An IMS-Based Middleware Solution for Energy-Efficient and Cost-Effective Mobile Multimedia Services

Paolo Bellavista, Antonio Corradi, and Luca Foschini

Dipartimento di Elettronica Informatica e Sistemistica - Università di Bologna
Viale Risorgimento, 2 – 40136 Bologna – Italy
Ph.: +39-051-2093001; Fax: +39-051-2093073
`{paolo.bellavista,antonio.corradi,luca.foschini}@unibo.it`

Abstract. Mobile multimedia services have recently become of extreme industrial relevance due to the advances in both wireless client devices and multimedia communications. That has motivated important standardization efforts, such as the IP Multimedia Subsystem (IMS) to support session control, mobility, and interoperability in all-IP next generation networks. Notwithstanding the central role of IMS in novel mobile multimedia, the potential of IMS-based service composition for the development of new classes of ready-to-use, energy-efficient, and cost-effective services is still widely unexplored. The paper proposes an original solution for the dynamic and standard-compliant redirection of incoming voice calls towards WiFi-equipped smart phones. The primary design guideline is to reduce energy consumption and service costs for the final user by automatically switching from the 3G to the WiFi infrastructure whenever possible. The proposal is fully compliant with the IMS standard and exploits the recently released IMS presence service to update device location and current communication opportunities. The reported experimental results point out that our solution, in a simple way and with full compliance with state-of-the-art industrially-accepted standards, can significantly increase battery lifetime without negative effects on call initiation delay.

Keywords: Mobile Multimedia, Energy Efficiency, Middleware, Standards, IP Multimedia Subsystem (IMS).

1 Introduction

Nowadays it starts to be more and more common that a number of mobile users require seamless access to multimedia services, such as audio/video voice calls, while they move across the available and possibly heterogeneous wireless infrastructures, spanning from IEEE 802.11 (WiFi) and Bluetooth (BT) to cellular 3G. Despite the great potential stemming from the possibility to integrate heterogeneous wireless networks, one of the most challenging issues in supporting mobile multimedia applications is still the reduction of energy consumption at battery-powered clients. Energy is a precious resource for portable devices and wireless interface cards are

C. Giannelli (Ed.): Mobilware, LNICST 0007, pp. 280–294, 2009.

widely recognized to be one of the most relevant components that contribute to drain battery power, with a relatively greater role as device size decreases [1].

Of course, power consumption also depends on the wireless technology of exploited interface cards and unfortunately widespread wireless technologies, such as WiFi, despite their low per-bit monetary transmission cost, have a high per-bit energy transmission cost [2], [3]. To overcome that problem, some seminal research activities have started to propose the usage of low-power-consumption wireless technologies, even if with high per-bit transmission cost, e.g., cellular 3G, to opportunistically control and switch on other wireless interfaces with reduced economic costs only when needed, e.g., triggered by incoming calls [4], [5]. However, notwithstanding their potential, none of the above power management solutions have been deployed nor penetrated into the mobile multimedia market primarily because of the lack of widely accepted session control standards.

A large group of standardization bodies has recently defined the IP Multimedia Subsystem (IMS) [6]. IMS defines an overlay architecture for session control in all-IP next generation networks to obtain openness and interoperability by adopting an application-layer approach, mainly via the Session Initiation Protocol (SIP). Some research activities have already exploited IMS to dynamically reconfigure ongoing working sessions when wireless access technology change, i.e., handoff management, but to the best of our knowledge none of them considered the possibility to use IMS to deliver highly innovative, interoperable, and ready-to-market power management solutions.

The paper tackles the above issues by proposing a novel solution with three original core properties. First, it exploits terminal-based decentralized monitoring to update client wireless communication context (user/device profile, device location, available wireless access infrastructures, …). Second, it uses such awareness to jointly save energy and reduce economic cost of transmission by automatically switching on high-consumption and low-cost wireless interfaces only for the duration of service sessions. Third, it is fully compliant with the IMS standard: the diffusion of context updates exploits the recently standardized IMS-based Presence Service (PS) and our power management facility has been implemented as a novel IMS application server, part of our wider IMS-compliant Handoff Management Application Server (IHMAS) infrastructure. In particular, our power management facility, originally presented here, integrates with existing open-source IMS implementation platforms and tools such as the Open IMS Core, the Open SIPS, and the UCT Client. The facility is publicly available for wireless practitioners and the IMS community as an external IMS-compliant service[1] [7], [8], [9].

2 Background and Related Work

The provisioning of interoperable multimedia session control over heterogeneous wireless communication infrastructures is still a challenging issue. To tackle the problem, a large group of standardization entities that range from the 3rd Generation

---

[1] Additional information, experimental results, and the prototype code of the IHMAS infrastructure are available at: `http://lia.deis.unibo.it/Research/IHMAS/`

Partnership Project (3GPP) and 3GPP2 to Internet Engineering Task Force (IETF) and Open Mobile Alliance (OMA) has recently defined IMS [6]. This section briefly introduces the IMS architecture and the IMS-based PS, which are the crucial technical elements to fully understand our proposal; then, the section overviews the existing related proposals in the literature.

