2004 IEEE International Conference on Communications

Volume 1

Paris

ICC 2004

20-24 June 2004 - Paris

Gate to the Global Information Society

IEEE

IEEE COMMUNICATIONS SOCIETY

Cooperative Sponsors:

E·U·R·E·L
Convention of National Societies of Electrical Engineers of Europe

ICC GLOBECOM

see

2004 IEEE International Conference on Communications

IEEE Catalog Number 04CH37577 (softbound)
04 CH37577C (CD-ROM)

ISBN 0-7803-8533-0 (softbound)
0-7803-8534-9 (CD-ROM)

Library of Congress 81-649547

Additional copies of this publication are available from

IEEE Operations Center
P.O. Box 1331
445 Hoes Lane
Piscataway, NJ 08855-1331 USA

1-800-678-IEEE
1-732-981-1393
1-732-981-9667 (FAX)
email: customer.services@ieee.org

Smart Download on the Go: A Wireless Internet Application for Music Distribution over Heterogeneous Networks

V. Ghini, G. Pau, *Member IEEE*, M. Roccetti,

P. Salomoni, *Member IEEE* and M. Gerla, *Fellow, IEEE*

Abstract— **The maturing distributed file sharing technology implemented by Napster has first enabled the dissemination of musical content in digital forms, permitting to costumers an ubiquitous reach to stored music files from around the world. In the post-Napster era, the Apple iTunes online music service has hit a record share of 16.7% in the MP3 player market [1]. This is only the most prominent example of the success of digital music distribution based on packet network technologies. However, to the best of our knowledge, the most noteworthy aspect of the success of digital music distribution is that little about this music delivery technology is really new. To deeply change the trend of this technology business, we claim that wireless technologies must come on the scene. In particular, the digital music delivery model may take benefit by the integration of the wired Internet with a plethora of several, alternative wireless technologies, such as, for example, WiFi, WPAN and 3G. In this challenging context, we have developed a wireless Internet application designed to support the distribution of digital music to handheld devices. The main novelty of our software application amounts to its ability in providing a seamless music delivery service even in the presence of horizontal and vertical handoffs. We have taken measurements from real-world experiments that show the efficacy of the system we have developed.**

Index Terms— **Internet Wireless Applications, Inter-technologies Roaming, Music Delivery Technologies, Music Consumer Networking, Wireless Networks.**

I. Introduction

Napster has been probably the most revolutionary and unprecedented in scale phenomenon of digital music distribution over packet switched networks. In the post-Napster era, several major record labels and Internet providers launched their latest creations for delivering MP3-based music contents. Following this trend, many important networked systems have blossomed that were developed for delivering modern multimedia entertainment services (music, video and games) to wired devices on a large scale. Perhaps the latest and most famous example of widespread access to a music storehouse amounts to the *Apple iTunes Music Store*. With such a kind of system, each browser-equipped music consumer can search new songs or legendary hits he have not heard in years. Previews for any desired song may be played out for free and then downloaded in pristine digital quality at a low cost, with just one click [2].

The novelty, now, is that with a lot of users adopting Internet-enabled cellular phones and similar handheld devices, a growing demand is emerging for wireless services that allow mobile consumers to link to music contents from the large song repository represented by the Web.

From a technological standpoint, the *General Packet Radio Service* (GPRS), offering several tens of Kb/s and the third generation (3G) mobile systems, offering soon data rates up to few Mb/s, plus appropriate compression techniques, may permit to consumers the access to video/audio entertainment services to be enjoyed on smart phones and similar smaller devices. Wireless LAN/ WPAN technologies, used to provide *local* access to the Internet, are likely to be coupled with the existing widespread telecommunication infrastructures (e.g., UMTS), thus providing an opportunity to drastically improve the range, throughput and performance of music distribution services [3]. In particular, stimulating scenarios for music distribution that may take benefit by the integrated use of global/local wireless infrastructures include the following:

1) *Music Delivery over cellular networks:* a wireless entry point to music delivery networks (e.g., wireless music portals) may represent an innovative and value-added service offered by cellular carriers to those mobile costumers wishing to use their 2.5/3G-enabled cellular phones for playing out songs, as well as other enhanced multimedia objects.

2) *WLAN-based music showers:* integrated wired/wireless infrastructures may be built to support the setting up of music showers that distribute digital musical contents to authorized customers.

