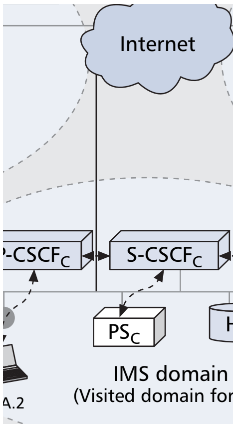
IMS-BASED PRESENCE SERVICE WITH ENHANCED SCALABILITY AND GUARANTEED QoS FOR INTERDOMAIN ENTERPRISE MOBILITY

PAOLO BELLAVISTA, ANTONIO CORRADI, AND LUCA FOSCHINI, UNIVERSITY OF BOLOGNA



The authors present an overview of very recent research contributions about the standard IMS presence service and some novel proposals for scalability and quality optimization. They propose a presence service solution that presents three original extensions for scalability and quality

ABSTRACT

Enterprise mobility and IP multimedia system corporate services are becoming more relevant; in particular, the IMS-based presence service is starting to provide different kinds of seamlessly mobile services, with context awareness in a wide range of enterprise-deployment scenarios. However, the deployment of presence-based enterprise services in wide and heterogeneous wireless networks still faces several challenges. First of all, scalability and differentiated quality are considered crucial, especially under heavy traffic conditions and when dealing with inter-domain mobility scenarios. This article presents an overview of very recent research contributions about the standard IMS presence service and some novel proposals for scalability and quality optimization. In addition, we propose a novel presence service solution that presents three original extensions for scalability and quality: a federated model to optimize inter-domain distribution of notification messages through locality-based aggregation, a proposal for differentiated quality levels, and an extension of client-side buffering for reliable delivery of presence messages, even during user roaming. The prototype of the enhanced presence service has been deployed and validated by obtaining performance results of improved scalability in terms of CPU and memory usage in infrastructure nodes and message traffic. The prototype is openly available to the IMS community for possible refinement and extension.

INTRODUCTION

A growing number of mobile users calls for seamless access to their services while they move across heterogeneous wireless infrastructures, spanning from IEEE 802.11 and Bluetooth to cellular third generation (3G) and beyond. Despite the recognized benefits of a full integration of wireless provisioning infrastructures, the development and deployment of enterprise mobile services over these networks still are challenging tasks due to the difficult service requirements of continuous accessibility, quality of service (QoS), and high scalability. In addition, the mobility of business users (within one enterprise and even between different enterprises) further stresses the service requirements and forces the consideration of innovative solutions for scalable mobility management, for example, in the case of abrupt changes in wireless access technologies, available resources and services, administration domains, and user- and service-related profiles (such as user location and online status — online, offline, busy, etc.). A deployment scenario that consists of a large number of business users moving between different and geographically distributed domains, called the *enterprise mobility scenario* in the following, requires prompt and interoperable notification of context changes for the dynamic adaptation of service provisioning. For this reason, presence services (PSs), traditionally exploited to maintain user online status only in the traditional, fixed Internet, are gaining the ambitious role of maintaining and disseminating the whole context of users and services in Internet Protocol (IP)-based mobile networks [1].

Given the recognized need of supporting mobile services over all-IP wireless networks, a wide range of standardization entities, from the 3rd Generation Partnership Project (3GPP) and 3GPP2 to the Internet Engineering Task Force (IETF) and the Open Mobile Alliance (OMA), has agreed to define the IP multimedia subsystem (IMS) [2]. IMS simplifies the design and implementation of mobile services by adopting an application-layer approach and by exploiting the Session Initiation Protocol (SIP) to harmonize session control. IMS recognizes PSs as a core support facility for a novel mobility-enabled service [3–5]; however, currently, the IMS-based PS exhibits several weaknesses that limit its widespread adoption for enterprise-level services. First, the current IMS PS specifications do not address coordination issues between PSs that are distributed in different IMS administration domains [6]. Second, IMS session signaling (and presence signaling, in particular) is likely to introduce relevant and non-scalable overhead, especially in 3G core networks and inter-domain interactions [7, 8]. Third, the IMS PS does not provide support for the differentiation of message handling priority yet, thus limiting the possibility of managing differentiated quality levels, for instance, to grant service levels to business users over non-business users, especially during traffic overload.