2.1 Background about IMS

IMS Standard. IMS mainly focuses on session control: it offers several facilities to allow the creation, modification, and termination of service sessions for an open set of potentially highly integrated and all-IP multimedia services (instant messaging, Video on Demand – VoD –, Voice over IP – VoIP –, …) independently of underlying data-link layer technologies and transport protocols.

The dynamic renegotiation and rebinding of multimedia sessions is one of the core session control activities in IMS, in order to react to the substantial and abrupt changes of the characteristics of wireless provisioning environments that may occur at runtime, e.g., when a Mobile Node (MN) disconnects from one access point and re-connects to a new one. To that purpose, IMS proposes a framework and exploits the Session Description Protocol (SDP) for endpoint localization/rebinding and for decentralized proxy-based session management.

Current IMS specifications do not include power management; however, the IMS standard facilitates the introduction of new services/extensions [6]. In the following, we rapidly introduce the three main components of the IMS infrastructure that directly participate in our power management solution, which is the central scope of the paper.

The first component is the IMS Client, which controls session setup and media transport by implementing all SIP extensions specified by IETF and 3GPP IMS-related standards. Any session is setup between two IMS clients. Here, without losing any generality and for the sake of presentation clarity, we always consider a fixed Correspondent Node (CN) as the originating SIP endpoint and MN as the terminating one. The second IMS component is the Application Server (AS), which allows the introduction of new IMS services and extensions. AS has full control over traversing SIP dialogs. The third one, the Serving-Call Session Control Function (S-CSCF), is the most important session control entity. S-CSCF authenticates and registers IMS clients to the IMS infrastructure; then, depending on filters/triggers specified by client profiles, S-CSCF may either route incoming SIP messages directly to their receiver or forward them to AS [6]. Let us stress that given its ability to change SIP message headers, S-CSCF can extend MN-to-CN session signaling paths through the interposition of convenient ASs. Fig. 1 shows the deployment of all main components presented in this and in the following subsections.

Other primary IMS components, such as Proxy-/Interrogating-CSCF, Home Subscriber Server (HSS), and the standard Internet Dynamic Host Configuration Protocol (DHCP) are not overviewed because out of the scope of this paper. For a comprehensive introduction to the whole IMS architecture, please refer to [6].

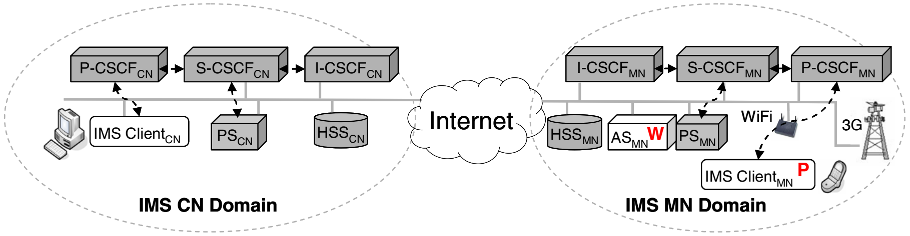

Fig. 1. IMS distributed architecture at Correspondent Node (*CN*) and Mobile Node (*MN*) domains. The core IMS components are: Home Subscriber Server (*HSS*); Proxy-Call Session Control Function (*P-CSCF*); Session-Call Session Control Function (*S-CSCF*); Interrogating-Call Session Control Function (*I-CSCF*); and Application Server (*AS*). The figure shows also IMS-based presence service components: Presence Server (*PS*); Watcher (*we use the red label W to tag an entity acting as watcher*); and Presentity (*we use the red label P to tag an entity acting as presentity*).

IMS-based Presence Service. Presence is a well-known concept in the traditional Internet and widely used in applications such as Instant Messaging or multiparty games [10]: presence permits users and hw/sw components, called presentities, to convey their ability and willingness to communicate with watchers. In order to receive PS publish/update messages from presentities, i.e., presence notifications, watchers subscribe to presence servers that act as intermediaries in any PS-related communication between presentities and watchers.

The concept of presence has been recently widened to include any context information useful to adapt service provisioning to the current state of the execution environment in a personalized way. That has made PS play a crucial role in several mobile services nowadays. For instance, instant messaging exploits the PS context information about one user's online/offline/busy status to check whether she is reachable, while voice/video conferencing uses PS context to adapt sessions depending on profiles of currently exploited devices and wireless interfaces. By focusing on power management, PS may be used to periodically update context information about MNs, including their battery levels, wireless interfaces, and available wireless access infrastructures at their current locations.

Recent IMS standardization efforts recognize PS as a core support facility for any novel mobility-enabled service [11], [12], [13]. The core IMS-based PS component is the presence server, which accepts and stores PS subscriptions from watchers, and notifies publish messages from presentities to registered watchers. The other two primary PS entities are presentities and watchers. More precisely, the presence user agent is the entity that provides PS information about a presentity (for the sake of presentation simplicity, in the following we use the single presentity term to refer both). In general, IMS clients and ASs may act as presentities or watchers depending on specific service requirements, as shown also in Fig. 1.

In addition, PS defines other components, such as the PS network agent to collect/combine different information about a presentity, and specifies protocols, such as the XML Configuration Access Protocol (XCAP), to manipulate PS-related management data (subscription authorization policies, resource lists, …). For additional details about IMS and its PS, please refer to [11], [12], [13].

