A NEW SCENARIO AND TECHNIQUES FOR CONTENT PREFETCHING IN 3G NETWORKS

Ricardo Romeral, David Larrabeiti, Manuel Urueña, Arturo Azcorra and Santiago Gallego *
Department of Telematic Engineering. Carlos III University of Madrid (UC3M). 28911 Leganés, SPAIN †
{rromeral, dlarra, muruenya, azcorra, ago}@it.uc3m.es

Abstract Web content prefetching is a well-known technique whose aim is improving the performance of web browsing. After a period of intensive study by the research community, the advent of wireless access technologies and 3G cellular systems in particular has brought back prefetching to the research front, and the suitability of this technique to optimise mobile wireless web surfing is being assessed in many recent works. This paper identifies specific issues in the application of prefetching to 3G networks, proposes a new scenario where prefetching can be effective, studies how this performance-oriented service can be integrated in the value chain (defining a realistic method for charging for this service) and describes a trial scenario being developed over a real 3G network supporting OSA.

Keywords: 3G, prefetching, OSA/Parlay

1. Introduction

The application of active devices to improve the performance of wireless networks is a major research topic at present. Indeed, two major challenges have drawn the attention of networking researchers: on the one hand, assessing the effectiveness of such devices, and, on the other, engineering methods to charge for these resource-consuming extra services. Examples of these performance-enhancing proxies are TCP relays,

*Disclaimer: this paper reflects the view of the authors and not necessarily the view of the referred projects.

†This work is being funded by the IST project Opium (Open Platform for Integration of UMTS Middleware) IST-2001-36063 and the Spanish MCYT under project AURAS TIC2001-1650-C02-01.

TCP spoofers, mobility-aware multimedia caches and web prefetchers. This paper focusses on this latter device.

During the last years, methods, heuristics and software to support prefetching have been developed [1], and it is with the advent of wireless access technologies -and the specific characteristics of each of these technologies- that this technique is being revisited by researchers e.g. [2] [3].

Today the mechanics of prefetching are well known and it is also clear that the performance gain is expensive in terms of bandwidth consumption and added traffic burstiness. However, in the context of Internet access through 3G networks, where the business model is quite different from the classical accesses, prefetching can be an added-value service that can be offered With this purpose, in this paper we present a new scenario for the practical implementation of a commercial prefetching service in 3G, identify the specific issues involved in the deployment of this service in a 3G network and present a tool to demonstrate it. In the following we shall use the term Instant Web Browsing (IWB) to refer to this prefetching service.

The paper is structured as follows. Section 2 reviews different scenarios for prefetching as studied in the literature. Section 3 cites specific works in the state of the art of prefetching techniques and tries to identify the most suitable approach for 3G. Section 4 isolates the problems of deploying the service in a concrete 3G architecture. Section 5 describes how to exploit the service in the 3G business model by making use of the 3GPP OSA (Open Services Arquitecture) interface. Finally, section 6 describes a prototype of prefetcher called *ink* being developed for demonstration of the IWB service.

2. Placing a prefetcher in a wireless network

Web prefetching is a technique that tries to improve the quality of service perceived in web browsing. The idea behind prefetching is rather simple: it tries to enhance an HTTP cache with initiative to retrieve in advance those web objects most likely to be downloaded by its users. This way, the hit ratio is increased and the effective latency perceived by the user is reduced, under the premise that there is an excess of bandwidth available. In this sense, effective prefetching is expensive in terms of bandwidth consumption; the best latency reductions are in the range of 40-50% at the expense of double bandwidth [4]. The limit comes from the increase of delay and burstiness caused by the extra traffic. Hence different strategies will be required depending on whether the link bandwidth is shared or not, and its cost. This depends on the

target scenario. Currently, there are three main application scenarios for prefetching:

- **Personal Local cache** or prefetching by clients from the web server. The prefetcher runs on the same machine as the user agent. This program, belonging to the family of so-called "Internet boosters", has as objective exploiting the spare bandwidth available in the user's low-speed circuit-switched Internet access (e.g. over a V.90 modem) to speed up web navigation. These devices observe and learn about the navigation behaviour of the user to compute link access probabilities.

