©2006 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works. The definitive version of this paper has been published in IEEE Wireless Communications and Networking Conference, 2006, 10.1109\/wcnc.2006.1696652.

Evaluation of a Pastry-based P2P Overlay for Supporting Vertical Handover

Tobias Hoßfeld, Simon Oechsner, Kurt Tutschku
Department of Distributed Systems
University of Würzburg, Germany

Frank-Uwe Andersen
Siemens AG,
Siemensdamm 62, Berlin, Germany

***Abstract*— Vertical handovers (VHO) are expected to be a key feature in Beyond 3G (B3G) networks. This paper presents and evaluates a Pastry-based P2P overlay network for supporting vertical handover in B3G networks. The P2P overlay is used to quickly locate attachment points (APs) for mobile entities and to retrieve rapidly the configuration and coverage information of these APs. The advantage of the P2P-based solution is its distributed nature, its scalability, and its self-organizing capability. The performance of the P2P-based VHO support architecture is evaluated in terms of the search time for APs, the scalability of the algorithm, and for homogenous and heterogeneous network layouts. In addition, the solution is compared with the conventional client-server approach.**

I. Introduction

For mobile communications beyond 3G, heterogeneous access systems are expected. A mobile terminal moving through a landscape and variety of access systems such as WLAN (802.11), WIMAX (802.16), 2.5G and 3G, and sometimes also fixed wireless access systems such as ISDN, DSL for fixed-mobile convergence, needs to perform *vertical handovers (VHO)* frequently, i.e. pass the ongoing connection from one access system to another, as well as from one operator to another. Access systems consist of numerous *attachment points (APs)*. An important task during a vertical handover is to locate quickly the appropriate access systems in range, respectively its APs.

In general, the execution of vertical handover can be accelerated if the involved entities are aware of the attachment points and their coverage areas. This approach saves considerable time which would otherwise be spent by the mobile or the attachment points by scanning the environment, e.g. channel frequencies or field strength, and to retrieve the configuration at the AP, e.g. used identification/authentication. The coverage areas of APs, however, might be not connected and highly volatile in their size due to radio wave propagation, interference, and system load. The identification of candidate APs is achieved by comparing the position, i.e. the geo-location, of the mobile device and the other attachment points. An entity controlling the execution of the VHO, e.g. a mobile or an AP, can query an *database* for AP candidates, their configurations, and their coverage areas.

Typically, such information will be stored in a central entity. Central database, however, are vulnerable to system failures and to overload situation. In particular, a central database is restricted if time-limited coverage area information, e.g. measurements of mobiles, needs to be stored and retrieved quickly.

In an earlier work, an architecture was proposed that replaces the central database by a distributed and self-organizing *peer-to-peer (P2P)* based mechanism [1], avoiding the disadvantages of centralized entities. The solution employs a modified Pastry P2P overlay network to distribute the database containing the configuration and coverage information. Besides solving the overload problem, the application of a self-organizing P2P mechanisms permits new ways of operating mobile communications systems. APs can be inserted easily with no or less manual interference, such as configuration tasks, due to the self-organization capability of the Pastry algorithm. Hence, access systems and APs of different technologies, and even of different operators, can easily be included into the system.

The rest of this paper is organized as follows. Section II shows the related work concerning vertical handover issues. The architecture concept based on P2P mechanisms and its advantages are briefly described in Section III and the numerical results of the performance evaluation are given in Section IV. In particular, the search time for AP and scalability of the algorithm is investigated. The P2P-based solution is compared with the classical alternative, a client-server approach. It will be shown that the new solution is competitive. Finally, the paper is concluded in Section V.

II. State of the Art

Vertical handover related issues address many different problems. However, the two main areas investigated by the scientific community are the one aimed at protocols to support the vertical handover phase and one addressing the resource management in order to fulfill the Quality of Service (QoS) requirements.