2.2 Related Work

As widely agreed [3], the best way to achieve energy saving is to take advantage from the inactivity/idle time intervals of wireless network interfaces. When inactive, the interface can be put in a low power-consuming state, either the sleep mode or the off state. By following this general design principle, various energy-saving solutions have been proposed for devices equipped with single/multiple wireless interface/s.

For single-interface devices, the very crucial point for energy saving is traffic shaping. To reduce the active (on state) time intervals of the MN wireless interface, those solutions aim to schedule transmissions from/to MNs in data bursts of short duration, periodically sent at the highest rate allowed by the wireless link. For instance, by focusing on WiFi which is a notable example of energy-consumptive wireless technology, several traffic shaping solutions have been proposed both at the lower datalink OSI protocol stack layer, e.g., IEEE 802.11 Power-Saving Mode and its evolutions, and at the application layer, by tackling different services, spanning from Web browsing to multimedia mobile applications [2], [3], [14].

Energy-saving solutions for multi-interface devices, instead, tend to use an out-of-band low-energy communication channel to switch on more energy-consumptive wireless interfaces only when needed for transmission. [4] and [5] exploit an always-on low-consumption interface to send context data about MN wireless access locality back to the network in order to enable network-controlled handoff decisions aimed to save energy at MN. More recently, similar energy-saving techniques have been applied also to Wireless Sensor Networks (WSNs); ultra-low power receivers for out-of-band wake-up signaling are currently under study and development [15].

Other work related to our proposal includes ongoing research efforts on IMS exploitation and IMS-based handoff. Application-layer IMS-based handoff management has its roots on SIP-based mobility management first proposed by Schulzrinne [16]. Thereafter, a number of SIP-based research efforts tackled session adaptation at different protocol stack layers [17], [18]. More recently, some support solutions are starting to face specific IMS-related handoff issues. For instance, Intelligent Network-Seamless Mobility Access (IN-SMA) proposes a new AS, called Mobility Application Server, for IMS-based 3G-to-WiFi handoff management [19]. By focusing on industrial standardization efforts, 3GPP Voice Call Continuity (VCC) is a recent IMS-based specification that tackles audio streaming continuity during inter-technology handoff [20]. However, differently from IHMAS, those and other similar IMS-based handoff solutions focus mainly on flow transfer and, to the best of our knowledge, none of them specifically address power management issues.

3 Power Management in Our IHMAS Infrastructure

In general, we claim the need for application-level supports to ease the design and implementation of mobile multimedia services [21]. By focusing on IMS-based power management, our IMS-compliant Handoff Management Application Server (IHMAS) proposal significantly differs from state-of-the-art work by combining several original technical aspects:

- IHMAS is the first proposal in the field that explores the use of IMS to realize advanced energy-efficient and cost-effective services. Power management ASs, interposed in the CN-to-MN session control path, control MNs by switching on the wireless interfaces with high consumption and low transmission cost, e.g., WiFi, only for the needed time slots, e.g., only for the duration of calls.
- IHMAS is context-aware. It monitors context changes at MNs and exploits context change notifications to update power management ASs with current MN conditions (user profile, wireless communications availability, battery level, …), thus enabling prompt management operations. To further reduce energy consumption, context updates are sent over communications channels with low consumption and high transmission cost, e.g., cellular 3G.
- IHMAS is a real and ready-to-deploy IMS-based solution. It exploits IMS-based PS for context update notification: ASs act as watchers and IMS clients as presentities. The implemented power management components can be easily deployed and installed over existing IMS networks, without requiring any modification to existing standards and off-the-shelf equipment.

To better understand the IHMAS original proposal for power management, let us overview first the main IHMAS components. The IHMAS architecture consists of two main components that interwork to handle the two core management functions: context monitoring and session control for power management. The former is realized by the Context Monitor (CM) that extends the IMS Client with lightweight and completely decentralized context monitoring via only local access to client wireless devices. The latter involves the AS for Power Management (ASPM), deployed at the MN home network, that participates to session initiation signaling to implement our IMS energy-saving optimization. In addition, IHMAS exploits the standard IMS-based PS to facilitate the interaction between CM and ASPM. ASPM registers to PS to receive context updates published by CM. Fig. 2 depicts the main IHMAS components and their interactions.

By delving into finer details, CM periodically monitors main context parameters, especially wireless infrastructure availability and battery level at MN. CM execution imposes very limited overhead since all monitoring data are local and data gathering periods are dynamically lengthened/shortened as monitored variations decrease/increase. If monitored context values overcome monitoring threshold values, CM notifies context changes to ASPM by using low-consumption and always-on wireless interfaces, i.e., typically 3G cellular (step 1-3 in Fig. 2). ASPM maintains the current state of its corresponding IMS clients (including any possible information about connectivity status and IP configuration) and consumes all call initiation SIP messages that are automatically re-routed by MN S-CSCF (not shown here, see also Fig. 1). When receiving an INVITE message, ASPM promptly activates our novel power management SIP-based procedure to redirect incoming call session signaling procedure to the wireless interfaces with high consumption and low transmission cost, if available (steps 4-6). After that, multimedia data transmissions are established over switched-on wireless interfaces (step 7). When the call terminates the high-consumption interface is switched off and MN re-registers itself to IMS over its always-on wireless card.

