Peer-to-Peer Solutions for Cellular Networks

Andreas Berl and Hermann de Meer
Computer Networks and Communications
University of Passau, Germany
Email: {berl,demeer}@uni-passau.de

Tobias Hoßfeld
Department of Distributed Systems
University of Würzburg, Germany
Email: hossfeld@informatik.uni-wuerzburg.de

Abstract

The participation of stationary computers with high-bandwidth links in peer-to-peer content-distribution networks is highly popular. Mobile devices (e.g. cell phones), however, could not yet be launched into the field to a satisfactory extent. This paper discusses mobile peer-to-peer approaches that cover this issue and compares two promising approaches in detail. The first approach supports mobile devices by adding new infrastructure elements to the mobile network operator's domain. In the second approach, voluntary peers provide support for mobile devices. Both approaches are able to foster the integration of mobile devices into peer-to-peer networks with a large user community.

1. Introduction

Peer-to-peer (P2P) content-distribution applications are widely spread on stationary computers in fixed networks today. But also *mobile devices* (MDs) in cellular radio networks, e.g. phones or personal digital assistants, are potential candidates for participating in these popular P2P networks. Their integration into popular P2P content-distribution networks (CDNs) with large user communities is a special instance of the *mobile peer-to-peer* (mobile P2P) research field.

Although current MDs mostly have enough hardware resources to join such P2P networks, they are barely able to benefit from them. Their participation together with a large number of stationary computers with high-bandwidth links, increases heterogeneity within P2P networks, which causes performance problems, especially for MDs, cf. [1]. In contrast to this heterogeneity, P2P systems apply homogeneous logical structures (the overlay), in which computers and links are considered to be equal, having similar properties and capabilities.

Several mobile P2P approaches have been suggested, focussing on cellular radio networks. Although some of them might be applicable to popular CDNs, MDs are still not widely integrated in them. These approaches are categorized and their shortcomings are discussed in Section 2.

In addition, two approaches towards an integration of mobile peers into popular P2P-based CDNs are presented in detail and compared to each other. First, Section 3 presents a mobile P2P architecture, in which the *Mobile Network Operator* (MNO) supports MDs in his domain by integrating additional infrastructure elements. The MNO aims at improving the customer's experience with the P2P-based CDN, while maintaining control over the network, e.g., in order to reduce expensive inter-domain traffic. Second, Section 4 describes a mobile P2P architecture, in which MDs are supported by specialized peers, enabling them to efficiently profit from the P2P network. MDs provide mobile services in return to compensate for this support. Section 5 compares the two approaches and Section 6 concludes this paper.

2. Mobile P2P Solutions

Mobile P2P approaches that focus on cellular radio networks are partitioned in three categories with respect to the support they provide to MDs: 1) no additional support, 2) support from the P2P protocol, and 3) other support not being part of the P2P protocol.

No support: Solutions of this category are reshaping P2P client software to the requirements of MDs, considering the MDs limitations, whereas the P2P protocol remains unchanged. MDs join P2P networks as ordinary peers without getting further support. In this paper, this approach is called the *straight-forward approach* or *no support solution*. Symella (http://symella.aut.bme.hu), e.g., is such a P2P client for the Gnutella [2] P2P file-sharing network. The

MD is able to download files, but upload of files is not supported. Another example of this category is Mopiphant [3], a P2P client for the eDonkey file-sharing network which is implemented for MDs.