3) *Opportunistic music communities:* under these scenarios,

V. Ghini is with the Department of Computer Science of the University of Bologna, Mura A. Zamboni 7, Bologna, 40127 Italy (e-mail: ghini@cs.unibo.it).

G. Pau is with the Computer Science Department, UCLA, CA 90095, USA (phone: 310-206-3212; fax: 310-825-7578; e-mail: gpau@cs.ucla.edu).

M. Roccetti is with the Department of Computer Science of the University of Bologna, Mura A. Zamboni 7, Bologna, 40127 Italy (e-mail: roccetti@cs.unibo.it)..

P. Salomoni is with the Department of Computer Science of the University of Bologna, Mura A. Zamboni 7, Bologna, 40127 Italy (e-mail: salomoni@cs.unibo.it).

M. Gerla is with the Computer Science Department, UCLA, CA 90095, USA (e-mail: gerla@cs.ucla.edu).

a *music community* comprised of several users may be set up where only a few members are under the coverage of a music shower from which their favorite songs are downloaded. Those musical contents may then flow from the users who are under the direct coverage of the music kiosk to all the other community members, following a chain of mobile customers who are used as extensions of the existing wireless network (WLAN/WPAN or cellular). A prominent example of opportunistic music communities may be envisaged where a society of WLAN/WPAN-equipped drivers use their cars to move on roadways along which music showers are distributed.

However, from a technical perspective, several important problems arise when wireless technologies are integrated with the fixed Internet, with the aim of distributing musical contents stored on the Web to mobile devices.

For example, it is well known that wireless networks represent one of the most complex environments for standard transport protocols, like TCP. In fact, traditional TCP (New Reno) has been designed to interpret unexpected delays and packet losses as symptoms of network congestion. The normal TCP's reaction to delays and losses is then to trigger a back off procedure that aggressively reduces the congestion window to achieve congestion stability. Contrariwise, in a wireless environment packet losses and delays are typically signs of problems caused by mobility, such as handovers, transmission errors and temporary link outages. Hence, if mobility is mistaken for congestion, and aggressive TCP procedures are activated, then exasperating phenomena of further performance degradation may be experienced during transport-level connections [4].

Further, another relevant problem for music distribution to nomadic devices is that of unpredicted link interruptions in the midst of a long music download activity. If it is not ensured that the song download activity is not interrupted when link communication is destroyed, due to very long link outages or handoffs, then the online music download becomes an endlessly frustrating experience, and the correspondent delivery service is not appreciated by customers. Mobile clients should be able to seamlessly operate, also in a heterogeneous wireless environment where different mobile network technologies (e.g., 802.11, 3G, satellite, etc.) can alternatively offer the wireless access to the available music showers [5].

In this context, we have designed, developed and experimented with on-the-field trials a wireless Internet software application that allows users to enjoy an online music -on-demand service on wireless devices.

Simply stated, our software architecture has been designed to include a *Session Management Layer* (SML) that permits online music distribution to mobile devices with the Internet as a backplane. At the heart of our SML there is its ability to provide support to:

1) an end-to-end IP continuity without failures in the midst of a download activity due to link outages or handovers, and

2) a vertical roaming scheme for switching across alternative wireless technologies (e.g., WiFi, WPAN and cellular).

In summary, the aim of our SML is no longer one of adjusting the transport protocol (TCP) performances to match the unstable requirements of the wireless environment, but one of ensuring a successful termination of the music download activity even when: (a) the underlying link connections are damaged or disrupted due to the device mobility (*horizontal handoffs*) and/or (b) a mobile user passes through different areas with signal coverage provided by different alternative wireless technologies (*vertical handoffs*).

As explained in the following Section 2, this goal is fulfilled by freezing the download state activity at the session layer when horizontal and/or vertical handoffs are detected, thus permitting to resume the music data stream as soon as the causes that have originated the interruption of the music download activity have been extinguished. We have made efforts to assess the effects that our wireless Internet application has on the download performance of digital music content. To this aim, download time measurements have been experimentally taken that show the viability of our approach.

The reminder of this paper is organized as follows. In Section II, we discuss the design of the architecture of the wireless Internet application we have developed. Section III presents the real-world experimental scenarios we have set up to test the efficacy of our wireless application. Session IV reports on a large set of performance results we have obtained from our on-the-field trials. In Section V, some related work is discussed and contrasted with the designed goals on which our wireless application is based. Section VI concludes the paper by providing some final remarks and planning for future works.