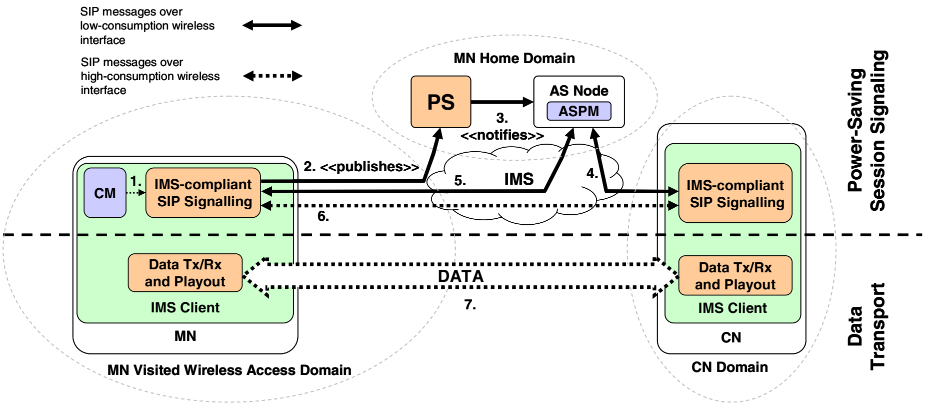

Fig. 2. The IHMAS architecture

Let us stress that the IHMAS distributed architecture achieves energy-saving improvements with minimum overhead. ASPM is interposed in CN-to-MN session signaling path only during the initial session initiation phase; successive interactions between CN and MN go on in an end-to-end way, as usual.

4 IHMAS Power Management Facility: Protocol Design and Implementation Insights

IHMAS can support several types of mobile multimedia services. For our design/implementation work on power management, we decided to focus on VoIP audio service for two main reasons: on the one hand, for its strict requirements in terms of session initiation time and, on the other hand, because it is one of the most widespread IMS services supported by most off-the-shelf IMS clients and devices. Moreover, our energy-saving technique has been tailored to WiFi given its wide diffusion and high energy consumption values, but remains valid and can be applied to other wireless technologies such as Bluetooth. This section presents our novel IMS-based energy-saving protocol and gives an overview of the most relevant implementation insights about ASPM, IMS client, and CM.

4.1 IHMAS Session Signaling for Power Management

Our original power management process consists of two main session signaling phases. The first one enables context change notifications between MN and ASPM. The second one is the core signaling function for power management that is in charge of switching on target MN interfaces and of redirecting incoming calls. Fig. 3 depicts those two phases. Continuous black lines represent the IMS session signaling protocol, dashed black lines are our original message flow extensions, and dotted black lines represent local events/calls (e.g., context change event at MN). Grey IMS components are out of IHMAS scope, while white ones are the core IHMAS components.

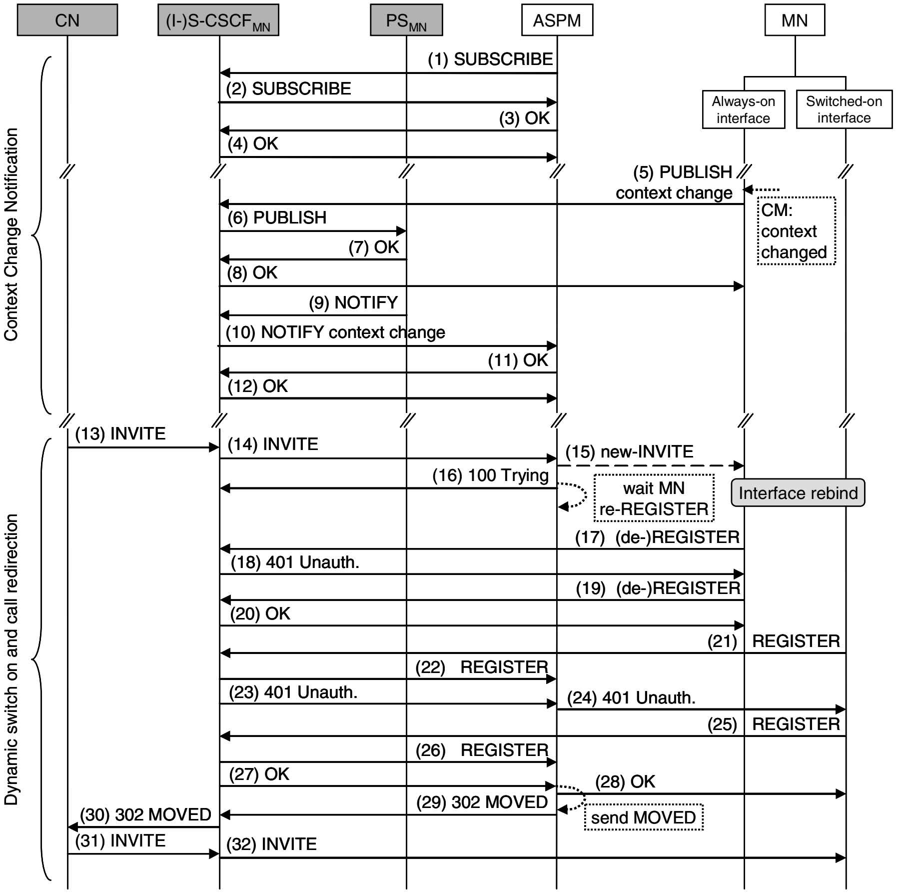

Fig. 3. The IHMAS power management protocol. For sake of clarity we omit CN P-/I-/S-CSCF and MN P-CSCF in the figure; other standard signaling messages, not central to the scope of this paper, are not reported as well, e.g., the NOTIFY message by PS after subscription.