Balasubramaniam et al. [2] address the resource management. They propose dedicated decision process that guarantees the preservation QoS requirements. The decision process evaluates user location changes and also QoS levels of the current and target networks. The basic method of this approach is to use a rich set of context information usually available to exploit seamless computing.

Protocol mechanisms for handover support are investigated in [3]. The authors propose a full-featured suite to support both vertical and horizontal handovers. The proposed solution has is denoted as Full Stack Adaptation (FSA) and its functionalities are guaranteed by integrating the Mobile IP within the FSA.

Siebert, Lott et al. [4] describe a *Hybrid Information System* (HIS) that aids inter-system handover or vertical handover by creating a database where information gathered from the different radio access technologies is made available. In [1], this architecture is enhanced by distributing the database used by Siebert, Lott et al. In the next section, a short overview on the enhancements is given. A more complete description of the architecture is provided in [1].

III. Architecture Concept

In this section we will describe the basics of the P2P-based VHO support architecture and where the original Pastry [5] algorithm has been modified to achieve a better performance.

The architecture proposed by Siebert, Lott et al. [6] improves the speed and the reliability of VHO by exploiting radio measurements at the expected locations of the mobiles where the handover will take place. When a handover decision has to be made, a database is queried by the *Intelligent Service Control* (ISC), cf. [7], on radio measurements of the coverage area of candidate APs which might serve the location of the mobile. The input of the database query is the expected location of the mobile.

This database is filled with measurement reports like *received signal strength indicator*, *block error rate* or *received signal code power*. These measurements are associated with the locations

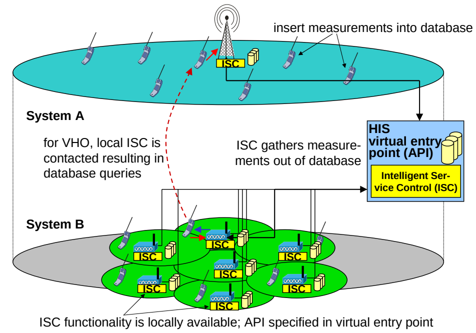

Fig. 1. Distributed information system: the database consisting of measurements is distributed among APs together with the ISC (enhanced from [7])

where they have been carried out. In that way, the database implements a map of the coverage area of the complete systems.

In [1] the architecture is proposed in detail, however, no performance evaluation is conducted. The basic concept is to distribute the database, along with the functionality of the ISC, by storing the radio measurements directly at the APs. The architecture that interconnects all of these partial databases is a modified Pastry [5] architecture. In this architecture, the peers are the APs. These peers form the overlay. An overview of the enhanced architecture is depicted in Figure 1.

The most far-reaching change of the original Pastry algorithm is the modification of the hash function. In our approach, geo-locations (represented as UTM coordinates) and access technologies are used to construct IDs and keys for the P2P overlay network. This is required in order to compare locations of mobiles, measurements, and peers (i.e. peers), respectively. This is different to the standard procedure in Pastry, where the hash function takes IP addresses or similar unique identifiers. Pastry uses prefix matching of the IDs for routing and determines in that way on which peer a measurement record is stored. As a result of the modified hash function, proximity is taken into account.

An AP has the two-dimensional coordinates (x, y) and uses a certain technology T (e.g. UMTS, WLAN, etc). The coordinates and technology of the AP are combined encoded in a Pastry ID. The location is represented as $2 \times n$ vector $(x, y) =_{\text{base } b} (x_1 x_2 \ldots x_n, y_1 y_2 \ldots y_n)$ to a base b; n determines the accuracy of the location representation. The technology of the AP is coded by a vector of m digits: $T =_{\text{base } b} T_1 T_2 \ldots T_m$ with b^m as the number of possible technologies.

We construct the ID of that peer in the overlay as $T_1 \ldots T_m x_1 y_1 \ldots x_n y_n$. We call the first part $T_1 T_2 \ldots T_m$ the *technology part* and the second part $x_1 y_1 \ldots x_n y_n$ the *location part* of the ID. In the performance evaluation, we used $b = 4$.