- **Shared Network cache** or prefetching by proxies from the web server. This device attempts to make use of spare network bandwidth by prefetching objects according to the navigation history of many users i.e. select the most popular links for prefetching. Multimedia caches in Content Delivery Networks belong to this class of devices, which are supported by a number of heuristics based on user profiles and content descriptor records. Since this sort of device usually serves many users, it relies on behaviour information averaged over a large number of users.

- **Proactive Network cache** or prefetching by clients from a proxy. This scenario is somewhat equivalent to the Personal Local Cache, although this time the prefetching intelligence resides in the proxy [5]. The cache predicts the user behaviour and takes advantage of idle time between user requests to push documents to the user.

The interest of Personal Local cache and Proactive Network cache in the 3G context is null. The reason is that, unlike in modem or WLAN access, the cost model applied is based on traffic. This makes this service not scalable and unaffordable.

From the previous discussion it follows that the Shared Network Cache approach is the only scenario suitable for prefetching in 3G networks. Let us now review a set of prefetching techniques and results by previous researchers before discussing which specific features must be present in this cache.

3. Applying existing techniques to 3G

The research carried out in techniques to optimize web caching is huge. Therefore we shall try to cite only those works that bring key ideas about how to apply prefetching today in a practical 3G scenario. A more complete survey can be found in [1].

An important early work in prefetching techniques is [6] where the authors analyze the latency reduction and network traffic for personal local caches, and introduce the idea of measuring conditional html link access probabilities statistically. With this method and by using web-server-trace-driven simulation they obtain a reduction of 45% in latency at the cost of doubling the traffic. This can be considered a practical bound for prediction based on *Client access probabilities.*

With regard to prefetching with shared network caches [7] estimates the theoretical limits of perfect caching + perfect prefetching in a reduction of client latency of 60%. Another interesting work on Personal Local Caches is [8] where mobility is addressed as nomadic change from an access technology to another. Prediction experiments carried out by the authors show that: For a given *Prefetch Threshold*, PT, a better *Successful Prediction Rate* is achieved by *Client access probabilities* than with *Server access probabilities.* However, for a given PT a better *Hit Rate* is obtained with *Server access probabilities*, that implies that more is prefetched in vain but the delay obtained is lower.

In a context where the bandwidth is unexpensive (e.g. LAN), it seems worthwhile to set a low *Prefetch Threshold* and base predictions on server probabilities, as firstly proposed in [4]. However, in practice, doing so requires extensions of HTTP or HTML (so that the server can convey the client the probabilities associated to each link in the page). Therefore the only practical way to optimise user prediction today at the proxy is having personal behaviour computed and stored in the proxy.

Thus, we introduce a new kind of prefetching scheme, optimal for wireless access that have a packet-based charging scheme (i.e. 3G network), the **Personal Network cache**. This is a network cache where prediction is based on per-user navigation history. In other words, from the user's perspective it is as if his local personal cache is pushed over the wireless link. However, this does not preclude cached contents from being shared. What is personalised (based on client access probabilities (option a)) is the prefetching task. Moreover, this makes it possible to personalise the sort of prefetching performed on behalf of the user and to charge accordingly.

4. The 3G Scenario

Once defined the type of prefetching suitable for a 3G mobile network we shall study the location of the prefetcher inside this network. At this point we have to focus on a particular 3G architecture, although most of the discussion can be applied to the other 3G systems. The one chosen is the ETSI European standard of the 3G IMT-2000 system, known

as UMTS (Universal Mobile Telecommunication System) [9], launched commercially in Europe in the fourth quarter of 2003. Experiments of prefetching using the application described in section 6 are being carried out in the pre-commercial UMTS testbed of Vodafone in the context of Opium [10], along with a number of OSA/Parlay experiences.

The UMTS architecture is strongly influenced by compatibility with the 2G digital telephony system (GSM) and the switched packet data service evolved from it GPRS (General Packet Radio Service). Two conceptually new elements have been introduced: the SGSN (Serving GPRS Support Node) and the GGSN (Gateway GPRS Support Node). These devices are in charge of data packet switching. In outline, the SGSN deals with mobility across RNCs (Radio Network Controller), with following mobile stations in its service area and AAA functions; whereas the GGSN is the actual gateway to Internet (see [9] for further details).