At its activation, ASPM subscribes to PS$_{MN}$ to receive notifications for publish events emitted by all the MNs that it is serving; served MNs are maintained in a list that can be updated through a Web interface (steps 1-4). Afterwards, whenever CM perceives a context change at the MN, IMS client sends a PUBLISH message, with MAC/IP addresses and Received Signal Strength Indication (RSSI) values for all available wireless interfaces and battery level of the MN, which is notified by PS$_{MN}$ to ASPM (steps 5-12). Those context data permit ASPM to decide the wireless interface to switch on at call arrival time. In addition, S-CSCF$_{MN}$ forwards to ASPM any INVITE message over the always-on wireless interface and any registration message over switched-on wireless interface, as indicated by IMS triggers specified for MN. Power management begins when ASPM receives the INVITE message from CN (steps 13-14). Thanks to its awareness of MN wireless communication context, ASPM can promptly modify the received INVITE message by adding a line in the

SDP session description that indicates to MN the target wireless interface to switch on (step 15). Then, ASPM sends a TRYING message back to CN to signal that session initiation is ongoing (step16) and waits until MN concludes re-registration operations, that ASPM identifies from the last OK message forwarded by S-CSCF$_{MN}$ according to IMS triggers (steps 17-27). After that, ASPM forwards the OK message to MN and sends the MOVED message to CN to redirect the incoming call to the switched-on wireless interface (steps 28-30). The MOVED message reception at CN triggers a new INVITE message over the switched-on wireless interface and this terminates session initiation (steps 31-32). When the call terminates, after BYE message exchange, MN re-registers to the always-on wireless interface (not shown in Fig. 3).

The proposed protocol is fully IMS-compliant. 302 MOVED call re-direction is supported by the vast majority of IMS clients and optional SDP fields are ignored by default by all IMS entities. In addition, power management only requires very limited modification at IMS client, as detailed in the next section. Moreover, let us note that in the current version of our prototype, CM obtains wireless interface context data (MAC and IP addresses, RSSIs, …) by periodically switching-on all wireless interfaces at MN. Even if already optimized by adaptively modifying the monitoring period, that operation is still energy-consuming. To further reduce battery consumption, CM could obtain and update only the context about the always-on wireless interface, i.e., MAC addresses and RSSI values of all cellular base stations in MN visibility, without any additional energy cost. ASPM would use notifications to keep track of MN location and to proactively obtain, from an external forecast mobile connectivity service, detailed information and security credentials to pass to MN in the INVITE message. Those new services are currently under analysis and development by both academia and industry, and several directory services are already available in the Internet to locate WiFi hotspots [22].

4.2 Implementation Insights

To grant wide interoperability and to facilitate deployment, we have based the development of our power management solution on the currently available standard technologies for next generation IMS-based mobile multimedia services. Hence, for session signaling we employed the OpenIMSCore, an IMS platform that is fully compliant with 3GPP IMS specifications [6], [8]. OpenIMSCore provides all the basic components of the IMS infrastructure, e.g., P-/I-/S-CSCFs and HSS, and a framework to support and manage new ASs. ASPM has been implemented in Java by exploiting the portable Java API for Integrated Networks (JAIN) SIP implementation by the National Institute of Standards and Technology [23]. CM has been implemented by using standard Linux tools for wireless interface query and designed to integrate with the open-source University of Cape Town (UCT) IMS Client that we significantly modified to support our power management protocol. Finally, we employed OpenSIPS for the PS server [9].

Fig. 4-a shows the **`processInvite`** Java method implemented by ASPM to build the new INVITE message. First, it extracts the SDP part of the message, then invokes the **`addPowerManagementAttribute`** method that chooses the wireless interface to switch on, adds our power management optional field at the end of SDP

description, and finally recomposes and forwards the INVITE message to MN by invoking the **`sendRequest`** method on the local JAIN SIP stack. The result is the insertion of the optional "**`a=switchoninterface`**" attribute indicating MAC and IP addresses of the wireless interface to switch on as reported in Fig. 4-b; we enclose the SDP description within the red box and our new field is highlighted as bold text.