The first part of this article provides an updated overview of the current status of IMS-based PS standardization, by surveying the primary PS optimizations proposed in the literature. The second part proposes a novel PS solution that couples three original characteristics. First, we adopt a federated PS-server model to improve scalability and to optimize inter-domain distribution of notification messages through local aggregation. Second, the article proposes an original PS extension for differentiated QoS levels. Third, it applies client-side buffering techniques to grant reliable delivery of PS messages even during user roaming. The proposed PS solution was implemented, deployed, and thoroughly tested over a distributed testbed, including several IMS domains, and made openly available to the IMS community. The reported experimental results demonstrate that the proposal outperforms known solutions in the literature in terms of scalability of central processing unit (CPU) and memory usage at infrastructure nodes and of message traffic, with a very limited increase in message-delivery delay.

Background for IMS-Based PS

Presence is a well-known concept in the traditional Internet and widely used in applications such as instant messaging and multiparty games [1]; presence permits users and hardware/software components, called *presentities*, to convey their ability and willingness to communicate with *watchers*. To receive PS publish and update messages from presentities, that is, presence *notifications*, watchers subscribe to *presence servers* that act as intermediaries in any PS-related communication between presentities and watchers.

The concept of presence recently was enlarged to include any context information that is useful to adapt service provisioning to the current state of the execution environment in a personalized way; this enlargement helped PS become a core component of several IMS-based mobile services today. For instance, instant messaging exploits the PS context information about one user's online/offline/busy status to check whether she is reachable, and voice/video conferencing uses PS context to adapt sessions depending on profiles of currently used devices and wireless cards. In addition, several other applications can be envisioned, for example, paging services that exploit PS context to contact employees in an office or doctors in a hospital in the quickest way.

To fully understand both the following overview of the current literature about IMS-based extension proposals and our novel solution, first we briefly give some background about IMS and the IMS-based PS [2–5]. The core IMS functional entities are the:

- IMS client, which controls session setup and media transport through SIP extensions specified by the IETF and 3GPP IMS-related standards. A unique HTTP-like uniform resource identifier (URI), such as sip:user@domain, identifies an IMS client.
- Proxy-call session control function (P-CSCF), which establishes secure associations with mobile clients and routes outgoing and ingoing SIP messages to the inner IMS infrastructure on their behalf.
- Interrogating-CSCF (I-CSCF), which is responsible to interconnect and route SIP messages securely among different IMS domains.
- Application server (AS), which allows the introduction of new IMS-based services. The IMS-based PS also is realized as a specific AS; any IMS domain runs at least one IMS PS server.
- Home subscriber server (HSS), which stores authentication data and profiles for registered clients.
- Session-CSCF (S-CSCF), which is the core component enabling the coordinated interaction of all IMS entities. S-CSCF initially registers IMS clients by interacting with HSS. Moreover, depending on the filters/triggers specified by client profiles, S-CSCF can differentiate the routing of specific types of SIP messages to different ASs. For instance, S-CSCF identifies PS messages and forwards them to the PS server.

Specifically focusing on the core components of IMS-based PS, they are:

- IMS clients, which can serve as either presentities or 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 term, presentity, to refer to both).
- Presence server (PS), which facilitates PS interactions. It accepts and stores PS subscriptions from watchers and sends `NOTIFY` messages, triggered by `PUBLISH` messages from presentities, to registered watchers.

In addition, the PS defines other components, such as the PS network agent, which collects and combines 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, etc.). For additional details about IMS and its PS, please refer to [2].

Figure 1 shows the deployment of the above components in a general scenario of inter-domain PS subscriptions: core IMS components are in grey, and PS servers in white. To refer to watcher 1 in domain A, we use the notation $W_{A.1}$; similarly, presentity 1 in domain B is denoted by $P_{B.1}$. For a domain, we annotate all watchers/presentities depending on the represented home or visited domain. Watchers and presentities execute at mobile clients; P-CSCF is deployed at the network edges of the networks visited by clients; HSSs, I-CSCF, S-CSCFs, and PSs are located in home networks.

The concept of presence recently was enlarged to include any context information that is useful to adapt service provisioning to the current state of the execution environment in a personalized way; this enlargement helped PS become a core component of several IMS-based mobile services.