II. System Architecture

The general architecture of our proposed wireless application is comprised of the following three software components, as detailed below [6]:

- *Mobile Clients*, i.e. music handheld devices connected through wireless network access points. This part of the wireless application supporting mobile clients has the responsibility for the subsequent activities of: (i) searching the MP3 files corresponding to the client's favorite songs over the Internet, (ii) downloading them on the handheld device, and, finally, (iii) playing them out as soon as the download has been completed. Currently, several alternative wireless networking technologies are supported, in particular the WiFi (802.11b), Bluetooth, CDMA2000 (1xRTT) and GPRS wireless technologies.
- The (IS) *Intermediate System*, this software component is hosted in an Internet server and represents the core of our application. It is in charge of managing all the communication between the handheld device and the wired Internet infrastructures. It consists of three software sub-systems, namely the *Application Gateway*, the

Discovery system and the *Download Manager*, that perform, in turn, the following functions:

1. *Wireless connectivity,* an *Application Gateway* has been developed that accepts and manages all the requests of songs arriving form the client connected to a given mobile device. The main characteristic of this software component is that of guaranteeing a reliable communication between the IS and the mobile client. In particular, the Application Gateway embodies a *Session Management Layer* (SML) that ensures a seamless IP continuity in the face of possible link outages and handoffs over the wireless channel, while allowing the client to switch across different mobile technologies in the midst of a music download activity.
2. *Resource discovery*, a *Discovery* system has been developed whose main tasks are to discover and to locate on the Web the music resources (i.e., the MP3 files) which correspond to songs that have been requested by the users.
3. *Resource download*, a *Download Manager* has been designed to carry out the activity of downloading the songs that have been identified by the Discovery system. The Download Manager incorporates a novel mechanism that is able to engage concurrently several identical copies of a given music resource, to speed up the download activity from the wired Internet to the IS. The mechanism periodically monitors the downloading performance of available replica servers and dynamically selects, at run-time, those replicas to which the client sub-requests can be sent, based on both the network congestion status and the replica servers workload [7].

- A set of *Web server replicas*. They are Web servers geographically distributed over the Internet which perform as replicated music repositories. The choice of using several replicas to store each different music resource has been taken with the aim to augment the service responsiveness. The song is downloaded contemporaneously from several replicas in order to maximize the performances and achieve load balance.

A. Search, download and play out of musical resources

As an example, a complete search/download/play-out session for music resources steps trough three different phases, and is as follows. In a first phase, a user from his/her handheld device issues to the Application Gateway a request for a given song. The Application Gateway passes this request down to the Download Manager. The Download Manager asks to the Discovery for the complete list of all the available music resources matching the request issued by the user. The Discovery performs the research of the songs required by the client using meta-description information as reported in [7]. Once this activity is completed, the query results delivered to the user. Upon receiving list of results, the user chooses one of the proposed songs. This choice activates an automatic process to download the correspondent MP3 file. Needless to say, in order for the system to work correctly, a preliminary phase has to be carried where each music repository announces the list of the songs it wishes to make available for distribution [7].

B. Providing connection continuity on the wireless link

Our wireless application has been structured based on the use of an open IP approach where the mobile device is allowed to function as other Internet connected device, and an end-to-end direct TCP/IP continuity is ensured by exploiting a TCP/IP protocol stack [8] aiming to provide a seamless internetworking scheme between the wired and the wireless segments.

Further, as one of the most crucial problems for multimedia distribution to wireless devices is that of an unexpected link interruption in the middle of a long download activity, we have augmented our wireless protocol architecture with a session layer developed on the top of the transport protocol (e.g., TCP). The aim of our session layer is to guarantee that the song download activity is not destroyed when long link outages or (horizontal/vertical) handoffs occur. It is very important to notice that the problem is ensuring a successful termination of the download activity even when the underlining connection is destroyed of definitively damaged at the transport layer due to device mobility.

We introduced an application layer on the top of the standard TCP/IP that has been designed as constructed out of two different sub-layers: (a) a *Session Layer*; this protocol layer is devoted to manage a download session which may provides users with the possibility of resuming a communication that was previously interrupted due to link problems; (b) *a Music Application Layer*; this protocol layer is in charge of supporting the different connections needed to search, download and play out songs.