If no further support is provided, MDs find themselves in severe competitive situations with stationary computers. Often hundreds of peers are requesting popular downloads concurrently. A peer which is providing such a download, distributes its upload bandwidth among the requesting peers. A certain number of peers is served instantly, other peers have to wait in queues to be served later. Peers in queues are in competition with each other, wanting to be served as soon as possible. MDs however, do not perform well in these competitions: *1)* Among peers competing for content usually those are preferred that provide content in return (tit-for-tat principle). Due to hardware limitations, restrictions of the wireless link, and short on-line times it is not possible for MDs to provide an equal quantity of content (or equal upload performance) as stationary computers do. *2)* MDs can manage significantly less simultaneous TCP connections than stationary computers [4], therefore they are not able to queue themselves in many queues simultaneously. Stationary computers are often waiting for content in up to hundreds of queues to increase the probability of being served. *3)* MDs in cellular-radio networks are often hidden behind firewalls. In this case, other peers are not able to establish direct communication with the MDs. This often leads to penalties for such firewalled peers within P2P networks, e.g. in eDonkey. *4)* If MDs go voluntarily or involuntarily off-line, e.g. because of dead spots or low battery charge, they are likely to be deleted from queues and have to restart their waiting periods again. Therefore, MDs have to wait much longer time periods for downloads in P2P CDNs, than stationary computers. During these increased download times the ongoing P2P signalling traffic prevents MDs from changing into dormant mode, which heavily affects their battery lifetime.

Support from P2P Protocols: Solutions of this category support MDs in P2P networks and the support is provided by the P2P protocol itself. To achieve this kind of support, either all peers or some specialized peers of a P2P network have to assist MDs. Peers are often determined to support MDs, because of having certain properties (e.g. high-bandwidth links). An example for this solution category is the *hybrid chord protocol* [5]. It modifies the well known chord protocol [6] to cope more efficiently with effects of mobility. Peers are divided into *static* nodes and *temporary* nodes. Temporary nodes (nodes with short on-line times), are relieved from storing object references, improving the overall network performance. The authors of [7] propose a distributed mobility management mechanism which is based on hierarchical DHTs. The mechanism differentiates between stable and unstable peers in order to handle peer mobility. Information about resource locations is stored on stable peers only. The optimal split between stable and unstable peers is further investigated in [8] and [9]. Other approaches suggest P2P networks, in which specialized peers are determined to support MDs by aggregating or filtering data for them. In these solutions, MDs are partly or entirely relieved from network maintenance and routing tasks. In [10] *proxy servers* are used to integrate MDs into a P2P architecture. In [11] *surrogate peers* support MDs and the JXME[1] project defines *relay peers* to connect mobile peers to the JXTA[2] P2P environment (nowadays MDs are also able to participate *proxy-less* in JXTA). Such solutions are difficult to apply to popular P2P networks, which already have large user communities. It is hard to convince an existing community to accept protocol modifications or newly designed protocols, especially when peers are forced to provide additional support for MDs.

Other Support: Solutions of this category support MDs without modifying P2P protocols of large user communities. Instead, either ”voluntary” peers within the network are changing or extending their protocols to provide support for MDs, or support is provided from outside the P2P network. An example for this solution category is *MobileMule* [12]. This is a project in which users support their own MD by a second, fully featured computer which has access to the P2P network. However, in this approach MDs do not really profit from the P2P network. MDs just remotely control the second computer, not being able to download or share any content at their current location. This solution category has not achieved a widespread integration of MDs into popular P2P-based CDNs, yet. However, it seems to be a promising category to achieve this goal.

In the following sections, two solutions of this category are discussed. An MNO-based approach introduces additional infrastructure elements to the P2P network in Section 3, while the peer-based approach relies on voluntary stationary peers that consume mobile services, provided by MDs, see Section 4.

3. MNO-Supported Architecture

The desire of MNOs is to add value to the P2P data flows and to turn them into services they can charge

1. http://jxme.jxta.org
2. http://www.jxta.org

for. However, this includes to preserve the basic P2P user experience and connectivity, while maintaining control over the network and the ability to charge for provided services. Furthermore, operators would like to keep traffic in their own domain to avoid costs due to inter-domain traffic, an issue which is currently discussed in the ALTO IETF group.

Oberender et. al [13] have suggested an architecture that enhances the eDonkey network by three parts that are located within the MNO's domain: An *enhanced index server*, a *cache peer* and a *crawling peer*, as shown in Figure 1. All proposed P2P components offer a value-added service.