Notwithstanding recent efforts on IMS infrastructure standardization, IMS-based service deployment is still an open issue, mainly in terms of scalability because of the heavy signaling traffic associated with IMS.

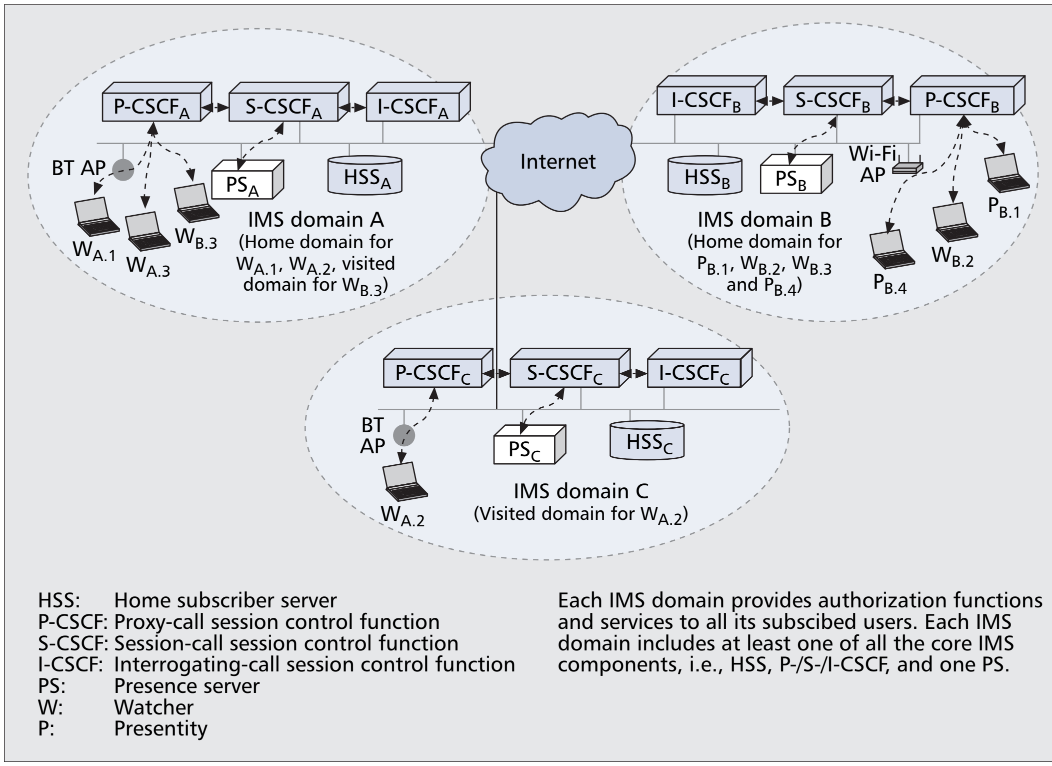


Figure 1. *Interdomain PS architecture.*

IMS-Based PS: Open Issues and State-of-the-Art

For a better understanding of our proposal, we summarize the primary issues that emerge in the enterprise-level exploitation of IMS-based services and briefly review the main research activities that recently addressed some of these issues.

Scalability and Distributed-Management Issues

Notwithstanding recent efforts on IMS infrastructure standardization, IMS-based service deployment is still an open issue, mainly in terms of scalability because of the heavy signaling traffic associated with IMS. SIP messages are rather verbose and text-based; the number of exchanged messages for each dialog is high, and a message should be confirmed over User Datagram Protocol (UDP) to grant reliability. When specifically considering the IMS-based PS, scalability is an even bigger issue (see [9] for an analytical evaluation of PS overhead). In fact, whenever a presentity sends a `PUBLISH` message, the PS server in the home domain must forward it to all subscribed watchers; that generates a number of `NOTIFY` messages that exponentially grows with the number of publications. Moreover, PS traffic coexists and competes with other session-control-signaling flows, which are likely to increase in the next few years because of increased diffusion of IMS and IMS-based services. We claim that there is definitely the need to support and enforce differentiated service management operations (for instance, depending on the service type, subscribed user contract, etc.) at both the IMS core components and the ASs, for example, the IMS-based PS.