While these modifications change the basic Pastry algorithm in large parts, its main functionality is still to provide a lookup service, in this case for measurement reports. To retrieve information from the system, queries are entered through one of the peers. It is not important if the peer itself searches for the information or an entity attached to it. This query just has to contain a key that is constructed in the same way as described above, with the technology part chosen according to the type of measurement reports that are of interest and the location part coming from the coordinates of the mobile in question. Thus, the query will be routed to the AP of the queried technology that is closest to the mobile. This should be the peer storing the most important information for the mobile, or is at least only a few hops away from an AP fulfilling this condition.

To reach peers of a different technology quickly, we enhance the Pastry routing table with *shortcut links* for each technology differing from the local node's technology. These entries contain the addresses of the physically closest nodes of the desired technologies. To understand the advantages of this modification, we will look at an example. Imagine a mobile that is currently connected via UMTS nodeB X. For this mobile, handover to WLAN is considered. Therefore, a request is spawned at X that contains a key constructed of the location of the mobile and the code for the WLAN technology. Without shortcut links, this request would be routed to a WLAN access point that is potentially physically far away, since in Pastry, the first routing step only matches the first bit of the target ID. By using the shortcut link, the request is forwarded to an AP that is in the coverage area of X, and thereby much closer to the requested measurements. The same improvements by the shortcut links apply to the storage mechanism.

Since VHO are typically carried out within a small range, queries will enter the system near to the storage location of the searched information. The application of shortcut links obtains a low number of hops resulting in short response times.

In case the peer targeted by the search is not the one storing the relevant information, it is very likely that the responsible peer is contained in its leaf set, meaning just one more hop is needed. As a result, the algorithm keeps searches local, since only physically close peers are involved – in contrast to the original Pastry overlay, where search paths can include peers that are a large distance away from each other.

IV. Performance Evaluation

In this section, we describe the simulation scenarios and the results of the performance evaluation of the P2P-based VHO support architecture. The numerical results were taken from a proprietary time-discrete simulation implemented in Java.

A. Efficiency in terms of Search Time

First, we examined the search times that were needed to look up measurement reports. We constructed a huge homogeneous network layout in order to show the capability of the proposed algorithm. The considered network consists of 60,000 UMTS nodeBs and 40,000 WLAN access points, which were uniformly randomly dispersed in a 40,000 km^2 area, resulting in a mean distance of 800m between two nodeBs. The transmission delay between any two nodes is derived from their distance. All attachment points have a connection with a capacity of 1 MBit, and the size of a search message is 204 bits. This value is the same for all considered architectures.

Search requests were created for locations in a random distance from a randomly chosen AP. This simulates a mobile in a 'source' cell having a certain technology, like UMTS, for which a handover is considered to a 'target' cell with a different technology, like WLAN. More than 800,000 of these requests were spawned and the time it took to reach the node responsible for responding with the measurement report was recorded. For the central server scenario, the server which implements the database was positioned in the center of the area. All messages from the clients, i.e. the attachment points, were directly routed to the server with one hop delay. The server is assumed to be perfect, i.e., when the request arrives at the server, it is considered as being processed immediately.

Figure 2 depicts the CDF of the search time distribution for the classical client/server system (denoted in the following figures with label 'server'), for the original Pastry implementation (label 'Pastry'), and for the modified Pastry with (label 'with shortcut') and without the shortcut improvement (label 'w/o shortcut'). One can see that the server is performing better than the classical Pastry system. This is due to the fact that Pastry routes messages along possibly long paths, since the overlay does

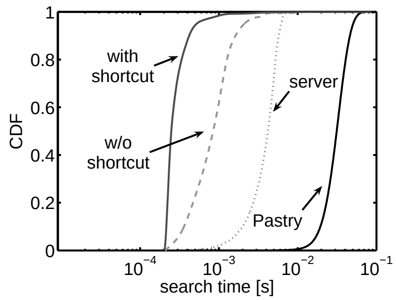

Fig. 2. CDF of the search time for a central server system, an unmodified Pastry, and our modified Pastry both with and without shortcut link.