(a)

```
private void processInvite(Request request) {
  SessionDescription sdp =
    sipUtils.getSessionDescription(request);
    sd=addPowerManagementAttribute(request, sd);
    request.removeContent();
    try {
      request.setContent(sd,
      headerFactory.
        createContentTypeHeader("application","sdp"));
    } catch (ParseException e) { e.printStackTrace(); }
    try {
      sipProvider.sendRequest(request);
    } catch (SipException e) { e.printStackTrace(); }
}
```

(b)

```
INVITE sip:alice@open-ims.test SIP/2.0
Record-Route: <sip:mt@scscf.open-ims.test:6060;lr>,
<sip:mo@scscf.open-ims.test:6060;lr>,
<sip:mo@pcscf.open-ims.test:4060;lr>
From: "Bob" <sip:bob@open-ims.test>;tag=1262340263
To: <sip:alice@open-ims.test>
Call-ID: 885605810@192.168.3.11
CSeq: 20 INVITE
Contact: <sip:bob@192.168.3.11:5062>
…
Content-Type: application/sdp
Content-Length: 414

v=0
o=- 0 0 IN IP4 192.168.3.11
s=IMS Call
c=IN IP4 192.168.3.11
t=0 0
m=audio 10281 RTP/AVP 3 0 14 101
b=AS:64
a=rtpmap:3 GSM/8000
a=rtpmap:0 PCMU/8000
a=rtpmap:14 MPA/90000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-11
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=switchoninterface:00:04:23:5E:48:DE 192.168.125.2
```

Fig. 4. ASPM implementation insights: new INVITE message construction

Fig. 5, instead, shows parts of the IMS Client code of the two C functions that process the new INVITE message and perform the switching on and MN re-registration, respectively **`ims_process_incoming_invite`** and **`ims_send_re_register`**. We pointed out all main management operations in bold. In the first function, the main operations are: the extraction of our novel power management attribute as a local IP address and the invocation of de-register/default invite depending on internal client state, i.e., **`newInvite`** flag. In the second function, they are: the execution of the scripts to switch on/off the wireless interface; the creation and initialization of a new eXosip stack (the C-based SIP stack used by UCT IMS Client); and the invocation of a new register phase. Finally, let us note that, depending on the internal state of the IMS client, re-registration may either occur on the wireless interface just switched on, for incoming calls, or on the always-on wireless interface, when the call has been terminated and **`is_bye`** flag is set.

As regards CM implementation, to portably read RSSI values we exploit the Wireless Research API (WRAPI) under Windows and the **`iwconfig`** tool under Linux [24], [25]. In addition, we have implemented several monitoring scripts to periodically invoke those system commands and update monitored values and monitoring periods.

```
void ims_process_incoming_invite(eXosip_event *je)
{ …
  sdp_message_t * sdp_message;
  eXosip_lock();
  sdp_message=eXosip_get_sdp_info(je->request);
  eXosip_unlock();
  switchOnAddress=extractPowerManAttribute(sdp_message);
  if(newInvite) ims_send_de_register();
  else { … /* standard session invite management */ }
}

void ims_send_re_register()
{
int port=5060, pid, status;
pid=fork();
if(pid==0) {  // child
  if(!is_bye) { execl("../scripts/switchOnInterface.sh",
                "switchOnInterface.sh", switchOnAddress, (char *)0 ); }
  else { execl("../scripts/switchOffInterface.sh",
         "switchOffInterface.sh", switchOnAddress, (char *)0 );  }
} else {      // parent
  wait(&status);
  if(!is_bye) { while( eXosip_listen_addr(IPPROTO_UDP, switchOnAddress,
                             port, AF_INET, 0) != 0 ) port++; }
  else { while( eXosip_listen_addr(IPPROTO_UDP, alwaysOnAddress,
                             port, AF_INET, 0) != 0 ) port++; }
  ims_send_register();
}
```

Fig. 5. IMS client implementation insights: new INVITE processing and re-registration

5 Experimental Results

We have thoroughly tested and evaluated the performance of IHMAS by deploying it in the heterogeneous wireless network at our campus. Our testbed consists of several Linux laptops equipped with two different wireless interfaces: 3G UMTS adaptors and OrinocoGold WiFi cards. The IMS infrastructure and IHMAS components run on Linux boxes, each one equipped with two 1.8 GHz CPUs and 2048MB RAM, by following the IHMAS deployment scheme of Fig. 1. To collect performance results, all machines are synchronized by using the lightweight precision time protocol, i.e., IEEE 1588 [26]. In addition, to analytically evaluate the battery saving achieved by our solution, we decided to use the Nokia N95 smart phone as our reference device. The choice of N95 is motivated by two main reasons. First, N95 was one of the first lightweight mobile devices to be equipped with WiFi and thus extensive battery consumption evaluations are already available for it. Second, N95 hosts the Java 2 Micro Edition (J2ME) and, as part of our ongoing work, we are porting the IMS client implementation presented in this paper to the J2ME platform – our new J2ME IMS client exploits the Ericsson reference implementation of the IMS services API [27].

In the following, we report experimental results, averaged over 1,000 session initiation cases, while provisioning a VoIP service that offers GSM-encoded audio flows with constant frame rate = 50 frames/s, buffer slots storing GSM/RTP packets