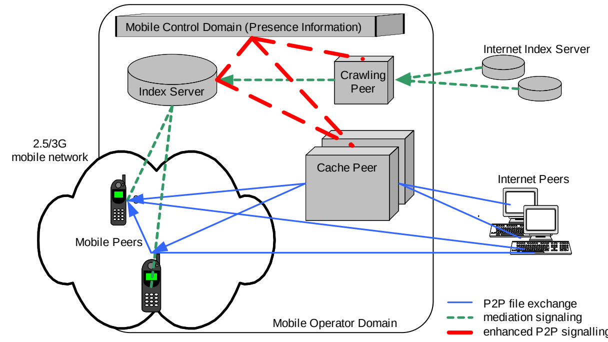

Figure 1. MNO-supported architecture

The **cache peer** is a specialized peer that stores popular files at the network core to reduce the amount of expensive air-interface usage. The peer cache owes its name to the fact that it is implemented as an ordinary peer that interfaces with the mobile domain controller and the index server. These elements negotiate which resources should be stored at the network-core. The cache peer receives information from the index server. It uses the list of popular resources to adopt its caching strategy and decide whether to fetch or to drop a cached resource. If the access characteristic measured at the index servers signals multiple downloads of a popular file, caching is initiated. For downloading of files, the cache peer uses the same mechanism as arbitrary peers. As such, the completion of chunks is signaled to the index server, which informs the requesting peers. This signalling is very important as it prevents any new downloads from mobile peers, instead the resource is shared from the cache peer only. Peers that join later will only receive a single source: the cache peer. As a consequence, the traffic is kept locally within the MNO's domain and the download performance is improved significantly [14].

Short churn times of mobile peers, representing high mobility, degrade the service performance strongly. This effect is reduced if a cache peer is used, cf. [14]. The application of the cache peer adds a new characteristic to the system: If a cache peer is applied, the downlink of the downloading peer is the limiting factor. If the cache peer is not applied (or files are not cached), then the uplink of the providing peer is the bottleneck.

The eDonkey protocol belongs to the hybrid class of P2P systems using weakly centralized resource mediation, which is provided by several index servers. The index servers provides two essential services: name search and source request. In name search a peer asks for all resources that match a given string. Secondly, when peers start downloading a certain resource, they ask for peers that currently share this resource. It is recommended that a single index server administrates all resources known inside the mobile domain. Thus, popular resources can be identified and then caching can be initiated. Two extensions of the **enhanced index server** can deliver enough information to bring the caching mechanism in place. First, source requests are logged by their resource ID. All peers that are connected to this index server, frequently request any new source that has been discovered lately. In reverse, from these requests a list of peers is created that are currently downloading a file. Second, the response messages of resource requests are altered. If the cache peer is contained in the result, all other sources are deleted.

The eDonkey community offers a large variety of resources. Unmodified eDonkey peers acquire resource mediation decentrally. If the primary index server does not return enough query hits, the software automatically connects to other available index servers. For the mobile context this behavior is undesirable, since the mobile domain index server cannot keep track of popular files. Besides, other index servers cannot distinguish cache peers and therefore cannot hide other sources. To maximize the benefits of the modified eDonkey architecture, mobile peers must connect to one of the enhanced index servers. The **crawling peer** is used for coordination between index servers of the mobile domain and index servers in the Internet. The index server requests unknown resources from the crawling peer, which fetches mediation data from Internet index servers. Thus, any resource available inside the global eDonkey community can be located and accessed. In [15], it is shown that the crawling peer is a very efficient solution to realize resource mediation in P2P file-sharing networks and that it distributes the load in the network among different index servers. Thus, it is possible to accomplish flash crowd arrivals of search requests without loosing the quality of the service.

The MNO which supports the P2P file-sharing service can dimension the cache peer, the index server, and the crawling peer in such a way that the experienced quality of the file-sharing service (e.g. in terms of short download and search times, or successfully answered search requests) satisfies the user.

