

# Augmenting Home Routers for Socially-Aware Traffic Management

Andri Lareida\*, George Petropoulos\*\*, Valentin Burger§, Michael Seufert§, Sergios Soursos\*\* and Burkhard Stiller\*

\* Communication Systems Group, CSG@IfI, University of Zürich, Switzerland, {lareida, stiller}@ifi.uzh.ch

\*\* Intracom SA Telecom Solutions, Peania, Greece, {geopet, souse}@intracom-telecom.com

§ Institute of Computer Science, University of Würzburg, Germany, {burger, seufert}@informatik.uni-wuerzburg.de

**Abstract**—Mobile users’ Quality-of-Experience (QoE) is degrading as network usage increases while Internet Service Providers (ISP) face increased inter-domain traffic. This paper presents a network traffic management mechanism, named RB-HORST, addressing these inefficiencies. RB-HORST exploits home routers by using them as caches and forming an overlay network between them to transfer content. To shift traffic from peak hours, RB-HORST employs predictions based on social network properties and based on similarity in the overlay network. To further improve user QoE, home routers allow trusted mobile devices to offload their mobile connection to the local WiFi. Simulation results show that an overlay is imperative for the success of the proposed caching mechanism. Especially ISPs with a large number of customers can benefit if only every thousandth user shares its router, reducing inter-domain traffic by half and superseding an ISP operated cache. The presented implementation proves that the concept is technically feasible and can be deployed and run on constrained devices.

## I. INTRODUCTION

Mobile devices are already a central part of everyday’s life, as shown by the average number of hours consumers spend with their smartphones or tablets [23]. Cisco estimated that *mobile video traffic* grew to 55% of the overall mobile traffic by the end of 2014 [10]. However, even if mobile data rates and Internet access bandwidth increase, so does the number of connected devices. In this ecosystem, the *Quality-of-Experience (QoE)* for the end-users plays an important role in the consumption of video content. As Ericsson reports after studying YouTube sessions tracked from users in Sweden and the US [11], there is a clear relation between perceived throughput and viewing session length: the higher the throughput, the longer the viewing time. With video resolution quality increasing, it is obvious that higher throughput will be required and additional video traffic will be destined from Cloud-based infrastructures to consumers’ mobile devices.

Furthermore, *Online Social Networks (OSN)* have also attracted a lot of attention. Ericsson’s report states that mobile data traffic from social networking amounts for 15% of the total mobile traffic, and that this percentage will be preserved until 2020, rendering the social networks as the second largest mobile traffic source. And this is of no surprise, since Facebook reported that by the end of 2014, the mobile Daily Active Users and the mobile Monthly Active Users have increased by 34% and 26% year-over-year, respectively [12]. Moreover, since June 2014 when video uploads were introduced in

Facebook, an average of more than 1 billion videos have been viewed every day. On average, more than 50% of people visiting Facebook every day in the US watch at least one video daily and 76% of people in the US using Facebook say they tend to discover the videos they watch on Facebook. One can easily understand that social networks constitute a modern and pervasive way for content dissemination and distribution.

It becomes apparent that the role of social networks cannot be neglected and trends for video consumption by mobile devices need to be considered upon designing modern traffic management mechanisms. Deducing content dissemination patterns from social networking relationships, when combined with traditional traffic management solutions, can result in efficient management of the generated traffic for the network operator and can guarantee a high QoE for the end-users. Moreover, existing social communities can be exploited to introduce smart WiFi roaming and mobile data offloading solutions.

In this context, this paper proposes the **RB-HORST mechanism**; a sophisticated combination of the “Replicating Balanced Tracker” (RB-Tracker) [19] and the “Home Router Sharing based on Trust” (HORST) [26] mechanisms. RB-HORST improves the QoE of end-users by exploiting available social networking information so as to prefetch interesting content likely to be consumed. To achieve this, RB-HORST allows for the formation of an overlay of *Home Routers (HR)*, based on the common content that their owners consume. Through this overlay, cached content on some HRs can be transferred to other HRs, whose owners are likely to request it in the near future. To estimate the probability of requesting a specific content, RB-HORST exploits available social information, as well as overlay metrics. In addition to the social-aware content prefetching, information available from social networks is used to enable the socially-aware WiFi roaming. In this case, social information is used to build a network of trusted users and allow for the exchange of WiFi credentials between a local Access Point and a trusted user in its vicinity.

The remainder of this paper is structured as follows: Section II provides an overview of existing technologies and research work that are similar to and partly address core aspects. Section III designs the RB-HORST mechanism, focusing on its core functionality. Section IV evaluates the system, with respect to its *caching* and *traffic reduction* performance. In

Section V, major details of the prototype implemented are provided, while Section VI draws conclusions.

## II. RELATED WORK

Several works relate to the RB-HORST approach. Thereby, the focus is laid on the three most important components of the RB-HORST mechanism: WiFi sharing, nano data centers and overlay networks, and social awareness.

Ubiquitous Internet access via WiFi is an emerging trend followed by both commercial services as well as research work. Fon [1] started to build a **WiFi-sharing** community by offering a home router with a public and a private encrypted SSID. The public WiFi was accessible for every community member. Similar approaches encompass the portable WiFi hotspot Karma [3] and hotspot databases like WeFi [7]. The research community investigated incentives and algorithms for broadband access sharing [21] and described ubiquitous WiFi access architectures for deployment in metropolitan areas, e.g., [30]. [18] presents a trust-based WiFi password sharing via an Online Social Network (OSN). RB-HORST is close to this work, however, RB-HORST extends it and combines it with content delivery via home routers and the incorporation of socially-aware traffic management.

