# A Framework for Mobile Applications based on a structured P2P Overlay

Mario Kolberg, Michael Wilson, Martin Blunn, Evan Magill, Peter Burtwistle

University of Stirling
Dept. Computing Science and Mathematics
Stirling FK9 4LA, UK
{mko,mbl,ehm}@cs.stir.ac.uk

Sysnet
Hillington Park Innovation Centre
Glasgow, G52 4RU, UK
{mw,ptb}@sysnet.co.uk

## Abstract

*P2P applications are increasingly popular. Unlike traditional client-server applications they do not require central server resources. This makes them less costly to host and maintain. While there have been a large number of P2P applications been developed for PC based systems, there are only very few for mobile appliances, such as mobile phones or networked PDAs, and even fewer which operate successfully on the public mobile data network (GPRS, 3G), rather than WiFi. This is largely due to the difficulties posed by the restrictive data access to mobile devices (NATs, Firewalls by the network operators) and traditionally high charges for data communications. Recently, the latter point has been virtually removed by the introduction of flat-rate data packages by the operators. This paper (and demo) present a framework and applications based on a structured P2P overlay network which can successfully operate on the public data network.*

## 1. Introduction

Peer-to-Peer (P2P) Overlay networks are becoming increasingly popular. More recently this also extends to Distributed Hash Table (DHT) based structured P2P overlay networks. The main advantage of P2P overlay networks when compared with client-server based systems is that they do not require central server components. P2P systems are self-managing, in that they can cope with nodes leaving and joining even at a high churn rate. Moreover P2P systems are very scalable and can cope with a large number of nodes. Typically P2P systems operate as an overlay network on top of the IP layer.

In a DHT based overlay network, each node is assigned a unique node ID. This is typically generated encoding their IP address with a secure hash function, such as SHA1. Likewise, data to be stored is assigned a file ID. Again this is generated applying a hash function on the data name or similar keyword. Each node stores data who's ID falls in a certain section of the overall ID space.

However, there are very few implementations of P2P overlays for mobile handsets. There are virtually no applications yet which can be used on mobile phones while connected to the public data network. Reasons for this are discussed in Section 2 below.

This paper discusses a prototype which is based on the Kademlia DHT implementation for Nokia S60V3 mobile handsets [1]. On top of the Kademlia layer we have implemented NAT/Firewall traversal functionality. Together, the Kademlia implementation and the Nat/Firewall traversal functionality is referred to as the Framework. Applications are developed on top of the Framework accessing framework functionality via an API.

Below, Section 2 investigates the issues with deploying a DHT-based P2P overlay on mobile handsets connected to the public data network. Section 3 describes the Kademlia overlay briefly, and Section 4 describes the demo application. Section 5 concludes this paper.

## 2. Issues with deploying P2P overlay network applications on mobile handsets

In order to maintain the overlay structure, P2P overlay networks need a constant data link setup by all nodes to exchange updates of nodes joining and leaving the network. Until very recently data communication from mobile handsets was expensive. This is less of an issue now as data communication has been packaged in mobile phone deals and data flat rates are on offer. However, each operator network is secured by a Firewall and/or NAT (Network Address Translator). This makes it very difficult to connect from one handset to another.

Furthermore, usually IP communications on mobile phones are restricted to outgoing connections only. This has mainly two reasons: firstly, there are not sufficient IPv4 addresses available, so that every mobile phone could be assigned one. To avoid this issues operators use dynamic IP address assignment and NATs (Network Address Translators). So every time a mobile phone acquires an IP address this may be different from before. Hence IP addresses are not an ideal choice to use when creating the node ID. The second issue is related to billing and security. If a connection arrives at the mobile phone, this traffic consumes airtime which needs to be paid for,

typically by the owner of that handset. If the user is unaware of incoming traffic, they could have a nasty surprise at the end of the month. Operators prevent incoming traffic by firewalls. Incoming traffic is only allowed in response to requests made by the phone.
These issues are addressed by the framework's NAT/Firewall traversal functionality and these will be demonstrated in the demo.

