the specific use cases.

4.1 Distributed Mobility Management with P2P

Along the bibliography related to distributed mobility management we may find a great diversity of contributions. For instance, in [7, 8] authors propose a session establishment approach, providing seamless connectivity for the MN while moving along different LMDs. Since both are based on PMIPv6, network entities such as LMAs and MAGs handle mobility on behalf of MNs. The main idea behind these drafts is to establish a session between the MN and the first reached LMA. For example, in [8] the first LMA becomes the Session Mobility Anchor (SMA) for this MN. Then, when the MN moves to another domain, the visited LMA and the SMA have to setup a tunnel between them, aiming to keep the roaming transparent from outside these domains. What is proposed here is to replicate the PMIPv6 tunnelling based performance to manage inter-domain mobility, but held by different entities: LMA and SMA instead of MAG and LMA. In these interesting works the scalability problem is thus avoided since tunnelling facilitates the connectivity between different domains. However, proper session maintenance plays an important role against UCN requirements in terms of flexibility. The intrinsic features of the UCN, in terms of the random availability of devices, do not allow us to rely on the uninterrupted performance of any device.

On the other hand, there is a trend within the related work that plays an important role in our solution. In [9], for example, the authors propose a translation from Mobile IP (MIP) to an innovative architecture based on a peer-to-peer overlay, with the aim to overcome all those inconveniences of the centralised systems, previously mentioned. One of the key points here is how they break the fixed and inflexible architecture of home networks and foreign networks by using a P2P overlay. All the entities, here called Mobility Agents (MA), start to implement both functionalities, assuming not only home agent responsibilities but also foreign agent ones depending on which mobile node (MN) they are managing. In this work they use three MAs to track this movement. The key point here is the virtualisation of the MM performance to distribute the system using the features provided by P2P techniques. With them, specifically by using the Data Hash Table (DHT) properties, they assign automatically the responsibilities between the MAs to manage a certain node, increasing the level of abstraction to a virtual one rather than a physical. Other work pointing to P2P networking as a basis for MM is described in [10]. Authors provide here a MM solution based on grouping the Home Agents onto P2P networks. Then they let the network operators create their own P2P domains, forming communities. Moreover, users’ mobile phone numbers are translated by using the Domain Name Service (DNS), with the purpose of identifying MNs in the P2P network.
with a unique identifier. Once the HA joins, the P2P overlay may accept users. These users are usually selected according to proximity criteria, in order to reduce the registration updates while moving. Then necessary user’s information, as the binding address, will be accessible by the HAs thanks to the P2P overlay. Thus, in [9] and [10] we have two examples of how useful the P2P mechanisms can be for distributing the MM protocol, providing a solution where the well known weaknesses of centralised systems; i.e., single point of failure, bottlenecks or low scalability, are overcome.

In summary, the DHT features provide us, on the one hand with the automatic distribution of the mobility context information and, on the other, with the proper abstraction level to make it feasible. It should be noted that the discussed approaches are based on the MIP-related standard instead of PMIPv6. However, the P2P concept can be transformed to be used with PMIPv6, as we will show in the following. Using a P2P based architecture has the following advantages:

- Decoupling of MM infrastructure entities
- Accessibility of mobility context from arbitrary MM nodes in order to allow for non-operator-provided MM infrastructure
- Self-management capabilities for user-provided networks
- Resilient infrastructure that can cope with parts of the MM infrastructure while nodes appearing/disappearing
- No need to rely on operator MM service

Applying P2P networking in the solutions however has some inconveniences as well. For instance:

- Messaging overhead, as P2P information, needs to be exchanged among the participating MM entities.
- Delay for queries, MNs can experience a longer delay when the DHT table is queried upon association with MAG.

In the paper at hand we have chosen an approach based on the well known Kademlia protocol, which was introduced in [11]. It is a structured P2P overlay based on DHT, where the nodes, here peers, are distributed among a virtual overlay regardless of the physical location. The virtual shape of the network depends directly on the implemented protocol, nevertheless all of DHT based protocols have in common the same principles:

- We have an N-bits virtual space with \(2^N\) addresses that may point both nodes or data objects.
- A hash function maps unique identifiers (UID) into N-bit keys, so that any identifier corresponds to a different address.
- After the hashing process of a node’s UID, each one is allocated into the virtual overlay and it receives an amount of keys to be responsible for in case they are fulfilled by data objects. The number of keys one receives may vary and depends directly on the number of nodes there are in the network to divide the entire space and their own keys as well.
This particular manner of distributing the nodes and the information among them ensures that every search returns successfully the requested object.