**Nano data centers (NaDa)** [29] determine a distributed computing platform on ISP-controlled home routers. They show to be significantly more energy efficient for content delivery compared to traditional data centers by reusing already committed baseline power, avoiding cooling costs, and reducing network energy consumption. [14] evaluates the performance of peer-assisted video-on-demand streaming. Shared WiFi routers can be utilized for prefetching of content [22], which reduces the perceived end-to-end delay up to 50%. [16] proposes for the end device, i.e., a NaDa serving as its own cache, to prefetch overnight both content, which is globally popular, and content, which is of personal interest. Results show a high potential to reduce both response time and energy consumption compared to different access technologies.

Different NaDas can form a CDN (Content Distribution Network) on their own by communicating via an **overlay network**. The B-Tracker (balanced tracker) approach [15] defines a decentralized tracking mechanism that can be used to track resources in an overlay network. B-Tracker uses a Distributed Hash Table (DHT) plus direct messages for tracking, which balances better than pure DHT tracking approaches. The RB-Tracker [19] approach is an extension to B-Tracker involving replication of content tracked between nodes and locality. These extensions are not possible in a pure DHT tracking approach, since direct messages are required. Furthermore, B-Tracker nodes actively participating in the sharing of a content are also responsible for tracking said content.

**Social awareness** defines a novel approach [27]. It comprises, for example, replica placement and cache replacement algorithms, which are enhanced with social information from OSNs to predict future access to user generated content, e.g., videos. [24] identifies social cascades are identified in an OSN and locations of potential future users (i.e., OSN friends of

previous users) are taken into account for placing replicas. [25] also analyzes social cascades and geo-information of users to recognize locally popular content and keep it longer in the cache. For video streaming, specialized solutions, e.g., [20], exist, which explore social relationships, interest similarity, and access patterns for efficient prefetching to improve users' QoE.

Compared to the presented related work, RB-HORST combines these different technologies and concepts in order to exploit synergies in an advanced and integrated approach. These synergies include the efficient content delivery for operators and high QoE for end-users based on individual content prediction (social awareness) and a distribution network of home routers (nano data centers and overlay), which additionally enables the offloading of mobile connections to WiFi (WiFi sharing).

## III. DESIGN

Driven by the design of the RB-HORST overlay network the functionality of mechanisms for caching, prefetching, and WiFi offloading are defined and discussed.

### A. RB-HORST Overlay

To download content from close nodes, HRs must be able to find these nodes first. Since a centralized approach, i.e. an index server, can have scalability issues with a large number of clients and induces management costs, a decentralized approach is used in RB-HORST to establish connections between nodes. Connecting nodes directly allows them to communicate and, therefore, discover new content and providers for such content independently.

Using an overlay in form of a DHT is a common way of solving this problem. However, node identifiers (ID) in a DHT are distributed randomly and subsequently the load on nodes makes it not feasible to add features like locality and prediction. For this reason, RB-HORST builds an overlay following the RB-Tracker approach [19], which uses a DHT for the first lookup and sends direct messages to nodes sharing a video to find closer ones. The main benefit of RB-Tracker compared to a pure DHT is the load balancing, because the more nodes share a specific video the more nodes will be responsible for managing the overlay.

Whenever a content is requested by a client or a prediction algorithm, the DHT will be queried to find an initial set of providers for this content. These providers are sorted by Autonomous System (AS) hops and the provider with the lowest hop count will be queried for more providers (i.e., a close provider will be queried for more close providers). Eventually, one or more providers from the same AS or a maximum of one (1) hop away will be found and the content will be downloaded from this provider. In case the closest provider is two (2) or more hops away, the content will be downloaded from its original source, e.g. the data center. One AS hop typically implies that the two ISPs involved have a peering agreement, since no transit can be involved.

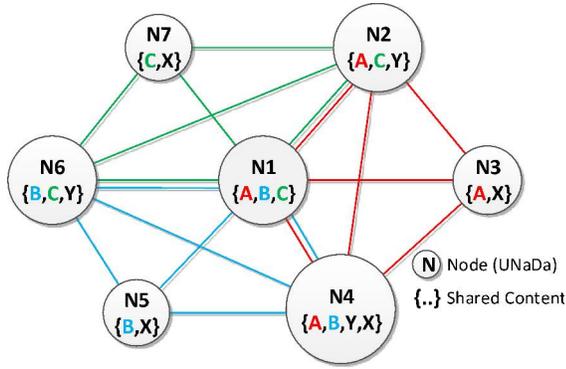


Fig. 1: Schematic view of an overlay example with node 1 at its center.

Figure 1 shows an example of how a simple overlay could look like from the perspective of node 1 (N1). N1 offers three content items (A, B, and C) in its cache and for each of these it maintains a list of providers that are depicted as edges between the nodes. For instance, node N1 knows N2, N3 and N4; all being providers of content A. N1 has two connections to nodes N2, N4, and N6, since N1 shares two common contents with each of them. However, keeping N2, N4 and N6 as providers for two contents does not imply that two connections on the transport layer, *e.g.*, TCP (Transport Control Protocol) or UDP (User Datagram Protocol) connections, are needed.

Downloading content from HRs reduces the load on data centers [29], but it can increase inter-domain traffic if the source is far away from the downloader. To prevent unnecessary inter-domain traffic, RB-HORST uses AS hops as a distance metric, because inter-domain traffic can only happen at AS borders. AS hops are resolved by using traceroute combined with an AS lookup service or database. Since CDNs bring content close to consumers, RB-HORST downloads only from HRs which are a maximum of one AS hop away. Thus, there can be no transit provider in the path, since transit would imply at least two AS hops.

### B. Caching and Prefetching

The goal of caching is to mitigate load on the network by avoiding repeated transmissions of the same content, while prefetching additionally utilizes off-peak periods (*e.g.*, at night) for downloading videos. However, content items to be prefetched have to be selected carefully in order to avoid additional traffic without any gain.