The transport of IP packets inside the UMTS network is complex, depends on the dynamic creation of *contexts* for each data session, and that the only fixed point in the network where persistent caching is possible is just behind the GGSN. This is a limitation because it is more convenient to have the proxy near the terminal in order to adapt better to the link conditions. But it makes it unnecessary to move the state information (user access profile) from one cache to another as the user moves. Another reason to place the IWB here is that the 3G operator may not be willing to introduce a external device inside his core network.

Another potential hinder for personal network caching, where identification could be based on the users IP address, is the current session-oriented approach that creates a non-scalable implementation of the "always-on" facility (a fixed global public IP address permanently allocated to the terminal) due to the need to keep and update session state information (needed to locate a given IP address within the UMTS network). The solution proposed to solve this problem is bound to the problem of identifying and charging a user for the amount of prefetched information. This is addressed in the next section.

5. Third-party provisioning

As the radio spectrum is a scarce resource, wireless data network operators had a tighter business model than its wired counterparts, which usually allows flat rate subscriptions. Thus, in wireless networks, charging is implicitly a traffic regulation mechanism, and hence traffic accounting and billing must be an integral part of the operator network infrastructure.

To fill the general need of opening the telecommunication business to new actors, the Parlay consortium was created by the IT industry, including important telcos and vendors. Its main objective was the development of open, technology-independent, application programming interfaces that enable third parties to develop applications and technology solutions that operate across multiple networking technologies. These specifications do not only cover the accounting and billing operations but all the network related ones (call control, user interaction, multimedia messaging,..) including a framework for authentication & authorisation, which is a vital issue when dealing with charging users in a mobile environment.

3GPP adopted Parlay as the basis for its OSA (Open Service Architecture). That is why many of the early adopters of this technology are 3G operators willing to provide innovative services as added value for its 3G network to compete against the widespread 2G technologies. In order to be operator-independent, the OSA/Parlay group defined the Parlay gateway as a broker between the operator and third-party service providers. The main components of the Parlay release 3.0 adopted by the 3GPP as the Open Service Access (OSA) interface are, Framework, Call Control SCF, User interaction SCF, Mobility SCF, Terminal Capabilities, Data Session Control SCF, Account Management SCF and Charging SCF.

Our objective is to demonstrate that it is also possible to exploit performance-oriented services, such as web prefetching, with the authentication and billing capabilities of OSA.

In this scenario there are several actors: the user, the service provider and the UMTS network operator as an intermediator between them. The UMTS network operator provides Internet connectivity to the end user, plus authentication and billing on behalf of the pre-fetching service provider. Once the user identity is known (user MISDN) -as deduced from it current IP address- it is possible to check if the user is subscribed to the service and then retrieve service preferences and browsing history to feed the prefetcher. By integrating the service in the billing system of the UMTS operator, any charging model is feasible, ranging from flat-rate, to per-Mbyte of prefetched pages charging, etc. All these techniques allow service differentiation and customised per-user behaviour of the prefetcher.

In outline, the charging process works as follows: the pre-fetcher periodically issues billing records to the Parlay gateway according to the service/user behaviour. The Parlay gateway maps these billing records in AAA records and sends them to the UMTS network AAA server employing the appropriate network-dependant protocol. Later, the AAA

information will be processed by the operator billing information system and at last, the service charge will be integrated in the user bill.

6. Implementation

Ink is a prototype of web prefetcher developed in the context of IST project Opium. *Ink* makes use of a web proxy cache to transparently handle user requests, manage and maintain the cached web pages and to request pages to the servers. Figure 1 shows *ink* and its interaction with the other entities of the scenario.

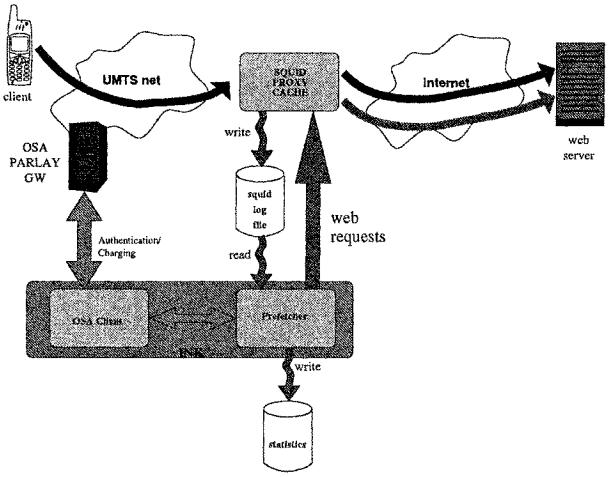

Figure 1. Software architecture of the *ink* program

The *Ink* process works in cooperation with *squid*. *Squid* is a well-known web proxy cache, developed as free software that has been widely deployed in the Internet. *Squid* can be used in transparent mode. All client request information are stored in a log file. This file contains a record of user requests (IP of the user and web page requested, if the solicited page is already in the cache, etc).