It is well known that the standard implementation of the TCP protocol interprets unexpected delays and packet losses as symptoms of network congestion and reacts reducing the congestion windows to achieve fairness and stability. This approach is not suitable for wireless networks where long delays and packet loss can be triggered by channel noise and mobility [8]. To overcome the performance problems several TCP variations have been proposed [4]. However, one of the most prominent (and lesser-known) problems for music distribution to mobile device is that of an unpredicted link interruption in the midst of a long song download activity. In other words, a mobile user may enter an area of no signal coverage for a given period of time [7]. Typically, if this time period of no coverage exceeds a given threshold (e.g., a few minutes) the network gateway located in between the wired and the wireless segments assumes that a link level interruption has occurred and consequently destroys the data link connection over the wireless link and all related transport connections as well causing very serious consequences as the music download activity is definitively interrupted with no possibility to resume the data stream. This is a clear evidence of the fact that a session mechanism is needed to manage the communication interruptions that may occur at the lower levels of the communication architecture.

With this in view, the aim of our session layer is ensuring a successful termination of the download activity even when the underlying TCP connections are disrupted or damaged due to the device mobility. Stated simply, the full success of a song download activity may be really guaranteed only by a session mechanism able: (a) to *freeze* the download activity in the presence of temporary communication outages and (b) to *resume* the data transfer as soon as the signal coverage is available again.

Thus, we have designed a session management layer (SML) which is able to manage possible interruptions at the lower levels of the communication architecture due to the phenomena of either horizontal or vertical handoffs.

Even though our session mechanism has been especially designed to recover from stable damages experienced by TCP connections, we have extended our protocol architecture to include, at the transport layer, one of the TCP wireless protocols which were devised to overcome the negative aspects of performance degradation induced by the standard TCP protocol stack [4]. In particular, we have embodied in our protocol architecture the TCP Westwood (TCPW) wireless protocol [9] specifically designed for mobile wireless. In particular, TCPW provides answers to the (initial) setting problems of standard TCP. The key idea of TCPW is to use a novel scheme that estimates the congestion level on the wireless link by exploiting information derived from the TCP acknowledgment packets (ACKs) [9].

Summing up, a TCPW source performs an end-to-end estimate of the *eligible bandwidth* along a TCP connection by measuring (and low-pass filtering) the rate of returning ACKs. This estimate is then used to compute new values for both the congestion window and the slow-start threshold to be exercised during the considered TCP connection. Our choice to use TCPW, instead of other alternative wireless transport protocol, is that TCPW has the precious advantage to require modifications of the TCP congestion control mechanism only at the sender side [10][11].

In conclusion, our session layer works at the Client side and at the Application Gateway as illustrated in the following Section.

C. Implementing the Session Management Layer

Our SML works at the client side is summarized below:

At the beginning of each download session, a mechanism has been implemented that selects the most *appropriate* radio interface among all those available on the mobile device. Through this radio interface (and the corresponding IP address) a TCP connection is established with the Application Gateway.

When the mobile device opens a download session with the Application Gateway through this TCP connection, the Application Gateway assigns a unique identifier to this new session and sends it to the client. The session state, in essence, a pointer to the last byte received and the session identifier, are saved at the client side and used to restart interrupted sessions.

After a download session interruption, an automatic mechanism is started that tries to establish a new TCP connection with the Application Gateway by concurrently exploiting all the radio interfaces which are available on the mobile device. With the first radio interface responding positively to this attempt, a new TCP connection is opened that permits to resume the interrupted download session with the Application Gateway; the session is terminated when the client detects that the download is eventually successfully completed. [7][8]

III. An Experimental Assessment

In this Section we present an experimental study we have conducted to assess the efficacy of our music on demand application. The main motivation behind our experimental assessment was that of investigating the quality and the performance of Web/mobile client music download sessions supported by our wireless application. During the period August 1- August 30 2003, we conducted 500 experiments consisting in the download of a set of different songs (MP3 files). In the next two Subsections we provide detailed information concerning, respectively, the experimental scenario where our trials were carried out (Subsection A) and the empirical result we gathered with our on-the-field trials (Subsection B).

A. Experimental Scenario

With the aim to provide several different replicas for the songs to be distributed, we exploited four different Web servers providing the same set of 20 different songs. The four different replica servers were respectively located in USA (Washington), Brazil (San Paolo), Japan (Tokyo), Italy (Cesena). They were all based on the Linux Debian operating System, and exploited an Apache HTTP server.