RB-HORST predicts content that will be consumed by clients of a HR. The prediction consists of two separate processes. The first is based on the overlay and the second is based on social information from Facebook. The overlay-based prediction uses information on cached/watched videos from other RB-HORST-enabled HRs. The social prediction considers, among others, content shared by friends, the location of a user, and the age of a video. Both prediction processes run in parallel and each has its own limit of cache size. An

additional mechanism must be designed that ensure, that a content is not prefetched twice.

In fixed time intervals (probably once an hour) the caches have to be updated. Thus, a new ranking is requested from the prediction components. The process starts prefetching the top ranked videos until the cache is full. Irrelevant content is replaced in an LRU (Least Recently Used) fashion.

1) *Overlay Prediction*: The overlay prediction uses information available from nodes directly connected, termed neighbors, to discover and rank the new content. The underlying assumption is that users watching the same videos have similar interests and are, therefore, a good source to discover new content. Over time, the overlay prediction will influence a HR to prefer connections to similar HRs rather than to HRs with different preferences. As a result, the overlay prediction produces a ranked list of content based on a calculated score with the highest score indicating the most likely one to be watched.

Every HR periodically contacts all its neighbors and asks them for a list of their cached contents. The HR compares each neighbor's content list to its own and the similarity, defined as the number of common content items, is calculated. The uncommon content items, *i.e.*, discovered content items, are new for the HR and are added to the ranking. The score of this ranking is calculated by going through the content items discovered and adding together the similarity values of each neighbor they were found on.

An example of the ranking algorithm is given according to Figure 1, which shows neighbors of Node A and the content cached (*e.g.* A,B,C) by those nodes. N1 is executing the prediction and collects the content lists from its neighbors: N2ACY, N3AX, N4ABYX, N5BX, N6BCY, N7CX. For each content list the similarity is calculated by counting common files to its own file set ABC: N2 = 2; N3 = 1; N4 = 2; N5 = 1; N6 = 2; and N7 = 1. In the next step, the discovered content items' ranking score is calculated by adding the similarity values of those nodes that are present together: Y = 2 + 2 + 2 = 6 and X = 1 + 2 + 1 + 1 = 5. The outcome is that even though X is more popular (shared by 4) than Y (shared by 3), Y is the better candidate to prefetch, because it has a higher chance of being in N1's interest. Figure 2 gives a general description of this algorithm in pseudo code.

Since this algorithm does not need to run on demand or in very short time intervals, the overhead generated by collecting content lists of all neighbors is expected to be low. Although connections are maintained for each content in a cache, many of them will be the same, since nodes will connect preferably to nodes that have many common content items.

2) *Social Prediction*: The goal of the social prediction is to improve the media experience of an individual user. Videos can be prefetched in higher quality and (due to the higher WiFi bandwidth) can be streamed to the end-user without stalling. The prediction is based on users' social network, in which users tend to connect with friends that are similar in behavior and taste. Therefore, every HR monitors both (a) the Facebook news feed of his owner with a dedicated client and (b) the

```

public List<String> getPrediction()
    local = getLocalContentList()
    discovered = new List()
    for each Ni in neighbors do
        Ci = Ni.queryContents()
        Si = 0;
        for each cont in Ci do
            if local.contains(cont)
                Si++;
            else
                discovered.add(cont)
        done
        for each cont in Ci do
            if discovered.contains(cont)
                cont.score += Si
        done
    done
    discovered.sort
    return discovered

```

Fig. 2: Overlay prediction pseudo code.

requested content in the system.

Periodically, the HR access the news feed of the owner and extracts all video URLs (Uniform Resource Locator). It ranks the content based on age (age of video), distance (distance of closest content resource in cloud or overlay), history (number of views of user), popularity (number of global views), and social information (percentage of Facebook friends who posted the video). These five dimensions were chosen based on the authors' reflections, which are supported by results from literature, *i.e.*, [13], [31] analyzed the popularity and age of user-generated content and their influence on content consumption, [20], [31] consider social and geographical locality, and [33] showed that a significant number of users tend to watch a video multiple times.

RB-HORST computes and updates the prediction based on the logged video consumption of the end-user. To account for the impact on the video consumption probability, for each of these five dimensions a score is obtained from appropriate functions, which were modeled based on the analysis of available Twitter datasets [17], [32] (*cf.* Table I). These functions use mathematical models to avoid overfitting to the training dataset. These five dimension scores of each video and the number of local views are stored in the user's access log. For each video in the user's news feed, the dimension scores are updated and the prediction score is computed by logistic regression. As it is not clear to what extent the model obtained from the Twitter dataset also applies to Facebook, we implemented an adaptive approach. Therefore, we keep the model functions of Table I, but the weights of each dimension are only initialized with the weights computed from the Twitter dataset and are periodically updated by applying a maximum likelihood estimation based on the actual (Facebook) data in the user's access log. The overall prediction score is used by RB-HORST to create a content ranking and to prefetch the

videos with the highest scores to the cache.

### C. WiFi Offloading

The RB-HORST system establishes two separate WiFis (a private and a public SSID (Service Set Identifier)) and enables mobile data offloading for end-users registered with the RB-HORST service. For that purpose, an end-user application and a Facebook application are required. The Facebook application is responsible for user management and social relationships, while the end-user application handles the user's authentication and transparent connection to the private SSID.

Initially, the end-user device application connects to the shared SSID and requests access to the private one, which triggers a decision, if the mobile user can be trusted. In case the user is found trustworthy, in terms of social relationships, the end-user application is provided with the private SSID credentials, connects to the determined WiFi and can access the Internet. This reduces the load on the cellular link and will eventually lead to less costs for mobile network operators and end-users, who do not have a flat rate data plan or who are roaming. Moreover, end-users experience a higher bandwidth and their mobile device has a lower energy consumption because, of lower signal strength and faster data transmission.

### D. Privacy

RB-HORST processes data obtained from Facebook which is by definition sensitive and needs to be treated with care. The main advantage of RB-HORST's distributed approach over more traditional cloud based services is that sensitive data never leaves the user owned device. Having only the data of one or a few users on a device reduces the exposure to attacks since the prospect of finding a few user's data is not worth the effort of hacking such a device.