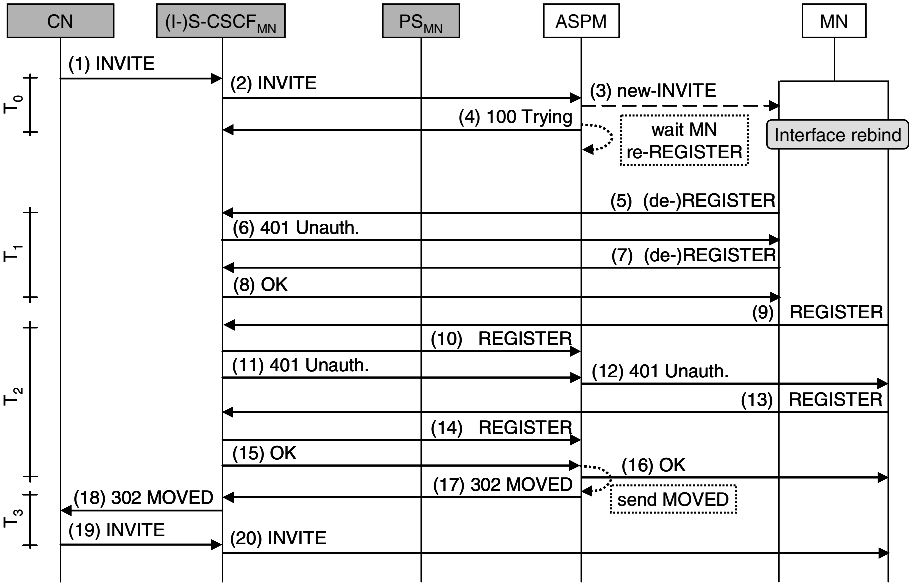

Fig. 6. Power management session initiation delay analysis

(89B payload per packet), and 13.2Kbps bandwidth. The reported experimental results aim to evaluate three different and crucial aspects of our proposal: i) how the proposed power management solution is compatible even with the strict time requirements of VoIP session initiation; ii) the battery lifetime improvements achievable via our energy-saving techniques; and iii) the overhead due to ASPM interposition and CM execution. All experimental results have been collected by using 3G as the always-on wireless interface and WiFi as the wireless interface that is switched on at runtime.

About session initiation delay imposed by call redirection, Fig. 6 shows the duration of all main power management phase. Results have been collected for the four main parts of the protocol. The first part starts with sending the CN INVITE message and terminates with TRYING reception; it occurs over 3G UMTS interface and lasts T_0=285ms with a standard deviation (st.d.) of 87ms. The second part is MN de-registration, which also occurs over the always-on interface and lasts T_1=570ms (st.d. 100ms). The third part is MN registration over WiFi that lasts T_2=825ms (st.d. 182ms), (assuming that MN IP address is configured). Finally, the fourth part relates to the time between the MOVED and the second INVITE messages sent at CN; we have measured T_4=93 (st.d. 68). Let us note that T_2 is longer than T_1 due to the higher number of messages exchanged caused by ASPM interposition; however, T_2 does not double T_1 because the WiFi response time is shorter than the UMTS one $T_{2.}$

The sum of all the above delays is always below 3s, which is indicated by E.721 as the recommended call setup delay for local calls [28]. Of course, this delay does not account for subsequent session initiation delay over WiFi; nonetheless, this can be considered as another session initiation phase since any IMS client, at the reception of 302 MOVED message, usually informs that a call re-direction is ongoing.

The second evaluation is an analytical evaluation of the benefits of our solution if deployed on off-the-shelf smart phones such as Nokia N95. Our evaluation also exploits the results of field trial experiments conducted by Arjona on always-on WiFi battery consumption of N95 [29]. Although the exact WiFi consumption depends on several parameters, including AP and client configuration, scanning frequency and additional power saving features, Arjona experiments demonstrate that the N95 WiFi antenna with default configuration shows an average consumption increase of 0.05W if left always-on (inactive); that is confirmed by the results for WiFi sleep mode consumption reported by Balakrishnan [14]. Considered that N95 battery lifetime is 950mAH and is charged at 3.7V, this means that battery lifetime is 950/((0.05/3.7)*1000) = 70.3 hours in the always-on configuration, by neglecting all other energy consumption contributions. That duration is significantly lower than the stand-by time of 200 hours with UMTS and of 188 hours with UMTS, IMS-based active registration, and network optimizations [29], [30]. With our power management technique, all the above WiFi energy cost can be completely cut off. Moreover, by focusing on WiFi call time duration and assuming 0.75W as the average consumption for active WiFi antenna [14], the total call time is 950/((0.75/3.7)*1000) = 4.6 hours, which is longer than the talk time duration specified by Nokia for UMTS, i.e., 160 minutes [30]. This also confirms the usefulness of our approach that enables the advantages of WiFi-based communication without paying its high standby energy costs.

The third reported result validates our solution with regard to the overhead in terms of CPU work load and scalability. As a general consideration, we have experimented that IHMAS relevantly improves energy consumption at the expense of a limited, even if not negligible, overhead. ASPM can serve up to 550 requests per-second; to further increase ASPM scalability we are considering the possibility to re-implement it as a new C-based OpenSIPS module. In addition, to this, there is an overhead related to CM execution. For all conducted experiments, we monitored CM CPU overhead. In the worst case, i.e., for high user mobility, it is up to 20%; for fixed users, thanks to dynamic monitoring period adaptation, it drops to 5%. However, as already mentioned, we are studying possible alternatives to further decrease that value and increase energy savings, such as connectivity forecasting.

6 Conclusions and Future Work

The research work accomplished within the IHMAS project demonstrates the suitability of power management solutions based on the standard IMS infrastructure. IHMAS practically shows that energy-saving techniques can relevantly increase battery lifetime when using high-consumption and low-cost wireless interfaces, by preserving full compliance with the standard and, thus, enabling the deployment of the proposal over already installed IMS-conformant networks. In addition, the session initiation delays introduced by the IHMAS facility for power management are compatible even with the strict requirements that are usual for VoIP calls.