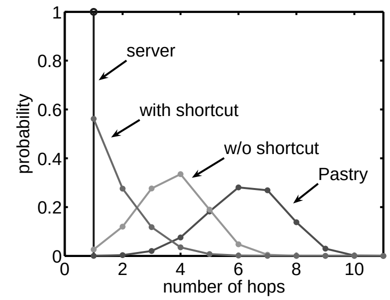

Fig. 3. Hop count distribution for a central server system, an unmodified Pastry, and our modified Pastry both with and without shortcut link.

not consider the physical network layout. However, if we introduce our locality-aware IDs, performance does not suffer from this disadvantage. Our approach even offers shorter search times than the server alternative. Moreover, the usage of the shortcut link speeds up the searches.

We can see the effect of the shortcut link even better in Figure 3 which depicts the hop count distribution. According to the simulation setup, the server scenario requires just one hop to deliver the information request. The modified Pastry overlay requires less hops to reach the destination than the original Pastry algorithm, regardless whether shortcuts are used or not. This is due to the fact that the modified Pastry algorithm utilizes the knowledge that requests are created close to their destination. The ID structure of the modified algorithm permits this feature. Hence, the request is routed to a peer that is also close in the ID space. In contrast, the unmodified Pastry algorithm routes the request to a peer that might be far away in the ID space resulting in a higher hop count, since only one more digit of the next hop is matched to the search key.

The shortcut again improves the result additionally, shortening searches to one or two hops in most cases. This means that due to the routing algorithm, the search message is routed through peers that are in the same vicinity in the ID space. With the structure of our ID, this translates to physically close peers. That proves that the search traffic is indeed kept local, which explains the high search speed.

B. Scalability

To show the scalability of our approach we simulated different scenario sizes, again for the four algorithm variants examined in the previous section. We varied the area in which we placed the attachment points while keeping the density and the proportion of UMTS nodeBs to WLAN acces points. The size of the area ranged from 100 km^2 to 40,000 km^2.

Figure 4 shows the mean search time for different network sizes. Although Pastry is scalable in the sense that it offers routing in O(logN) with N being the number of peers in the network, search times increase with larger scenarios. This can be explained with the fact that a larger scenario also means longer possible routes between two arbitrary peers. Since Pastry does not consider the physical network layout, longer routes are taken as often as short ones, leading to high search times in many cases. This may not be important for a normal file search, for which Pastry was originally designed for. However, this is critical for the considered application.

Since the distance from the edge of the simulated area to the centrally placed server increases, search times become higher also for this solution. A less significant increase can be seen for the modified Pastry without the shortcut improvement, since the larger scenarios allow for a higher hop count to reach the destination of a request.

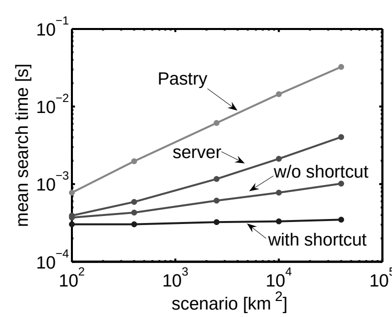

Fig. 4. Mean search time for different network sizes

The best behaviour is shown in Figure 4 by the modified Pastry algorithm using shortcut links. Due to the fact that these links cut off the larger part of the prefix-matching process of Pastry, the overall hop count is also kept low. Therefore, only a slight increase can be observed, which shows that this system is scalable with respect to our requirements.