Since the persistence of the data is ensured by the P2P protocols even when nodes leave the network, the entities performing MM relay on the P2P layer to manage the necessary information. In our solution we propose to promote only the MAGs as peers because they are the MM entities who are closer to the MNs and therefore they have the current information per node on real-time. To achieve this functionality the mobility management layer makes use of the API provided by the Kademlia implementation, as shown in Fig. 4. Then by using this API every MAG may perform the following actions in a transparent manner: join, leave, store, update and remove. According to the use of a P2P overlay there is still an issue left. This aspect is related to the stored information. Intuitively, if we wonder what information we would need to manage the movement of a mobile object, the position sounds as a reasonable answer. Applying the logic we can obtain the answer for our specific case. It is important to know the position of the MN to manage it while moving. However here, in a world controlled by the IP rules, the physical one is not as important as the IP addresses of the entities that keep the MN attached to the network. Hence, each MN will be represented in the system by an information unit containing the IPs of its current MAP and MAG, as well as the MCF domain where they are located. Finally, this information unit will be stored with the key obtained by hashing a unique identifier representing the MN, in this case the MAC address.

In the following section we will show how the MM layer and the P2P layer cooperate to manage MNs roaming among communities.

4.2 UDMM: User-Centric and Distributed Mobility Management

In UDMM we distinguish between two operation modes. The first one is called intra-domain mode and is related to the MCF performance explained in Sec. 3. The second one is referred to as the inter-domain mode and distributes the first mode performance among communities by using the P2P overlay. As a design decision the MAGs are the only entities with P2P capabilities, as explained earlier. Since MAGs have a new role in this architecture, they become the key-entity here, where both operation modes converge. In Fig. 5 we may observe the system performance when a node is being managed while moving. In the upper side of the figure we have arrows representing intra-domain operations, while in the lower one the arrows indicate the inter-domain mode:

1. The MN reaches the network (both diagrams).
2. The MAG sends a request for context information about this concrete node. The result may be either unsuccessful or successful search. As the node is new in the MM system the result has been the first one, which means no context information (inter-domain).
3. Starting from scratch: MAG sends a MAP request to the MCF (intra-domain).
4. The MCF answers with the selected MAP (intra-domain).

5. MAG and selected MAP establish the binding with the messages PBU and PBA introduced in the PMIPv6 protocol (intra-domain).

6. Then the MAG creates/updates the information related to this MN in the DHT. The information fields may be extended but so far they are current MAG, MAP and MCF-domain (inter-domain).

7. The MN moves to other domain (both diagrams).

8. It tries to attach to a new MAG there (both diagrams).

9. The new MAG queries the DHT for the context information. In this case the answer is successful search. Some time later (represented by the label 9'), the old MAG sends another request, querying for the MN context information. When there is no change in the MAG field of the resulting mobility context information (it is still its own identifier), it removes the entry for the MN from the DHT, assuming that this MN has left the MM system. Otherwise it does not need to do anything because this MN now is under the management area of another MAG. With this mechanism we ensure that the system is only storing the mobility context information about the nodes that are active in the global MM domain.
10. In order to maintain the session, the MAG establishes a binding with the previous MAP, by using again PBU and PBA\textsuperscript{3} messages (inter-domain).
11. Afterwards the MAG sends another request to the MCF for a possible MAP placed in their domain (intra-domain). This request may be sent in some other cases as well. For example when the MAG notices that the current MAP has disappeared (intra-domain).
12. Where appropriate the MCF answers with another MAP, which brings us to the 5th step (intra-domain).

\textbf{Fig. 5.} System performance split in abstraction layers.

\textsuperscript{3} Denotes standard PMIPv6 Proxy Binding Update and Acknowledgement messages
5 Conclusion

In this paper we have presented a MM solution to meet the specific requirements of UCNs. The MCF provides a brokering entity for the user provided MAPs and applies a simple loadbalancing mechanism. A prototype has been implemented and tested under lab conditions, however an extensive field testing remains to be done within the future work. As an ongoing work we are also preparing an Internet Draft that proposes the dynamic mobility anchor coordination presented in this paper, using PMIPv6 and the Runtime LMA assignment as a basis for our efforts.

The extension of the MCF-based mobility management domain with the help of P2P technology enables a simple to use infrastructure that interconnects the different ULOOP communities. A prototype has been implemented for this part as well, further work includes testing QoS parameters and analyzing the delay in establishing new mobility bindings as well as the examination of inter-domain roaming performance.

References