The encouraging results that we have already obtained with IHMAS are stimulating our current research work on the extension and refinement of our solution prototype. On the one hand, we are developing a J2ME version of CM and IMS client, tailored to any consumer device hosting a J2ME platform. On the other hand, we are carefully evaluating the different IHMAS energy-saving results that can be obtained via either C-based or J2ME-based prototypes of IMS clients through extensive measurements of energy consumption in real and wide-scale deployment environments.

References

1. Kravets, R., Krishnan, P.: Power Management Techniques for Mobile Communication. In: 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking, pp. 157–168. ACM, New York (1998)
2. Anastasi, G., Conti, M., Lapenna, W.: Power Saving Policies for Wireless Access to TCP/IP Networks. In: 8-th IFIP Workshop on Performance Modelling and Evaluation of ATM and IP Networks (2000)
3. Anand, M., Nightingale, E.B., Flinn, J.: Self-Tuning Wireless Network Power Management. J. Wireless Networks 11(4), 451–469 (2005)
4. Inoue, M., et al.: Novel out-of-band Signaling for Seamless Interworking Between Heterogeneous Networks. IEEE Wireless Communications 11(2), 56–63 (2004)
5. Inoue, M., et al.: Context-Based Network and Application Management on Seamless Networking Platform. Wireless Personal Communications 35(1-2), 53–70 (2005)
6. Camarillo, G., García-Martín, M.A.: The 3G IP Multimedia Subsystem (IMS). John Wiley, Chichester (2006)
7. UCT IMS Client Project, `http://uctimsclient.berlios.de/`
8. Open IMS core Project, `http://www.openimscore.org/`
9. OpenSIPS Project, `http://www.opensips.org/`
10. Shacham, R., et al.: Composition for Enhanced SIP Presence. In: 12th Annual IEEE International Symposium on Computers and Communications, pp. 203–210. IEEE Press, New York (2007)
11. 3rd Generation Partnership Project (3GPP): Presence Service: Architecture and functional description. TS 23.141 v8.0.0 (2008)
12. 3rd Generation Partnership Project 2 (3GPP2): Presence Stage 3. X.S0027-003-0 v1.0 (2008)
13. Open Mobile Alliance: Presence SIMPLE Specification. TS-Presence_SIMPLE-V1_1-20080128-C (2008)
14. Krashinsky, R., Balakrishnan, H.: Minimizing Energy for Wireless Web Access with Bounded Slowdown. In: 8th Annual ACM International Conference on Mobile Computing and Networking, pp. 135–148. ACM Press, New York (2002)
15. Pletcher, N., Rabaey, J.M.: Ultra-Low Power Wake-Up Receivers for Wireless Sensor Networks. Technical Report No. UCB/EECS-2008-59. EECS Department, University of California, Berkeley (2008)
16. Schulzrinne, H., Wedlund, E.: Application-layer mobility using SIP. ACM Mobile Computing and Communications Review 4(3), 47–57 (2000)
17. Vali, D., et al.: An efficient micro-mobility solution for SIP networks. In: IEEE Global Communications Conference, pp. 3088–3092. IEEE Press, New York (2003)
18. Wu, W., et al.: SIP-based vertical handoff between WWANs and WLANs. IEEE Wireless Communications 12(3), 66–72 (2005)
19. Kalmanek, C., et al.: A Network-Based Architecture for Seamless Mobility Services. IEEE Communications Magazine 44(6), 103–109 (2006)
20. Bennett, A.: Voice Call Continuity (VCC) between Circuit Switched (CS) and IP Multimedia Subsystem (IMS)., Technical Specification (TS) 23.206, 3GPP (2007)

21. Bellavista, P., Corradi, A., Foschini, L.: Context-Aware Handoff Middleware for Transparent Service Continuity in Wireless Networks. Pervasive and Mobile Computing Journal 3(4), 439–466 (2007)
22. Nicholson, A.J., Noble, B.: BreadCrumbs: Forecasting Mobile Connectivity. In: 12th Annual ACM International Conference on Mobile Computing and Networking, pp. 46–57. ACM Press, New York (2008)
23. NIST, jain-sip Project, `https://jain-sip.dev.java.net/`
24. WRAPI Project, `http://sysnet.ucsd.edu/pawn/wrapi/`
25. Tourrilhes J., Wireless Tools for Linux Project, `http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/`
26. Precision Time Protocol Daemon Project, `http://ptpd.sourceforge.net`
27. JSR 281: IMS Services API, `http://jcp.org/en/jsr/detail?id=281`
28. ITU-T, Network Grade of Service Parameters and Target Values for Circuit Switched Services in the Evolving ISDN, Recommendation E.721, Telecommunication Standardization Sector of ITU (1999)
29. Arjona, A., Yla-JMski, A.: VoIP Call Signaling Performance and Always-On Battery Consumption in HSDPA, WCDMA and WiFi. In: International Conference on Wireless Communications, Networking and Mobile Computing, pp. 2964–2967. IEEE Press, New York (2007)
30. N95 Technical Specifications, `http://europe.nokia.com/A4323297`