Content Availability and Signaling Overhead in DHT Systems for Mobile Environments

Stefan Zoels[1], Simon Schubert[1], Wolfgang Kellerer[2], Zoran Despotovic[2]
[1]Institute of Communication Networks, Munich University of Technology, Germany
[2]DoCoMo Communications Laboratories Europe, Munich, Germany
stefan.zoels@tum.de, corecode@fs.ei.tum.de, {kellerer, despotovic}@docomolab-euro.com

I. INTRODUCTION: MOBILE P2P COMPUTING AND DHTS

In recent years, Peer-to-Peer (P2P) traffic has become a major part of IP networks. A large field of current P2P research focuses on the extension of the P2P paradigm to ubiquitous computing scenarios, where mobile devices such as cell phones and PDAs or devices in a home network access the P2P overlay network via wireless technology. Such systems form a capable platform for mobile ubiquitous environments. For example, they leverage the composition of user-provided services or the deployment of context-aware applications. Moreover, from a mobile network operator's point of view, P2P systems save infrastructure costs, because they are based on user equipment and operate mostly without central entities.

P2P architectures that are based on distributed hash tables (DHTs), such as Pastry [1], Chord [2], CAN [3] or Kademlia [4], are appropriate architectures for mobile ubiquitous environments. By using a hash function to assign unique IDs to all peers and their provided content, they set up a predetermined overlay structure, e.g. a tree, a ring, or a *d*-dimensional torus. Provided content is mapped onto the closest peer in the ID space. "Closest peer" in this context refers to the peer with the lowest distance to the content's ID in the DHT's distance metric, e.g. the peer within the smallest subtree in Pastry or the subsequent peer on the ring in Chord. Resulting, DHTs allow direct routing of queries to the responsible peer, avoiding the need of flooding, as done in unstructured overlays. This reduces signaling overhead significantly, which is especially important for resource-constrained peers such as mobile devices. A further advantage of DHTs is their capability of guaranteed query resolution, whether the requested content is available or not.

II. SIGNALING TRAFFIC CAUSED BY PERIODIC REFRESH

DHTs map content that is provided by peers onto the responsible peer in the DHT, according to the DHT's mapping rules. For example, if peer *p* provides a document with ID *k*, Chord will store a key-value-pair <*k, p*> on the responsible peer for *k*, i.e. the peer with ID succeeding *k* on the Chord ring. The main drawback of this ID mapping is that peers store references to content that is provided by other peers. In case of a peer failure, all references stored on the failed peer are lost. Consequently, the related content is unavailable, although the providing peers are still online.[1] This is especially true for mobile environments, where we find high failure probabilities of participating peers, e.g. due to wireless link breaks or discharged batteries. To keep content references within the DHT up-to-date and to ensure content availability even in the presence of peer failures, all participating peers have to periodically refresh the references to their provided content. Huebsch *et al.* [5] evaluate this scheme by determining content availability as a function of node failure probability and refresh period length. However, they do not consider the signaling traffic that is generated by refreshing references, as described in the following.

Usually, refreshing a reference requires a lookup for every provided document to find the currently responsible peer in the DHT, and naturally this generates signaling traffic in the overlay network. As a result, we face a trade-off between content availability and signaling overhead: On the one hand, short intervals for refreshing provided content ensure high content availability and therefore high user acceptance, but on the other hand also generate a high amount of signaling traffic, which may be harmful to resource-constrained peers such as mobile devices.

---

[1] One approach to handle this problem is the replication of references on multiple peers. We abandon from this method because it generates additional signaling traffic and may not be reliable enough if a high percentage of participating peers fail, as expected in mobile environments. Moreover, replication linearly increases the number of references stored in the overlay network and thus the maintenance traffic generated when peers join or leave the system.

To estimate the impact of refresh intervals on the availability of provided content in a heterogeneous overlay network, we performed multiple simulations in a Chord overlay with 1,000 peers that were equally partitioned into five classes: Cable, DSL, ISDN, UMTS and GPRS. Particularly, we assigned varying mean online times, failure probabilities, number of shared documents and mean time between queries to each class. The detailed values are given in Table 1.

Table 1: Modeling of peer classes

	Cable	DSL	ISDN	UMTS	GPRS
Mean online time	10 hrs	5 hrs	30 min	10 min	5 min
Failure probability	5%	5%	15%	30%	50%
Number of shred documents	100…200	50…100	25…50	10…20	5…10
Mean time between queries	10 min	10 min	5 min	60 s	30 s

To evaluate content availability we refer to the success rate of queries. The success rate of a query is defined by dividing the number of hosts that are returned for a query for key *k*, by the actual number of hosts in the DHT that are currently providing the document mapped onto key *k*. For example, if a query result for *k* contains peers p_0 and p_1 as hosts of the document with hash value *k*, and peers p_0, p_1 and p_2 are currently providing this document, then the query success rate will be ⅔ or 67%.