For the overlay-based prediction, the HRs need to exchange lists of the contents residing in their caches. These lists are handed out to any HR asking for it. However, RB-HORST offers no means to map a content list to an actual user. The HRs are only known by their IP addresses to their neighbors, which does not allow to deduce the identity of an individual user.

## IV. EVALUATION

To evaluate the performance of a CDN supported by home routers, the paper simulates two scenarios. The first scenario simulates requests to a CDN with caches organized in a tree structure and compares isolated caches to cooperating caches to assess the benefit of the overlay. The second scenario adds an AS topology with peering and customer-to-provider links to evaluate the inter-domain traffic saving potential. A customer-to-provider link exists between a customer ISP and its transit provider, if the customer ISP pays the transit provider to forward its traffic destined to parts of the Internet that the customer ISP does not own or cannot reach.

To assess the benefit of the RB-HORST dependent on the number of shared home routers and the size of the ISP, the paper assumes and evaluates a tiered caching architecture with

TABLE I: Functions for computation of dimension scores in social prediction.

Dimension	Input	Score Function
Age	Time difference in weeks	$4.898 \cdot 10^{-4} \cdot \exp(-2.951 \cdot x) - 2.961 \cdot 10^{-7} \cdot x + 1.111 \cdot 10^{-4}$
Distance	Distance in km	$2.309 \cdot 10^{-3} \cdot \exp(-0.01 \cdot x) - 2.45 \cdot 10^{-9} \cdot x + 7.92 \cdot 10^{-5}$
History	Number of local views	$\min(0.1377 \cdot \log(0.2952 + 0.5 \cdot x) + 0.1683, 1)$
Popularity	Number of global views	$3.98 \cdot 10^{-2} \cdot \exp(-6.66 \cdot 10^{-8} \cdot x) + 2.4 \cdot 10^{-6}$
Social	Percentage of friends who posted video	$1.892 \cdot 10^{-3} \cdot \exp(5.641 \cdot 10^{-2} \cdot x) + 1.511 \cdot 10^{-2}$

resource locations at three different tiers, including the main data center of the content provider, CDN caches, and end-user equipment. The number of different content items to be downloaded or streamed from the resources is specified by the catalog size  $N$ . A Tier-1 resource is the data center of the content provider, where all  $N$  content items are stored. Tier-2 resources are edge caches and ISP caches, typically organized in a CDN, which are located close to Internet exchange points or within ISP networks. Requests served by ISPs or edge caches produce less or no inter-domain costs. Thus, these caches are referred as ISP caches in the following. The capacity of ISP caches is given as a fraction of  $N$  and is specified by  $C_{ISP}$ . The caching strategy of ISP caches is LRU. Within tier-3, the caches are placed on shared HRs that run the RB-HORST mechanism. These caches are referred to in the following as home routers (HRs). The cache capacity of HRs is specified by  $C_{HR}$  and their caching strategy is LRU. In this study the  $C_{HR}$  is set to four (4) content items.

The paper evaluates the performance dependent on the autonomous system size  $n_{user}$ , in terms of the number of end-users in the autonomous system. The probability that an end-user has RB-HORST installed and shares contents from its HR is given by  $p_{share}$ . The probability that a user requests certain content items depends on the content’s popularity distribution, which is specified by the Zipf exponent  $\alpha$ .

To evaluate the performance of the overlay, two cases are considered (a) the tree case and (b) the overlay case. In the tree case (a), each user is assigned to one shared HR in its AS which runs RB-HORST. If a user shares its HR, it is assigned only to its HR. A requested item is looked up in the assigned HR initially, i.e. in the tier-3 cache. If the requested item is not found, the request is forwarded to the next tier. The hierarchic caching strategy is leave-copy-everywhere, which means that the video is cached in each cache on the look up path. In the overlay case (b), a requested item is looked up in the HR of the user, if it is not found, it is looked up in shared HRs in the same autonomous system using the overlay. If no tier-3 cache in the AS contains the item it is looked up in tier-2 caches and finally in the data center of the content provider. The hierarchic caching strategy is leave-copy-everywhere, too, with the constraint, that the item is cached in the tier-3 cache only, which was looked up first.

As the goal of this evaluation is to assess the potential of the RB-HORST mechanism and to identify success scenarios, the simulation model assumes that the upload rate of caches is unlimited. However, in practice the upload rate limits the number of requests that can be served by a cache, especially for smaller devices like HRs. The evaluation uses a static

and global popularity distribution. In practice the item request process is dynamic and dependent on personal and regional preferences. The simulation uses a catalog size of  $N = 10^6$ . The results obtained show the average of ten simulation runs with  $10^6$  requests and their respective 95% confidence intervals.

Figure 3a shows the hit rate of the overlay dependent on the sharing probability for a constant ISP cache capacity of  $C_{ISP} = 0.01$ . In the tree case, where each user is assigned to a HR as a tier-3 cache, the hit rate is independent of the sharing probability. The hit rate is limited by the cache capacity of the HR. If HRs are organized in an overlay, their hit rate increases with the sharing probability, since requested content items are looked up in all HRs belonging to the overlay. This shows that an overlay highly increases the performance of a caching system with a high number of small caches. Hence, the RB-HORST approach highly benefits providers and end-users. The hit rate increases with the size of AS  $n_{user}$ , because a higher total cache capacity is available.