C. Overhead vs. Performance

The selection of the Pastry specific parameters also impacts the performance of the proposed VHO support architecture. One of the parameters is the leaf set size of the Pastry routing table. A larger leaf set might lead to shorter search paths; however, this also introduces more overhead, since more information about nodes has to be distributed and kept up-to-date. To judge whether a change in the leaf set size has a significant impact on the performance of our system, we simulated 10,000 peers (6,000 nodeBs and 4,000 WLAN access points) with leaf set sizes of 2, 4 and 8. We did this both for the algorithm with shortcut links and without it.

Figure 5 shows the impact of the leaf set size on the search time. While small improvements can be observed for larger leaf sets in both architectures, the gain is not significant enough in both cases to warrant the overhead introduced by them. Especially with the shortcut enhancement, the cost of larger leaf sets outweighs the gain. Therefore, we consider a leaf set size of 4 as sufficient which is large enough to cope additionally with node failures.

D. Heterogeneous Network Layout

So far, we have only considered homogeneous networks, i.e., the radio access nodes were uniformly distributed in space. In reality, this assumption is not valid, since mobile networks comprise areas with a higher density of APs, e.g. large cities, and areas with a smaller number of APs, e.g. rural areas. The main difference between these two scenarios is the amount of traffic a single node has to handle. While the load is evenly distributed for the homogeneous scenario, the heterogeneous layout could lead to hotspots with a high number of requests. To make sure that our results are valid also in more realistic scenarios, we have to compare them to values retrieved from these scenarios.

To this end, we simulated a scenario based on the actual radio access node distribution in a large city and its surrounding area and compared it with a homogeneous network of the same size. However, requests still were distributed uniformly among the attachment points, since in a high-density area, cell sizes tend to be smaller and therefore the number of request per cell is about the same in every scenario.

In Figure 6, the CDFs for the maximum message throughput that a peer has to handle are compared for the two scenarios. Each pair of curves corresponds to a different load offered to the system.

While small differences can be seen especially for the scenarios with a high load, the general appearance of the curves is the same for each pair. Since the maximum message throughput is considered, these differences are small enough to show that there is no significant difference between the simulation of a homogeneous and a heterogeneous layout. Thus, the results obtained with a homogeneous network are still valid for realistic conclusions.

E. Carrier Grade P2P

In the case that the P2P system is applied in a carrier environment, the algorithm has to fulfill

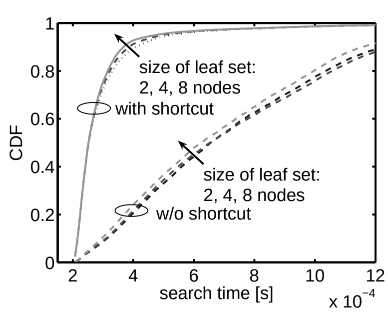

Fig. 5. Comparison of search time distributions for different sizes of the leaf set.

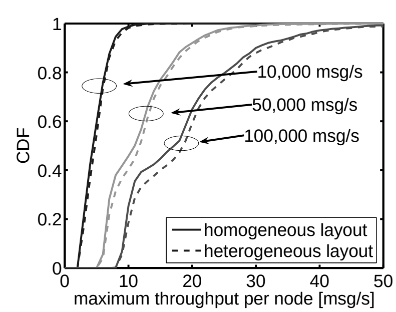

Fig. 6. CDF of maximum message throughput for homogeneous and heterogeneous network layouts.

certain quality requirements. The carrier grade objectives comprise fast searches while being scalable and keeping the load at a single peer low enough in order to prevent high costs for hardware and connectivity. The capability to locate quickly APs and their data was shown in Section IV-A and the scalability of the algorithm was investigated in Section IV-B.

Figure 7 shows the CDFs for the maximum message throughput per node for different load scenarios of a network consisting of 60,000 nodeBs and 40,000 WLAN access points. The load varies from 10,000 requests per second to 100,000 requests per second. Figure 7 shows that a tenfold increase in the average number of requests per second does not result in an explosion of the maximum message throughput per node. This means that the required message throughput is maintained on a low level even for high load scenarios. Hence, the proposed VHO support architecture the above stated carrier grade requirements.