## 3. The Kademlia P2P Overlay

Kademlia [2] is DHT based structured overlay. For this project, Kademlia was chosen because of its relatively light-weight algorithm. Furthermore Kademlia has been used in a number of other (PC based) implementations and used extensively for file-sharing applications in conjunction with BitTorrent.
Routing tables in Kademlia nodes consist of a number of lists, or buckets. Each entry in these lists contains information to locate the corresponding node (typically IP address). Each bucket contains a maximum of node entries k (typically 20). As the Kademlia algorithm is based on "distance" between two nodes (in terms of difference between two node IDs) there is a list for each bit in the node ID (i.e. with a 128 bit node ID there will be 128 lists). More precisely, the distance between two node IDs is the XOR of both node IDs. Hence each bucket contains nodes with a certain "distance" to the current node.
Nodes in the $n^{th}$ list differ in the $n^{th}$ bit from the current node's id, all bits < n must match with the ones from the current node. Consequently, the first bucket contains 1/2 of the nodes in the network (long distance). The next bucket only contains 1/4 of the nodes in the overlay, and the next 1/8 of the nodes. Hence, buckets with nodes close to the current node will have fewer entries than buckets with nodes further away. Furthermore, buckets with nodes close by will fully map all nodes in that section of the network which are known to the current node, whereas buckets with nodes further away will only contain k entries. Hence the network near by will be better known than sections of the network far away.
Kademlia defines four different messages: Ping, Store, Find_Node, Find_Value. Ping is used to check if a node is alive. Store inserts a key value pair in the network, by storing it on one node. A node which receives Find_Node responds with a list of k nodes from its routing table which are the closest to key. Find_Value is very similar to Find_Node, but if the recipient of the message actually stores the looked for value, it will return this.

## 4. Mobile P2P Applications

As mentioned above, applications contain two parts: the generic part consisting of the Kademlia implementation plus NAT/Firewall traversal functionality, and the application specific code with the API interface linking the two parts. This design is shown in Figure 1.

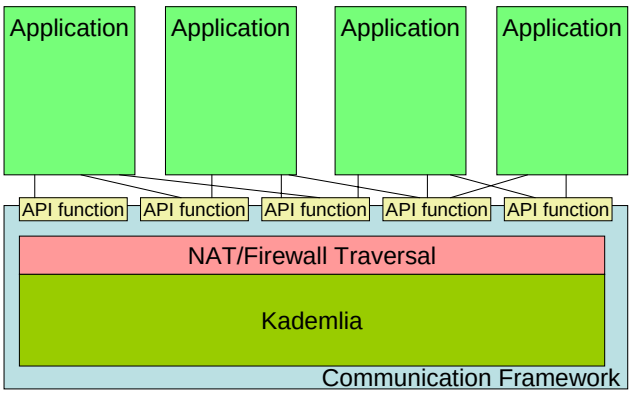


Figure 1: Software Architecture of Applications

The Demo will show applications developed using this architecture. Any data used by the applications will be stored in the P2P overlay network, i.e. on mobile nodes. No central data server will be used. The applications are typical group applications and are emphasizing the sharing of data. For instance, one application to be shown will indicate the location of different subscribers. Each node will learn its location through a GPS link, and will insert this data into the network together with a current status. Status can include a variety of messages, such as 'Available', 'Busy', or 'Do not Disturb', but might also indicate more details on the activities of people, such as 'At work', 'In a meeting', 'Teaching', 'Going home', 'With customer' etc. Additional data may also be saved in the network, e.g. photographs or voice recordings. These can also be tagged with location information (Geo-Tagging). A feature included, is the 'I'm lost' functionality. Here a person takes a photo of a landmark and sends this together with a message 'I'm lost' to other subscribers. Recipients might recognize the location in the photo and be able to give directions.

## 5. Conclusion

This paper illustrates the technology behind mobile P2P based applications. These applications share a common DHT and can communicate via the public data network. This is novel as previous applications and DHTs deployed on mobile handsets require WiFi links to be operational. The demo will show a group application on S60 V3 mobile handsets using the communication framework. No further equipment is required.

## 6. References

[1] I. Kelényi, B. Forstner. Distributed Hash Table on Mobile Phones, 5th IEEE Consumer Communications & Networking Conference (CCNC 2008), 2008
[2] P. Maymounkov and D. Mazieres. Kademlia: A peer-to-peer information system based on the XOR metric, Proceedings of IPTPS 2002, Cambridge, USA.