Figure 3b shows the ISP cache contribution dependent on the sharing probability for a constant ISP cache capacity of  $C_{ISP} = 0.01$ . In a tree structure the sharing probability has no significant impact on the ISP cache contribution. This depends on the fact that the hit rate of tier-3 caches is low and independent of the sharing probability. All remaining requests are forwarded to the ISP cache and in case of a hit the ISP cache contributes. If the HRs are organized in an overlay, the ISP cache contribution decreases because more requests can be served from the overlay. In this case the ISP cache also gets less efficient because it is only requested for rare items that are not cached in the overlay. For large ASes with a high number of end-users the ISP cache contribution approaches zero, if at least every thousandth user shares its HR. In this case the ISP cache can be shut down, which saves operating costs and energy. This shows that especially large ASes can benefit from systems like RB-HORST.

Figure 3c shows the inter-domain traffic dependent on the sharing probability for an AS with  $n_{user} = 10^6$  end-users. If no overlay is present the sharing probability has close to no impact on the amount of requests served locally. In this case the inter-domain traffic can only be reduced by increasing the ISP cache capacity. In the overlay case the number of requests served locally increases with the sharing probability, which decreases the inter-domain traffic. Dependent on the ISP cache capacity a higher fraction of shared HRs is necessary to reduce inter-domain traffic.

The RB-HORST overlay is not only used to access content from HRs in the same AS, but also from HRs in neighboring

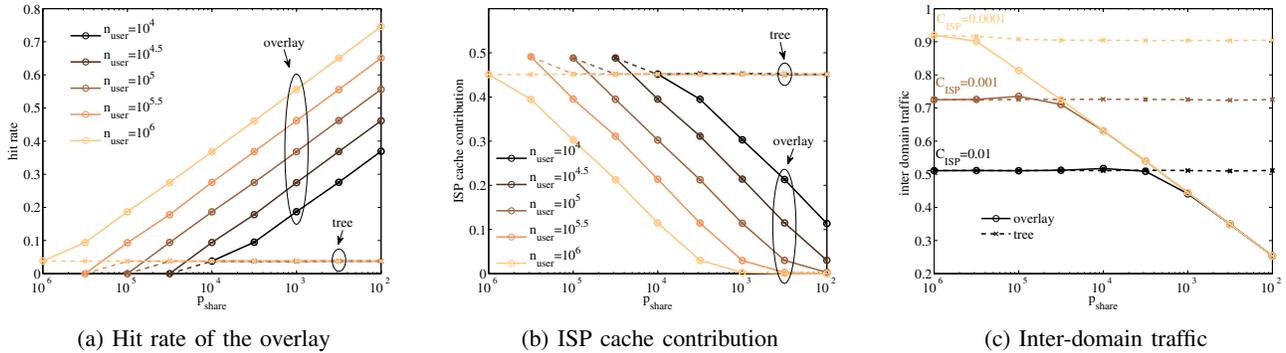


Fig. 3: Overlay and ISP cache contribution and inter-domain traffic dependent on home router sharing probability.

ASes. If the neighboring AS is a peering or customer ISP, no transit costs are incurred. To assess the inter-domain traffic saved by RB-HORST, an AS topology is added to the simulation. The AS relationship dataset provided by caida.org [8] of January 2015 is used and it specifies peering and customer-to-provider links of each AS. The data set consists of 46,172 ASes and 177,000 links. To be able to process the simulation the topology is limited to RIPE NCC EU ASes. The remaining subset still consists of 31,256 ASes and 77,382 links. The number of users per AS is determined by evaluating the Internet Census Dataset [9], which provides a scan on active IP addresses in the Internet. Assuming that the number of users in an AS is proportional to the number of active IP addresses and the probability of a user being in an AS is set accordingly. To save costly inter-domain traffic and to mitigate load on ISP caches, the following resource selection policy is applied:

If an item is not found on RB-HORST-enabled HRs in the same AS, it is requested from other resources in the order:

- 1) HRs in peering ISP ASes
- 2) HRs in customer ISP ASes
- 3) ISP cache in local AS
- 4) ISP cache in peering ISP ASes
- 5) ISP cache in customer ISP ASes
- 6) content provider

A policy designed to prioritize ISP caches to remote HRs did not have a significant impact on traffic savings. The threshold  $\theta$  specifies the minimum number of users an AS must have to host an ISP cache. If  $\theta = \infty$  no AS hosts an ISP cache and content delivery is solely supported by HRs. To investigate the performance of our approach, the impact of the HR sharing probability  $p_{share}$  on the inter-domain traffic and on the ISP cache contribution is studied. The share of traffic within the local AS, peering and customer-to-provider links is evaluated. For the generation of content item requests a Zipf popularity distribution with slope  $\alpha = 0.99$  was applied.

Figure 4a shows the share of requests served locally dependent on the HR sharing probability. More than 20% of requests can be served locally, if the ISP cache can store

1% of the catalog size. With an increasing threshold  $\theta$  the number of ASes hosting an ISP cache decreases and, thus, the share of requests being served locally. If the number of shared HRs increases, more traffic can be kept locally. This effect is stronger for a lower ISP cache capacity. In case of  $\theta = \infty$  where no ISP caches are available, the sharing probability has the strongest impact on inter-domain traffic. For a high sharing probability the ISP cache size has only little impact on the inter-domain traffic.

Figure 4b shows the ISP cache contribution dependent on the HR sharing probability. The number of requests an ISP cache can serve increases with its capacity. As for the inter-domain traffic, the sharing probability has a high impact on the ISP cache contribution. For high sharing probabilities the ISP cache contribution approaches zero. This means that ISP caches can be shut down, if a sufficient amount of users would share their HRs. For a lower threshold  $\theta$  more ISP caches are deployed and the ISP cache contribution increases.

To study the requests served per domain, the HR sharing probability is set to 1% and the threshold  $\theta$  to 100 users. Figure 4c shows the share of requests served per domain. Almost none of these requests can be served by the personal HR. This might depend on the fact that items are requested according to a global popularity distribution. If personal interests are considered in the demand model, higher hit rates and contributions from personal caches are expected. Dependent on the ISP cache capacity, 20 to 25% of requests can be served locally and 15 to 20% from neighboring ASes. Still about 2 out of 3 requests are served by the content provider. This depends on the fact that with Zipf slope of  $\alpha = 0.99$  content item requests are highly heterogeneous. In practice, temporal and social dynamics of users' interests will lead to temporal and local correlations in requests, which improve the performance of local and personal caches.