4. Peer-Supported Architecture

Most current MDs are able to process complex JAVA software, play music, or show videos. Some are able to receive TV programs or radio stations. Sometimes MDs have integrated video cameras, GPS modules or thermal sensors. Besides the common telephone service, MDs are able to send SMS (Short Message Service) or MMS (Multimedia Messaging Service) messages, facsimiles, or Emails. Due to a unique identifier, MDs are reliably authenticated by operators. Therefore, MDs can be located, for instance, or payment/micropayment can be done by calling special service numbers [16]. Some of these features and services of MDs are not (or barely/expensively) available to stationary computers in fixed networks and can be provided to stationary computers by MDs, using their JAVA environment. These *mobile services* are able to turn MDs into valuable trading partners within P2P networks.

In [17] *partnership schemes* between stationary computers in P2P networks and MDs are suggested that define the cooperation of stationary computers in P2P networks with MDs in cellular-radio networks. Two partnership schemes are discussed in detail in [17], an SMS-based scheme and an advertisement-based scheme: A stationary peer of P2P-based CDN supports MDs by processing downloads on their behalf. To be able to do this, the stationary peer has to extend its P2P software (other peers in the original P2P network do not have to change their P2P software). This stationary peer is called *extended peer*. MDs use specialized software to communicate with extended peers and are not part of the original P2P network. An MD schedules a download job on an extended peer and goes off-line to save energy. When the extended peer has finished the download job, the data is transferred (with highest possible throughput) to the MD. MDs are completely relieved from the costly competition for resources, because they are not part of the original P2P network. Additionally they get support in the energy-efficient consumption of resources, because they are enabled to stay off-line while the extended peer processes the job and they receive the requested data. MDs compensate for this support by providing mobile services to extended peers. While the requested data is transferred to the MD, additionally SMS messages or advertisements are transferred to the MD. If MDs receive SMS messages and phone numbers from the extended peer, they send the message to a required number. If MDs receive advertisements from extended peers, they display them to the user (e.g. pictures, banners, or small videos). Many other partnership schemes are possible (e.g. based on MMS services or micropayment services) and can be implemented by using currently available technologies (e.g. cell phones in GPRS networks).

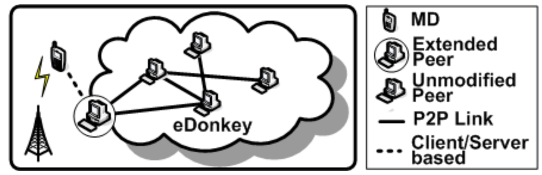

Figure 2. Peer-supported architecture

Figure 2 illustrates the suggested mobile P2P architecture. It can be observed that a P2P network is enclosed as a component. An extended peer is shown, which is able to communicate within the P2P network as well, as with MDs. MDs are not directly integrated in the P2P network, they communicate neither with unmodified peers nor with other MDs. Instead, they are separated from the P2P network. This simple structure of the mobile P2P architecture enables an easy establishment within existing P2P networks. The extension of a single peer is sufficient to instantly enable participation of MDs. The number of supported MDs is solely limited by the performance and the configuration of the extended peer. The scalability of the enclosed eDonkey network is not influenced by the proposed P2P architecture, because the newly added elements (extended peers and MDs) are not visible to the network. Extended peers appear to be (very active) ordinary peers to the eDonkey network.

Specialized communication and application protocols are used to explicitly support wireless links of MDs. Standard P2P protocols are usually not considering varying delay or bandwidth which are common for MDs. Also, the existence of dead spots and IP changes of MDs is usually not considered (cf. Section 2). Improved transport protocols for wireless communications are discussed e.g. in [19]. The application layer protocol uses compression of data and data resuming, as it is done in the *File Transfer Protocol* [20]. During communication, extended peers identify MDs by a pseudo-unique ID (chosen by the extended peer), to be resistent against IP address changes of MDs.