V. Conclusion

In this work, we analyzed a Pastry-based P2P overlay architecture for supporting vertical handover in a Beyond 3G network. The P2P overlay is used to locate quickly attachment points for mobile entities and to retrieve coverage information of these APs. The performance of the VHO supporting architecture was evaluated in terms of search time for information, the required throughput, and the scalability of the algorithm. In addition, the solution was compared with the conventional client/server approach.

The investigations have revealed that the standard Pastry algorithm is not sufficient to meet the carrier grade requirements. The introduced modifications to the original Pastry algorithm, however, overcome these disadvantages and permit a fast search for information, which is the main criterion for its applicability in carrier networks. The improvements are achieved by exploiting the local nature of searches implemented by the modified hash function, together with the introduced shortcut links. As a further result, the load is evenly distributed among the peers.

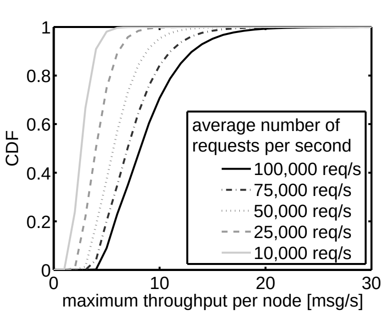

Fig. 7. Maximum number of messages per node in one second

Additionally, we investigated and demonstrated the scalability of the suggested architecture. Therefore, we considered scenarios with a homogeneous and heterogeneous network layout. The modified Pastry algorithm scaled well in both cases. In addition, we studied the effect of the Pastry leaf set size on the performance of the algorithm. It turned out that the size has little impact. This effect demonstrates the robustness of the algorithm against the selection of parameters.

Future work will include the investigation of the resilience of the architecture, i.e., its behaviour under churn which could be introduced e.g. by third-party WLAN access points.

Acknowlegment

The authors would like to thank Thomas Zinner for implementing the simulation and his sedulous effort in assisting the authors.

References

[1] Tobias Hoßfeld, Simon Oechsner, Kurt Tutschku, Frank-Uwe Andersen, and Luca Caviglione, "Supporting vertical handover by using a pastry peer-to-peer overlay network," in *Mobile Peer-to-Peer Computing MP2P'06*, Pisa, Italy, 3 2006.

[2] Indulska J. Balasubramaniam, S., "Vertical handover supporting pervasive computing in future wireless networks," *Computer Communication Journal, Special Issue on 4G/Future Wireless networks*, vol. 27/8, pp. 708–719, 2003.

[3] Y. Zhang G.G. Richard III S. Helal, C. Lee, "An architecture for wireless lan/wan integration," in *Proceedings of the IEEE Wireless Communication and Networking Conference (WCNC)*, 2000.

[4] Matthias Siebert, Matthias Lott, Marc Schinnenburg, and Stephan Göbbels, "Hybrid information system," in *Proceedings of IEEE Semiannual Vehicular Technology Conference (VTC2004-Spring)*, Milan, Italy, May 2004.

[5] Antony Rowstron and Peter Druschel, "Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems," *Lecture Notes in Computer Science*, vol. 2218, 2001.

[6] Matthias Siebert, Daniel Bültmann, and Matthias Lott, "Inter-system handover and coverage detection for 3g/wlan cooperation (invited)," in *Proceedings of 11th European Wireless Conference 2005*, Nicosia, Cyprus, Apr 2005, pp. 328–336.

[7] Matthias Siebert, Marc Schinnenburg, Matthias Lott, and Stephan Göbbels, "Location aided handover support for next generation system integration," in *The 5th European Wireless Conference, Mobile and Wireless Systems beyond 3G (EW2004)*, Barcelona, Spain, Feb 2004, pp. 195–202.