The simulation results depicted in Figure 1 and Figure 2 verify our above assumptions. Figure 1 shows increasing content availability for shorter refresh intervals, i.e. the more often references are refreshed, the

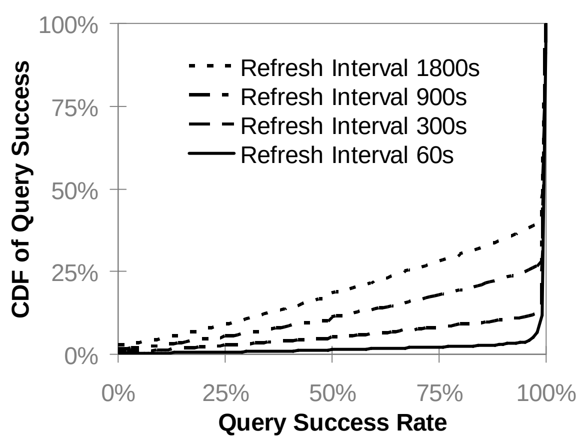

Fig.1: Content availability in conventional DHT

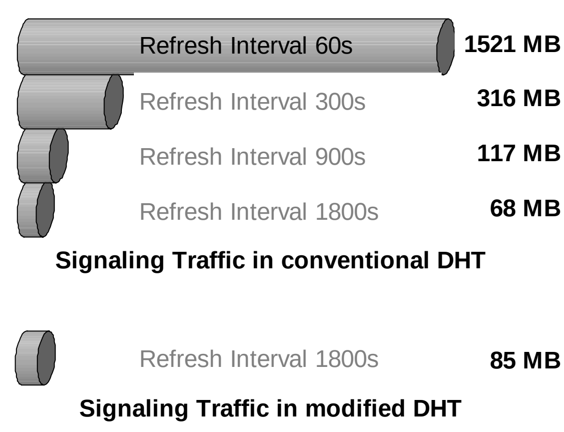

Fig.2: Signaling traffic caused by refresh

lower is the proportion of unsuccessful or low-successful queries. However, we also encounter increasing signaling traffic with shorter refresh intervals, as shown in the upper part of Figure 2.

III. SOLUTION

To avoid the loss of references and thus the unavailability of content caused by frequent peer failures, we divide participating peers into two classes, namely reliable peers and unreliable peers, and store references only on reliable peers with low failure probability. In contrast, unreliable (e.g. mobile) peers store only a pointer to their closest reliable peer in the DHT, where they forward all incoming queries for keys they in fact were responsible for. In practice, such differentiation requires only a minor modification to the conventional DHT's join procedure: When connecting to the overlay network, a new peer determines whether it classifies as reliable or unreliable peer, and accordingly it stores content references or only a next-closest-peer pointer. This differentiation, depicted for Chord in Figure 3, ensures that all peers that store references to shared documents are very unlikely to fail. Thus it offers high content availability, while at the same time it allows longer intervals for refreshing shared content, and therefore reduced signaling overhead.

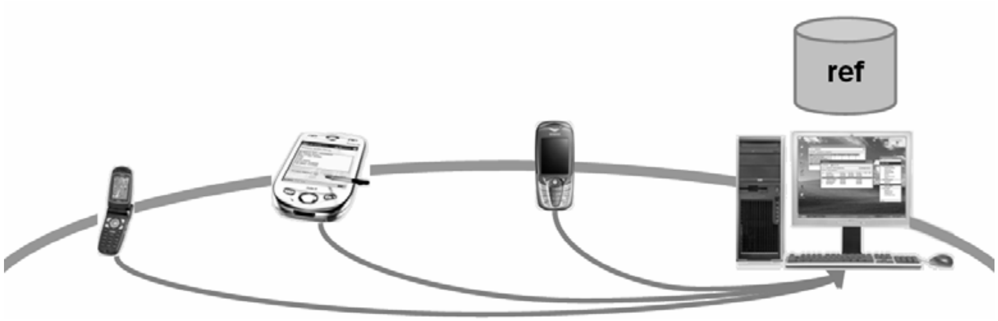

Fig.3: Content mapping in a modified Chord DHT

Simulation results shown in Figures 2 and 4 prove that our modification significantly increases content availability while allowing long intervals for refreshing provided content. In our particular scenario, we compare storing references only on reliable peers (Cable and DSL peers) and refreshing them every

1800s, to storing references on all peers and refreshing them every 60s and 1800s, respectively. While our modification requires only a slightly increased signaling traffic (resulting from forwarding incoming queries from unreliable to reliable peers, see Figure 2), it offers an excellent content availability. As shown in Figure 4, nearly all queries offer a 100% success rate when storing references only on Cable and DSL peers and refreshing them every 1800s. The content availability is even higher than for a conventional Chord DHT with a refresh interval of 60s.

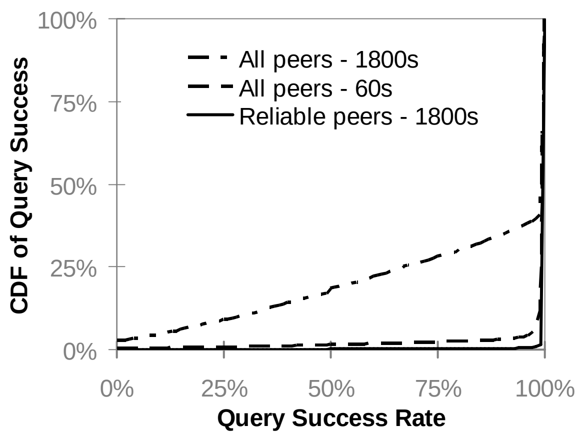

Fig.4: Content availability in modified DHT

REFERENCES

[1] A. Rowstron, and P. Druschel, "Pastry: Scalable, Distributed Object Location and Routing for Large-Scale Peer-to-Peer Systems", IFIP/ACM International Conference on Distributed Systems Platforms (Middleware), 2001.

[2] I. Stoica, R. Morris, D. Karger, M. Kaashoek, and H. Balakrishnan, "Chord: A Scalable Peer-to-Peer Lookup Service for Internet Applications", SIGCOMM 2001.

[3] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, "A Scalable Content-Addressable Network“, SIGCOMM 2001.

[4] P. Maymounkov and D. Mazieres, "Kademlia: A Peer-to-Peer Information System Based on the XOR Metric", IPTPS 2002.

[5] R. Huebsch, J. M. Hellerstein, N. Lanham, B. T. Loo, S. Shenker, and I. Stoica, „Querying the Internet with PIER”, VLDB 2003.