The proposed mobile P2P architecture was proto-

Table 1. Comparison of the mobile P2P architectures

Challenge	MNO-supported Mobile P2P	Peer-supported Mobile P2P
General properties		
Support is provided by	MNO's infrastructure elements	Voluntary (fixed) peer in the P2P network
Separation of MDs from P2P network	Moderate integration in a separated subset of the P2P network	MDs are fully separated from the P2P network
Incentive	Traffic is kept in MNOs domain, utilization of MNO's infrastructure is raised	Consumption of mobile services
Mechanism	Cache peer, crawling peer, index server	Extended peer
Security and privacy	High - assured by MNO	Low - unknown peer
Implementation costs	High - change of MNO's infrastructure and implementation of mechanisms for supporting entities	Low - update of peer software
Deployability	Difficult - change of MNO infrastructure	Easy - update of peer software
Expenses of mobile users	None	According to offered mobile service
Support of MDs		
Computation power and memory	P2P related tasks are minimized for MDs	MDs are relieved from P2P related tasks
Battery consumption	MDs are not discriminated $\Rightarrow$ they have usual online-times	Online-times of MDs are minimized
Dormant mode	P2P activities can be paused to switch to dormant mode	Dormant modes can be activated during P2P downloads
Variable link quality, dead spots	The cache peer lessens such effects	Special communication protocols and client/server based communication
IP Address changes	Mobile IP can be efficiently used [18]	Pseudo unique ID
Short online-times of MDs (churn)	Leveraged by cache peer	Supported by extended peer
Tit-for-tat principle	Supported by cache peer and index server	Replaced by mobile services
Concurrent TCP connections	Index server returns only a few sources	Only single TCP connection needed
Firewalls and low id	Supported by index server	Client/server based communication

typed and evaluated in [17]. A typical cell phone (Sony Ericsson S700i) had to download a popular MP3 file from the eDonkey network: Without support, the MD never managed to download the file in less than 25 minutes. It had to spend most of its time by waiting for the download to begin, while being on-line and consuming energy. The supported MD, however, had an online-time of less than 10 minutes. Also the provision and consumption of mobile services (advertisement service and SMS text message service) were evaluated. An advertisement (a ".png" file) was displayed to the user during the MP3 transfer and an SMS text message service was applied to the s700i.

5. Comparison of the Architectures

Table 1 compares the two presented architectures with regard to 1) general properties (e.g. costs of their application or the related incentives) and 2) with regard to the support that they provide to MDs. The table lists challenges of the discussed research area, together with concrete solutions that are provided by the architectures.

The table shows clearly, that both architectures are able to successfully support MDs in their participation in popular mobile P2P-based CDNs. Shortcomings of MDs, as they were described in Section 2 are tackled sufficiently by both of the architectures and MDs are enabled to participate in P2P-based CDNs.

It can also be seen in the table that the peer-based mobile P2P architecture is the more flexible approach. The costs of its application are low for both, the provider and the consumer of support. The architecture is easy and fast to establish and the MDs are able to contribute to the P2P network according to their special abilities. However, trust and privacy issues are severe problems of this approach. The extended peer has to trust the MDs and vice versa, to enable the discussed partnership schemes. A download or an SMS might contain undesired content, there might be freeriders on both sides, and the partners are able to eavesdrop contents.

The MNO-supported architecture is the more reliable and secure solution. An official MNO is able to provide security, privacy, and reliability to the users of mobile devices. However, the complexity and costs of the solution are obstacles in introducing such a technology in real environments.

6. Conclusion

A comprehensive integration of mobile devices into widely spread P2P-based CDNs networks has not been achieved, yet. In this paper mobile P2P solutions have been categorized with regard to this research area. Furthermore, two mobile P2P architectures were described in the paper that provide a solution to the problem.

First, an MNO-supported architecture was presented, where mobile devices get support from their provider. The provider extends his infrastructure by special elements (cache peer, crawling peer, and index server) that exclusively support the mobile devices in his domain. Second, a peer-based architecture was presented, where mobile devices get support from voluntary peers within the P2P network. In this solution mobile devices are enabled to contribute mobile services to peers, according to their abilities (e.g. SMS based or advertisement based mobile services).

This paper has compared the two architectures and outlined their benefits and shortcomings.

7. Acknowledgments

This project was funded by Deutsche Forschungsgemeinschaft (DFG), no. ME 1703/4-2 and TR 257/22-1, and by EuroNF - Network of Excellence, no. 216366.