The Intermediate System was running on a Pentium III machine (800 MHz, 128 MB RAM) equipped with the FreeBSD v. 4.7 operating system, and was located in the LAN (100 Mb/s) of the Network Research Laboratory in the Department of Computer Science at UCLA (USA).

Further, the wireless device, on which the client of our application was running, was installed on a HP iPAQ H5450 PDA (400 MHz, 64 MB RAM) equipped with the Windows CE 3.0 operating system.

We conducted our experiments using four different radio technologies installed on the wireless device. We exploited the four following radio interfaces: WLAN 802.11b (infrastructure-based), WLAN 802.11b (ad hoc), CDMA-2000 (1xRTT), cellular GPRS. In particular, we used the wireless network adaptors with the following characteristics: (i) Infrastructure-based WLAN: iPAQ embedded interface working at 11 Mb/s. (ii) Ad hoc WLAN: external interface (PCMCIA), Orinoco Gold produced by Lucent Technologies, working at 2 Mb/s in ad hoc mode. (iii) 3G CDMA2000 (1xRTT): external interface (PCMCIA), Sierra Wireless AirCard 555, nominal data rate of 144 Kb/s. (iv) GPRS: external interface (PCMCIA), Sierra Wireless AirCard 750 (triband 1900, 1800, 900 MHz), working at the data rate of 56Kb/s.

It is worth mentioning here that the two different configurations (namely, with infrastructure and ad hoc) exploited to test the WLAN 802.11b amounted to the two following topologies: (a) infrastructure-based WLAN (11 Mb/s), here the wireless PDA was interconnected to the IS through the access point (and the default IP router) connecting the WLAN and the wired LAN segments of the Network Research Laboratory at UCLA; (b) ad hoc WLAN (1 Mb/s), in this case, the wireless PDA was directly connected, through its radio interface, to the machine hosting the IS, based on an ad hoc configuration (1 Mb/s).

To provide the reader with an approximate knowledge of the transmission times experienced on the considered Internet links, it is important to notice that the average measurements, obtained with the *ping* routine, of the Round Trip Times between the IS, located at UCLA, and the four different Web servers, i.e., Washington, San Paolo, Tokyo and Cesena were 79, 189, 132 and 157 ms, respectively.

Instead, depending on the adopted wireless network technologies, the transmission times and the percentages of packet losses experienced over the wireless channels connecting the IS with the mobile client, and measured at the IP level, ranged in the following intervals:

- infrastructure-based WLAN → 3-6 ms, 1-15 %;
- ad hoc WLAN → 5-53 ms, 1-20 %;
- CDMA2000 - 1xRTT (144 Kb/s) → 309-930 ms, 1-12 %;
- GPRS (56 Kb/s) → 550-610 ms, 1-18 %.

B. Empirical Results

This Subsection reports on a large set of results obtained within the experimental scenario we have described above. In particular, we will present the measurements of the download times we obtained for the distribution of MP3 songs from the Web to the mobile clients for all the three following download situations: (a) The entire song download process is conducted by using the same wireless network technology, without any kind of interruptions. (b) The song download process is split into a few different subsequent phases (typically, 2 or 3) which are separated each from other due to horizontal handoffs that cause the download interruption. (c) The song download process is split into two different phases which are separated each from other due to vertical handoffs that cause the download interruption E.g., we have a transition from WLAN to CDMA2000 (1xRTT).

As to the results we gathered in the experimental scenario where the same kind of radio access technology was used, we have collected the obtained empirical measurements in four different Tables (Tables I-IV). Respectively, each of those four Tables records the song download times (expressed in seconds) for one of the following situations: (a) The user who downloads songs is still and does not suffer from any download interruption (Table I). (b)The user who downloads songs is in motion and does not experience from any download interruption (Table II). (c) The user downloading songs is in motion and experiences exactly one download interruption (Table III). (d) The user downloading songs is in motion and experiences exactly two download interruptions (Table IV).

TABLE I
DOWNLOAD TIMES (SECONDS), STILL USER, 0 INTERRUPTIONS

	TCP Standard	TCPW
WLAN infrastructure	38	34
WLAN Ad Hoc	46	48
GPRS	481	480
1xRTT	369	325

The first general consideration related to the above four Tables is concerned with the following fact: the larger the number of interruptions, the larger the amount of time needed to download the song. Based on this consideration it is clear that a user in motion typically experiences larger download times w.r.t. the case when the user is still.