## V. PROTOTYPE

A proof-of-concept prototype of the RB-HORST mechanism has been implemented to validate and demonstrate the feasibility of the approach designed.

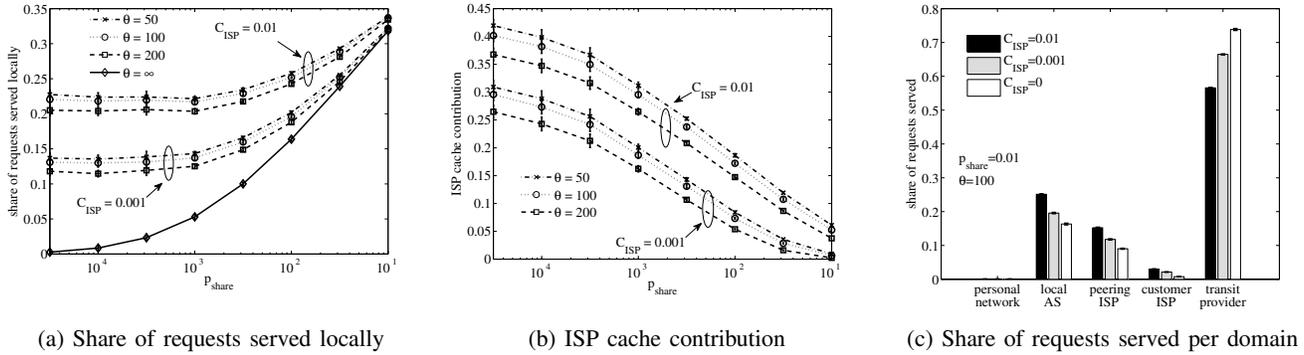


Fig. 4: Share of requests served locally and ISP cache contribution dependent on sharing probability and share of requests served per domain.

### A. Architecture

The RB-HORST mechanism focuses on end-users and is deployed on devices owned and controlled by them. The set of the designed functionalities map into two architectural entities (cf. Figure 5): the *Home Router (HR)*, which is expected to be hosted on residential access point devices, and the *End-User entity*, which is deployed in end-user devices, e.g., smartphones or tablets. The color-coding of components denotes whether a component is implemented by the RB-HORST mechanism (orange, grey) or is provided by external systems and software (white).

The formation of overlays between remote HRs is enabled by the *Overlay Manager*. It resides at each peer and is responsible for the communication between overlay nodes to advertise offered resources, to ask for available content, and to predict which content shall be prefetched by other peers. The Overlay Manager requires and receives input from the *Topology and Proximity Monitor*, which provides information on the distance between remote overlay peers, specifically measuring the number of AS hops towards them.

The social monitoring and aggregation functionalities are performed by the *Social Monitor* and the *Social Analyzer* components, respectively. The Social Monitor gathers information about social interactions of groups of end-users. It implements an interface to the *Facebook Graph API* to query

end-users' Facebook friends and posts and the *Vimeo API* to retrieve relevant information of Vimeo videos posted. This social information is provided to the Social Analyzer, which periodically predicts content likely to be watched.

The *Controller* determines the coordinating component of the HR, and prepares high-level decisions. Specifically, it receives and merges the output of social and overlay prediction algorithms, selects the content highly ranked to be prefetched and manages the local cache. The RB-HORST mechanism gives a high priority to highly-ranked content from the social prediction algorithm. For this purpose, score thresholds have been defined to avoid prefetching all ranked content from the social and overlay prediction algorithms. These thresholds periodically change and adapt to users' behavior and nodes requirements, especially to maintain the system's scalability and performance. If this content can be fetched from overlay peers belonging to the same or peering ISP domains, content is fetched from overlay nodes, otherwise directly from Vimeo servers. This also applies to content predicted by the overlay prediction algorithm. Low-ranked content is either removed from the local cache or ignored, if not cached. The pseudocode of the cache management algorithm is presented in Figure 6. LRU content is periodically removed, in case the cache is close to full. In addition, the Controller exposes an interface towards the End-User entity in order to receive authentication requests to access the private SSID.

Three additional supporting components also reside in the HR. The *Database (DB)* and *User Interface (UI)* components support the basic system and provide the necessary interface to the end-user to login and participate in the RB-HORST platform, as well as configure and manage his local router. The *Proxy* component intercepts end-users' requests and rewrites them in case the content exists in the local cache, while a new request results in the caching of the requested video.

The key End-User entity component is the *Mobile Network Traffic Manager*, which transparently communicates with the HR to retrieve SSID credentials to enable access to the Internet, following the WiFi offloading procedure described in Section III-C.

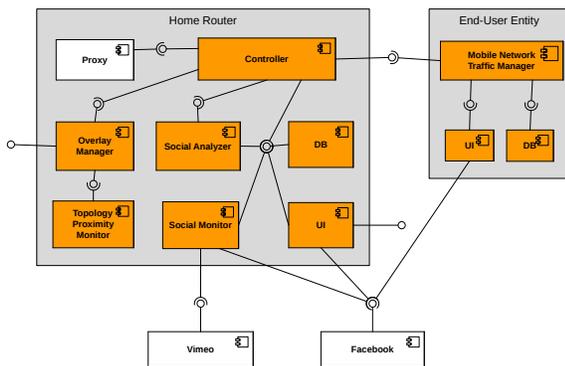


Fig. 5: The RB-HORST architecture.

```

public void updateCache(socialContentList,
    overlayContentList)
for each cont in socialContentList do
if not overlayContentList.contains(cont)
if socialScore > socialThreshold
    and newCacheSize < cacheSize
    VimeoDownloader.download(content)
else
    ignoreList.add(content)
done
for each cont in overlayContentList do
if overlayScore > overlayThreshold
    and newCacheSize < cacheSize
if overlayHops <= 1
    OverlayDownloader.download(content)
else
    VimeoDownloader.download(content)
else
    ignoreList.add(content)
done
deleteIgnored(ignoreList)

```

Fig. 6: Cache management pseudo code.