Ink starts to work when a new event occurs in the log file. *ink* reads the new entry. Then obtains the page solicited by a certain IP address (mapped to a user). *Ink* makes a resolution request (if this request is not previously sent) to the *OSA client* module to identify the user from the IP address of the request. The *OSA client* module connects to the OSA/Parlay Gateway and obtains the user identifier. The OSA/Parlay Gateway queries the appropiate entities inside the UMTS network (typically a RADIUS server in this case) in order to resolve the assignment between the user-IP address pair in this session.

Once the user identity is known -actually his/her MISDN number-, *ink* can find out if this user is a subscriber or not. If it is, *ink* loads the client

profile (only the first access). This profile contains the user history, the type of service, and the charging model. Up to date, there are different types of services defined in the prototype, No prefetching, Prediction algorithm (Jiang/Kleinrock) + extra heuristics, Prediction algorithm without images (and other multimedia links), Complete prefetch without recursion and Complete prefetch with a one level of recursion.

The Jiang/Kleinrock prediction algorithm uses the history of the web pages visited in the past by the user as well as from which page was visited to calculate the probability of the links in the web page that has been requested by the user. At this point, we have modified the process, to include some heuristics to add extra factors to these probabilities. These heuristics increase or decrease the download probability of a link. The prediction algorithm are explained in [8].

The links are ordered according to the altered access probabilities, and only the ones greater than a prefetch threshold are requested in advance. This threshold is evaluated in real-time based on subscriber category. For example: maximum bytes per time unit that the user can download, percentage of unused bandwidth, number of links, etc.

7. Measurements

For the test-bed we use the simulated scenario of the figure 2. We have simulated several scenarios: where no proxy, only the proxy without cache, with the cache clean and full (100% hit ratio), and when *ink* is present. In this case we have simulated three possibilities, 80%, 53% and 27% hits of requested pages over the training pages of *ink*. These data corresponds with 10%, 7% and 6% of requested pages over prefetched pages, respectively.

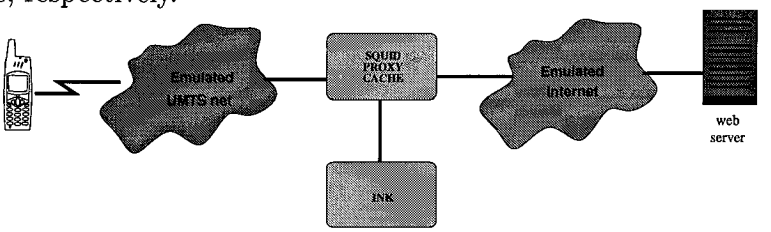

Figure 2. Measurement scenario

In figure 3 we can see the delay that a user have to wait when downloading 15 pages in different scenarios. The selected network parameters are:

	UMTS Network:	Internet
BW	230 Kbps	Variable, 56, 1000 or 2000 Kbps
RTT	200 ms	300 ms

Similar results were obtained on changing these conditions.

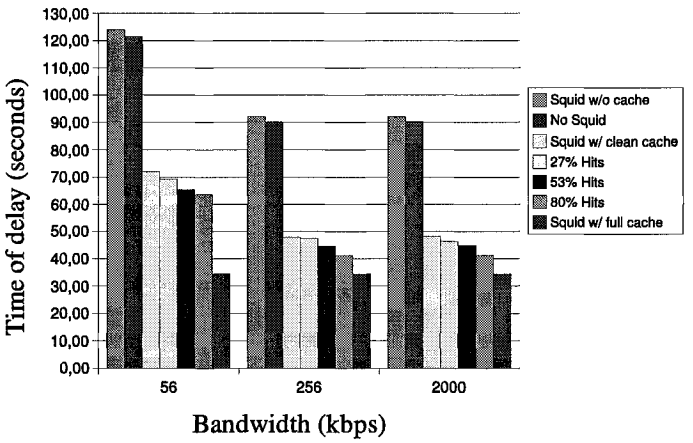

Figure 3. Delay time

Theoretically the worst case is the scenario where the proxy does not have cache. And the theoretical best case is the scenario in where all pages that the user wants to download are cached, the proxy with full cache scenario. The challenge is to approach this theoretical case as much as possible, improving the performance of simple caching without excessive use of extra bandwidth. The improvement, related to the best case, is better than 50% delay gain than the simple proxy cache scenario and 80% less that a proxy scenario. The cost of this improvement are an increase of 45% of Internet bandwidth.