TABLE II
DOWNLOAD TIMES (SECONDS), USER IN MOTION, 0 INTERRUPTIONS

	TCP Reno	TCPW
WLAN infrastructure	81	77
WLAN Ad Hoc	91	94
GPRS	632	630
1xRTT	414	360

This is an expected result which explains the smaller download times obtained with the WiFi technologies w.r.t. the cellular infrastructure (GPRS-1xRTT). As a third comment, we wish to observe that WiFI technologies are typically more robust in the sense that, within this technology, it is rare the case when one or more interruptions occur. This justifies the fact that Table IV (2 interruptions) does not report data concerned with WiFi technology. A second consideration touches upon the fact that smaller download time maybe obtained with radio access technologies able to guarantee a larger bandwidth.

TABLE III
DOWNLOAD TIMES (SECONDS), USER IN MOTION, 1 INTERRUPTION

	TCP Reno	TCPW
WLAN infrastructure	109	105
WLAN Ad Hoc	121	131
GPRS	697	690
1xRTT	462	415

The final, and perhaps most important, consideration regards the fact that TCP Westwood (TCPW) outperforms TCP Reno in most cases, in particular with cellular technologies. However, TCPW does not clearly outperform the traditional TCP in the experiments conducted with the WiFi technology. As shown in the Tables, in fact, in some cases with the WiFi technology, the download times obtained with TCP Reno are smaller than those obtained with TCPW.

TABLE IV
DOWNLOAD TIMES (SECONDS), USER IN MOTION, 2 INTERRUPTIONS

	TCP Reno	TCPW
GPRS	749	761
1xRTT	515	502

TABLE V
DOWNLOAD TIMES (SECONDS), WIFI, REMOTE IS

	TCP Reno	TCPW
still, 0 interruptions	63	50
in motion, 0 interruptions	74	69
in motion, 1 interruptions	125	118
in motion, 2 interruptions	178	155

Two of the authors of this paper, who are also co-designers of

TCPW, suggest as motivation behind this phenomenon the fact that TCPW has been designed to efficiently transport data when a large bandwidth x delay product is experienced. The experimental scenario we have built with the WiFi technology, indeed, has such a product quite low, due to the fact that our IS has been placed quite near to the wireless client (1 hop). To confirm this hypothesis we have developed an additional experiment where our IS was placed far from the mobile client. In particular during this experiment our IS was located in Cesena, while the mobile client downloaded song moving in UCLA.

The measurements taken with this additional experiment are reported in Table V and confirm the hypothesis suggested by our colleagues. As shown in this Table, TCPW outperforms TCP Reno even with the WiFi technology when the source of multimedia information (i.e., IS) is located far from the destination (i.e., mobile client).

TABLE VI
VERTICAL HANDOFFS

Experiment #1	ADHOC - 51 s, 24% interruption - 2 s GPRS - 236 s, 30% interruption - 1 s WLAN - 36 s, 46% download time=326 s
Experiment #2	WLAN - 31 s, 37% interruption - 1 s ADHOC - 81 s, 33% interruption - 1 s ADHOC - 27 s, 0% interruption - 4 s 1xRTT - 110 s, 30% download time=255 s
Experiment #3	1xRTT - 144 s, 34% interruption - 18 s WLAN - 56 s, 41% interruption - 18 s ADHOC - 26 s, 25% download time=243 s
Experiment #4	ADHOC - 51 s, 24% interruption - 2 s GPRS - 236 s, 30% interruption - 1 s WLAN - 36 s, 46% download time=326 s
Experiment #5	WLAN - 38 s, 52% interruption - 1 s ADHOC - 78 s, 35% interruption - 2 s GPRS - 99 s, 13% download time=218 s
Experiment #6	GPRS - 38 s, 3% interruption - 1 s WLAN - 72 s, 44% interruption - 2 s GPRS - 104 s, 0% interruption - 2 s ADHOC 38 s, 53% download time=257 s