References

[1] B.-C. Seet, Ed., *Mobile Peer-to-Peer Computing for Next Generation Distributed Environments: Advancing Conceptual and Algorithmic Applications*. Information Science Reference, ISBN: 978-1-60566-715-7, 2009.

[2] M. Ripeanu, "Peer-to-peer architecture case study: Gnutella network," in *First International Conference on Peer-to-Peer*, Linköping, Sweden, 2001.

[3] "Homepage of the mopiphant project," http://www3.informatik.uni-wuerzburg.de/staff/mopi/.

[4] "Current mobile phones and features," https://developer.sprint.com/show_devices.do.

[5] S. Zöls, R. Schollmeier, W. Kellerer, and A. Tarlano, "The hybrid chord protocol: A peer-to-peer lookup service for context-aware mobile applications," in *International Conference on Networking*, 2005.

[6] I. Stoica, R. Morris, D. Karger, F. Kaashoek, and H. Balakrishnan, "Chord: A scalable peer-to-peer lookup service for internet applications," in *ACM SIGCOMM*, San Diego, USA, 2001.

[7] K. Park, T. Kwon, and Y. Choi, "SAMP: scalable application-layer mobility protocol," *IEEE Communications Magazine*, vol. 44, no. 6, 2006.

[8] S. Zoels, S. Schubert, W. Kellerer, and Z. Despotovic, "Hybrid DHT Design for Mobile Environments," in *International Workshop on Agents and Peer-to-Peer Computing*, Hakodate, Japan, 2006.

[9] S. Zoels, Z. Despotovic, and W. Kellerer, "Cost-based analysis of hierarchical dht design," in *Sixth IEEE International Conference on Peer-to-Peer Computing*, Cambridge, UK, 2006.

[10] H. Sumino, N. Ishikawa, and T. Kato, "Design and Implementation of P2P Protcol for Mobile Phones," in *3rd IEEE International Workshop on Mobile Peer-to-Peer Computing*, Pisa, Italy, Mar. 2006.

[11] T. Horozov, A. Grama, V. Vasudevan, and S. Landis, "Moby - a mobile peer-to-peer service and data network," in *International Conference on Parallel Processing*, Vancouver, Canada, 2002.

[12] "Homepage of the mobilemule project," http://mobil.emule-project.net.

[13] J. O. Oberender, F.-U. Andersen, H. de Meer, I. Dedinski, C. Kappler, A. Mäder, and K. Tutschku, "Enabling mobile peer-to-peer networking," ser. Lecture Notes in Computer Science, vol. 3427, 2005.

[14] T. Hoßfeld, K. Tutschku, F.-U. Andersen, H. de Meer, and J. Oberender, "Simulative performance evaluation of a mobile peer-to-peer file-sharing system," in *1st Euro-NGI Conference on Next Generation Internet Networks*, Rome, Italy, 2005.

[15] T. Hoßfeld, A. Mäder, K. Tutschku, and F.-U. Andersen, "Time-discrete analysis of the crawling strategy in an optimized mobile p2p architecture," ser. Lecture Notes in Computer Science, vol. 2429, 2006.

[16] "Infin micropayment," http://www.infin-online.de:2080/minis/mp/index.php.

[17] A. Berl and H. de Meer, "Integration of Mobile Devices into Popular Peer-to-Peer Networks," in *5th Euro-NGI Conference on Next Generation Internet Networks*, Aveiro, Portugal, 2009.

[18] M. Duelli, T. Hossfeld, and D. Staehle, "Impact of vertical handovers on cooperative content distribution systems," in *7th IEEE International Conference on Peer-to-Peer Computing*, Galway, Ireland, 2007.

[19] Y. Tian, K. Xu, and N. Ansari, "Tcp in wireless environments: Problems and solutions," *IEEE Communications Magazine*, vol. 43, no. 3, 2005.

[20] J. Postel and J. Reynolds, "File transfer protocol (ftp)," *IETF, RFC959*, October 1985. [Online]. Available: http://www.ietf.org/rfc/rfc959.txt