8. Conclusions and further work

Today prefetching is a well known technique that seems not to scale well for large populations of clients. The interest of this technique has returned to the foreground of research with the advent of wireless networks. This paper has focused on the challenges behind the practical deployment of this service in 3G networks and has identified the most important technical issues of prefetching in this context. Furthermore, we have presented a new scenario for the practical implementation of a commercial prefetching service in 3G based on a new way of using OSA/Parlay. Finally, we have shown the feasability of this approach with a prefetching application that is being tested over a real UMTS network, including a number of new practical heuristics to improve next-link prediction.

References

[1] Jia Wang. A survey of Web caching schemes for the Internet. *ACM Computer Communication Review*, 25(9):36–46, 1999.

[2] S. Venkatesh N.J. Tuah, M. Kumar. Resource-aware speculative prefetching in wireless networks. *Wireless Networks*, 9(1), 2003. Kluwer.

[3] Savvas Gitzenis and Nicholas Bambos. Power-controlled data prefetching/caching in wireless packet networks. Infocom, 2002.

[4] Venkata N. Padmanabhan and Jeffrey C. Mogul. Using predictive prefetching to improve World-Wide Web latency. In *Proceedings of the ACM SIGCOMM '96 Conference*, Stanford University, CA, 1996.

[5] Li Fan, Pei Cao, Wei Lin, and Quinn Jacobson. Web prefetching between low-bandwidth clients and proxies: Potential and performance. In *Measurement and Modeling of Computer Systems*, pages 178–187, 1999.

[6] J. G. Cleary and I. H. Witten. Data compression using adaptive coding and partial string matching. *IEEE Transactions on Communication*, 32:396–402, 1984.

[7] Tom M. Kroeger, Darrell D. E. Long, and Jeffrey C. Mogul. Exploring the bounds of web latency reduction from caching and prefetching. In *USENIX Symposium on Internet Technologies and Systems*, 1997.

[8] Z. Jiang and L. Kleinrock. Web prefetching in a mobile environment. *IEEE Personal Communications*, 5:25–34, October 1998.

[9] Umts forum. http://www.etsi.org.

[10] Opium official site. http://www.ist-opium.org/.

[11] Parlay Group. http://www.parlay.org.

[12] Ard-Jan Moerdijk and Lucas Klostermann. *Opening the Networks with Parlay/OSA APIs*, March 2002.

[13] G. Abdulla S. Williams M. Abrams, C. R. Standridge and E. A. Fox. Caching proxies: limitations and potentials. In *4th International WWW Conference*, Boston, MA, December 1995.

[14] Azer Bestavros and Carlos Cunha. Server-initiated document dissemination for the WWW. *IEEE Data Engineering Bulletin*, 1996.

[15] Mark Crovella and Paul Barford. The network effects of prefetching. Technical Report 1997-002, 7, 1997.

[16] Carlos R. Cunha and Carlos F. B. Jaccoud. Determining WWW user's next access and its application to pre-fetching. Technical Report 1997-004, 24, 1997.

[17] Edith Cohen, Balachander Krishnamurthy, and Jennifer Rexford. Improving end-to-end performance of the web using server volumes and proxy filters. In *SIGCOMM*, pages 241–253, 1998.

[18] Tong Sau Loon and Vaduvur Bharghavan. Alleviating the latency and bandwidth problems in WWW browsing. In *USENIX Symposium on Internet Technologies and Systems*, 1997.

[19] Themistoklis Palpanas and Alberto Mendelzon. Web prefetching using partial match prediction. In *Proceedings of the 4th International Web Caching Workshop*, 1999.

[20] ink web page. http://matrix.it.uc3m.es/ opium/ink-0.1.tgz, July 2003.