As the final consideration, we wish to mention that session layer mechanism we have adopted ensures that song download may be carried out even if several link interruption occur due to the switch of radio access technology. In simple words our wireless application is able to maintain active the download session even in the case when long signal interruption are caused by vertical handoffs. Further, our wireless application is able to freeze the status of the download session when a radio access technology is no longer able to transport data, while switching to the best available alternative technology. (Experiment #2 and #6). It is very important to mention that without the use of our session layer mechanism the download activity should be restarted from scratch with each new interruption. As to the experiments conducted to test our wireless application in the case of vertical handoffs, we may report here the fact that in all those experiments (100%) our mechanism was able to guarantee the completion of the download session even in the presence of several radio technologies switches (and correspondent vertical handoffs). As significant examples of our experimental trials, we report below in Table VI the traces of six different download sessions conducted switching through different radio access technologies (e.g., WiFi and GPRS/1xRTT). Out of those six traces, three are concerned with the experiments conducted switching from WiFi (infrastructure-based and ad hoc) to GPRS, while the other three were conducted switching from WiFi to 1xRTT. In particular, for each given trace, the following information are reported in the Table: (a) Radio access technology used to download the song before interruption; download times (expressed in seconds) and correspondent amount of downloaded song (percentage) before interruption. (b) Duration of the interruption (in seconds). (c) Total download time (in seconds).

IV. CONCLUSION

We have developed an Internet wireless application able to distribute the music from the Web to mobile clients. The main characteristics of wireless application are: (i) its ability to provide support to mobile clients connected to the Web through different radio access technologies (i.e., WiFi, GPRS, 1xRTT), (ii) its ability to provide song download continuity even in the presence of horizontal and vertical handoffs. Summing up, we have designed our wireless application to incorporate sophisticated software mechanisms that guarantee to the mobile user always the best download alternative.

Measurements have been presented that confirmed the efficacy of our approach.

REFERENCES

[1] J.Mc Hugh, "Why Wi-Fi is a good business" in *Wired*, Sept. 2003, pp. 25–26.

[2] G. Prem Premkumar, "Alternate distribution strategies for digital music", *Communications of the ACM*, Sept. 2003, Volume 46 Issue 9, pp. 89-95.

[3] J. De Vriendt et al., "Mobile Network Evolution: A Revolution on the Move," *IEEE Commununication Magazine*, vol. 40, no. 4, Apr. 2002, pp. 104–11..

[4] H. Balakrishnan,V, N. Padmanabhan, S. Seshan, R H. Katz, "A Comparison of Mechanisms for Improving TCP Performance over Wireless Links", *IEEE/ACM Transactions on Networking*, vol. 5, no. 6, June 1997, pp. 756-769.

[5] C. Hesselman, H. Eertink, A. Peddemors, "Multimedia QoS Adaptation for Inter-tech Roaming", Proc. Sixth IEEE Symposium on Computers

and Communications (ISCC'01), July 03 - 05, 2001, Hammamet, Tunisia, pp 554-561.
[6] M. Roccetti, P. Salomoni, V. Ghini & S. Ferretti, "Bringing the Wireless Internet to UMTS Devices: A Case Study with Music Distribution", Multimedia Tools and Applications, an International Journal, Kluwer Academic Publishers, accepted for publication, December 2002.
[7] V. Ghini, F. Panzieri, M. Roccetti, "Client-Centered Load Distribution: A Mechanism for Constructing Responsive Web Services", Proc. 34th International Conference on System Sciences, Maui, Hawaii, USA, January 2001.
[8] M. Roccetti, P. Salomoni, V. Ghini, S. Ferretti & S. Cacciaguerra, "Delivering Music over the Wireless Internet: from Song Distribution to Interactive Karaoke on UMTS Devices", In Wireless Internet Handbook: Technologies, Standards and Applications, (B. Furht, M. Ilyas eds.), CRC Press, Boca Raton (USA), March 2003, 537-565.
[9] R. Wang, M. Valla, M. Y. Sanadidi, M. Gerla, "Adaptive Bandwidth Share Estimation in TCP Westwood", In Proc. IEEE Globecom 2002, Taipei, Taiwan, R.O.C., November 17-21, 2002
[10] S. Mascolo, A. Grieco, G. Pau, M. Gerla, C. Casetti, "End-to-End Bandwidth Estimation in TCP to Improve Wireless Link Utilization", IEEE European Wireless Conference, Florence, Italy, Feb. 2002
[11] M. Gerla, B. K. F. Ng, M. Y. Sanadidi, M. Valla, R. Wang " TCP Westwood with Adaptive Bandwidth Estimation to Improve Efficiency/Friendliness Tradeoffs ", To appear in Computer Communication Journal