### B. Deployment

The presented system architecture is mapped into the deployment diagram of Figure 7, which presents the hardware, processes and artifacts used to deploy an instantiation of the RB-HORST mechanism, as well as the interfaces and protocols between entities and external systems.

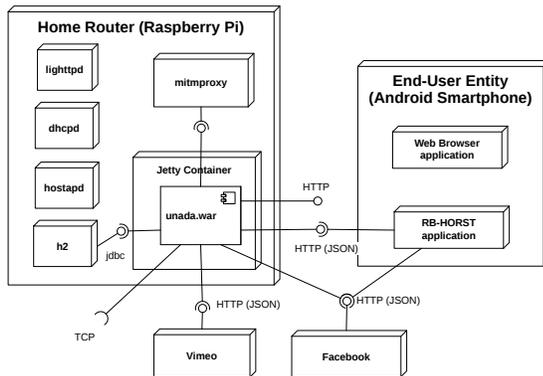


Fig. 7: The RB-HORST deployment diagram.

For the reference implementation of RB-HORST, the Raspberry Pi [5] was selected as the HR hardware, with Raspbian OS installed. Any hardware including a low-cost computer with minimum of 256 MB RAM, a WiFi dongle in support of two SSIDs, and an SD-card with a light Linux installed can host the HR.

The RB-HORST software for HRs is a Java Web application, which is deployed in the Jetty Web server [2], running the social and overlay prediction algorithms, and managing the system’s cache. The mitmproxy [4] software acts as

the system’s proxy, intercepting end-users’ requests, caching watched videos, and rewriting content requests when the requested video exists in the local cache. The End-User entity is implemented as an Android application and is installed on Android smartphones.

Each HR advertises or asks other overlay peers for content, location, or metadata, using the TomP2P [6] library. The Facebook and Vimeo APIs are periodically queried over HTTPs by the Social Monitor to retrieve and store social information. To support the Facebook Graph API monitoring functionality, a Facebook application was developed, where each user logs in and accepts the RB-HORST’s service permissions, in order to participate in the platform.

In addition, each router exposes a Web user interface, accessible through any Web browser, where end-users can login with their Facebook credentials and manage their local cache and access-point. A REST API is exposed by the HR and is triggered by the RB-HORST Android application, used for end-users’ authentication in remote HRs.

The source code of the RB-HORST mechanism has been released as open-source to Github [28] under the Apache License Version 2.0.

## VI. SUMMARY AND CONCLUSIONS

This paper presented RB-HORST – a socially-aware traffic management mechanism, which is based on user-owned HRs. Leveraging synergies from related work, the mechanism was designed such that HRs can be utilized for caching and prefetching of Vimeo video content relevant for the HR owner based on social information from Facebook. All participating HRs form an overlay, which allows for content distribution among close HRs, *i.e.*, HRs that are at most one AS hop away. Additionally, HRs offer access to WiFi for trusted users, which reduces the load on mobile networks. The design of this mechanism included the formation of the overlay, the overlay- and social prediction for caching and prefetching, the selection of the best content resource, and the WiFi offloading. By implementing a prototype based on a Raspberry Pi as HR hardware, it has been confirmed that the theoretical design of such a mechanism is practically feasible. Additionally, the source code of the mechanism’s software has been released as open-source.

The benefits of this modern mechanism were numerically evaluated in terms of caching and inter-domain traffic. The approach prototyped shows that an overlay is imperative for the success, especially by consisting of a high number of small caches like RB-HORST. Moreover, by investigating the share of locally served content requests, the impact for the network operator has been quantified. Those results obtained indicate that the RB-HORST mechanism significantly reduces the inter-domain traffic and required contribution of an operator-owned cache. In conclusion, the definition and operation of RB-HORST results in a win-win situation for both i) end-users, who benefit from improved video QoE, and ii) operators, who save costs at the same time. This holds true, once at least every

thousandth user shares his HR in large ISPs, as in consequence the ISP's-owned cache can be discontinued.

As future work, the authors will extend the numerical evaluation to cover the efficiency aspects of the overlay and social prediction. Moreover, the prototypical implementation will be used for an extensive experimental study of the RB-HORST for evaluating its performance in a medium-scale setting with real users. Both directions will lead to further strengthen the QoE improvement argument of RB-HORST, both theoretically and in real life scenarios.

#### ACKNOWLEDGMENTS

This work was partly funded in the framework of the EU ICT Project SmartenIT (FP7-2012-ICT-317846) and by Deutsche Forschungsgemeinschaft (DFG) under grants HO 4770/1-2 and TR257/31-2. The authors address their thanks to all project partners for their valuable contributions.

#### REFERENCES

- [1] "Fon Wireless, Ltd." [Online]. Available: <http://www.fon.com>
- [2] "Jetty - Servlet Engine and Http Server." [Online]. Available: <http://eclipse.org/jetty/>
- [3] "Karma Mobility, Inc." [Online]. Available: <https://yourkarma.com/>
- [4] "mitmproxy." [Online]. Available: <https://mitmproxy.org/>
- [5] "Raspberry Pi Foundation." [Online]. Available: <https://www.raspberrypi.org/>
- [6] "TomP2P." [Online]. Available: <http://tomp2p.net/>
- [7] "wefi." [Online]. Available: <http://wefi.com/>
- [8] Caida.org, "The CAIDA AS Relationships Dataset, 20150101," 2015. [Online]. Available: <http://www.caida.org/data/as-relationships/>
- [9] B. Carna, "Internet Census 2012 Port Scanning /0 Using Insecure Embedded Devices," 2013. [Online]. Available: <http://internetcensus2012.bitbucket.org/paper.html>
- [10] Cisco, "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2014-2019," Cisco, Tech. Rep., Feb. 2015.
- [11] Ericsson, "Ericsson Mobility Report - MWC Edition," Feb. 2015. [Online]. Available: <http://www.ericsson.com/ericsson-mobility-report>
- [12] Facebook Investor Relations, "Fourth Quarter and Full Year 2014 Results," Jan. 2015. [Online]. Available: <http://investor.fb.com/>
- [13] F. Figueiredo, F. Benevenuto, and J. M. Almeida, "The Tube over Time: Characterizing Popularity Growth of Youtube Videos," in *Proceedings of the 4th ACM International Conference on Web Search and Data Mining (WSDM '11)*, Hong Kong, China, 2011.
- [14] J. He, A. Chaintreau, and C. Diot, "A performance evaluation of scalable live video streaming with nano data centers," *Computer Networks*, 2009.
- [15] F. Hecht, T. Bocek, and B. Stiller, "B-Tracker: Improving Load Balancing and Efficiency in Distributed P2P Trackers," in *IEEE International Conference on Peer-to-Peer Computing (P2P)*, Kyoto, Japan, 2011, pp. 310-313.
- [16] E. Koukoumidis, D. Lymberopoulos, K. Strauss, J. Liu, and D. Burger, "Pocket cloudlets," *ACM SIGARCH Computer Architecture News*, 2011.
- [17] H. Kwak, C. Lee, H. Park, and S. Moon, "What is Twitter, a social network or a news media?" in *Proceedings of the 19th international conference on World wide web*. ACM, 2010, pp. 591-600.
- [18] C. B. Lafuente, X. Titi, and J.-M. Seigneur, "Flexible Communication: A Secure and Trust-Based Free Wi-Fi Password Sharing Service," in *Proceedings of the 10th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, Changsha, China, 2011.
- [19] A. Lareida, T. Bocek, M. Waldburger, and B. Stiller, "RB-Tracker: A Fully Distributed, Replicating, Network-, and Topology-aware P2P CDN," in *2013 IFIP/IEEE International Symposium on Integrated Network Management (IM 2013)*, Ghent, Belgium, May 2013, pp. 1199-1202.
- [20] Z. Li, H. Shen, H. Wang, G. Liu, and J. Li, "SocialTube: P2P-assisted Video Sharing in Online Social Networks," in *Proceedings of the IEEE INFOCOM*, Orlando, FL, USA, 2012.
- [21] L. Mamas, I. Psaras, and G. Pavlou, "Incentives and Algorithms for Broadband Access Sharing," in *Proceedings of the ACM SIGCOMM Workshop on Home Networks*, New Delhi, India, 2010.
- [22] A. J. Mashhadi and P. Hui, "Proactive Caching for Hybrid Urban Mobile Networks," 2010.
- [23] Salesforce Inc., "2014 Mobile Behavior Report," 2014. [Online]. Available: <http://www.exacttarget.com/sites/exacttarget/files/deliverables/etmc-2014mobilebehaviorreport.pdf>
- [24] N. Sastry, E. Yoneki, and J. Crowcroft, "Buzztraq: Predicting Geographical Access Patterns of Social Cascades Using Social Networks," in *Proceedings of the 2nd ACM EuroSys Workshop on Social Network Systems (SocialNets)*, Nuremberg, Germany, 2009.
- [25] S. Scellato, C. Mascolo, M. Musolesi, and J. Crowcroft, "Track Globally, Deliver Locally: Improving Content Delivery Networks by Tracking Geographic Social Cascades," in *Proceedings of the 20th International Conference on World Wide Web (WWW2011)*, Hyderabad, India, 2011.
- [26] M. Seufert, V. Burger, and T. Hofffeld, "HORST - Home Router Sharing based on Trust," in *2013 9th International Conference on Network and Service Management (CNSM)*, Zürich, Switzerland, Oct 2013, pp. 402-405.
- [27] M. Seufert, G. Darzanos, I. Papafili, R. Łapacz, V. Burger, and T. Hofffeld, "Socially-Aware Traffic Management," in *Social Informatics - The Social Impact of Interactions between Humans and IT*, K. A. Zweig, W. Neuser, V. Pipek, M. Rohde, and I. Scholtes, Eds. Springer, 2014.
- [28] The SmartenIT Consortium, "The source code of the SmartenIT software." [Online]. Available: <https://github.com/smartenit-eu/smartenit>
- [29] V. Valancius, N. Laoutaris, L. Massoulié, C. Diot, and P. Rodriguez, "Greening the Internet with Nano Data Centers," in *Proceedings of the 5th international conference on Emerging networking experiments and technologies*. Rome, Italy: ACM, 2009, pp. 37-48.
- [30] P. Vidales, A. Manecke, and M. Solarski, "Metropolitan Public WiFi Access Based on Broadband Sharing," in *Proceedings of the Mexican International Conference on Computer Science (ENC 2009)*, Mexico City, Mexico, 2009.
- [31] Z. Wang, L. Sun, X. Chen, W. Zhu, J. Liu, M. Chen, and S. Yang, "Propagation-based Social-aware Replication for Social Video Contents," in *Proceedings of the 20th ACM International Conference on Multimedia*, Nara, Japan, 2012.
- [32] J. Yang and J. Leskovec, "Patterns of temporal variation in online media," in *Proceedings of the fourth ACM international conference on Web search and data mining*. ACM, 2011, pp. 177-186.
- [33] M. Zink, K. Suh, Y. Gu, and J. Kurose, "Characteristics of YouTube Network Traffic at a Campus Network - Measurements, Models, and Implications," *Computer Networks*, vol. 53, no. 4, pp. 501-514, 2009.