In addition, inter-domain interactions and user mobility exacerbate negative effects on PS performance. In fact, inter-domain communications increase the number of involved IMS entities and also are more challenging because network links between different IMS-domains are typically slower than intra-domain ones [7, 8]. In particular, in wide-scale deployment scenarios with inter-domain user mobility, PSs must face four main critical situations:

- *Single watcher of multiple presentities in another domain* — When one watcher registered in one domain subscribes to several presentities in a different domain, the presentities PS must deliver multiple `NOTIFY/OK` message pairs (one for each subscription) [10]. For instance, in Fig.1, consider PS_B and the watcher $W_{A.1}$, which subscribes to its buddies $P_{B.1}$ and $P_{B.4}$ from domain A.
- *Multiple watchers of a single presentity in another domain* — When several watchers in one domain subscribe to the same presentity registered in a different domain, the presentity PS must generate multiple `NOTIFY/OK` transmissions (one for each watcher in the foreign domain) [10]. For instance, see PS_B in Fig.1 and consider both $W_{A.1}$ and $W_{A.3}$ subscribing to $P_{B.1}$.
- *Triangulation* — When one roaming watcher subscribes to a presentity in a foreign domain (different from its current one), all the `NOTIFY` messages are routed though the S-CSCF home component of the watcher, without taking into account the watcher-to-PS distance. For instance, see Fig.1 with PS_B and $W_{A.2}$ subscribing to $P_{B.4}$.
- *Traversing node number* — The PS standard specification does not limit the number of traversing IMS entities in the path from pre-

Solution	IMS-compliant	Federated PS	Delayed NOTIFY	Reduction of exchanged NOTIFY messages
Batched notification	Yes; requires SIP event framework extensions (not addressed)	Yes	Yes	Depends on batching period and policy (per-watcher/domain)
Common NOTIFY	Yes; SIP event framework extensions (not addressed)	Yes	No	For each presentity P_i, it is possible a reduction of 1/(number of P_i watchers in other domains)
View Sharing	Yes; proposes SIP event framework extensions	Yes	N/A — see Common NOTIFY	N/A — see common NOTIFY
TCP adoption	No; requires to eliminate OK messages for NOTIFY dialogs	No	No	All OK messages
MSRP adoption	No; requires intra-domain coordination among PS and MSRP relays	No	Yes	Depends on MSRP relays

Table 1. *Comparison of interdomain PS optimization techniques.*

sentities to watchers. Hence, especially if intermediate entities (e.g., PS network agents) are deployed for presence-data gathering/aggregation, it is possible to generate long session-control paths and even circular subscriptions.

To solve the above problems, new models and standards are required for better inter-domain PS coordination [11]. New requirements include: distributed storage and management of user data, new protocols for inter-domain PS communications, and the non-disclosure of the private data of a presentity to unauthorized watchers.

Emerging Partial Solutions for Inter-Domain PS Coordination

Some solutions to reduce PS traffic on the last hop in wired-wireless-integrated networks have already been standardized, but only for intra-domain scenarios (e.g., signaling compression, resource lists to enable multiple subscriptions through a one-only `SUBSCRIBE` message, pull-based interactions by using standard `SUBSCRIBE` messages with expiry time 0, partial notifications to reduce `NOTIFY` message length, etc.) [5]. In contrast, the research work addressing inter-domain scenarios is still in its infancy. The very few SIP-compliant proposals that have emerged in the field include:

- *Batched notification* — This technique solves the first issue raised in the previous section by proposing the reduction of the number of `NOTIFY/OK` messages through aggregation; aggregation can be performed on an either per-watcher or per-domain basis [10]. The main weakness is that `NOTIFY` aggregation introduces significant delays when compared to the delivery of single `NOTIFY` messages, depending on the selected batching period.
- *Common* `NOTIFY` *to multiple watchers* — This technique solves the second issue raised in the previous section by suggesting the delivery of a single `NOTIFY/OK` message to the watchers' PS that redistributes it locally to all watchers [10]. This solution implies that watchers' data is passed to the watchers' PS; hence, the assumption is a trust relationship between PSs that respect agreed-upon privacy rules.
- *Optimizing federated presence with view sharing* — This optimization proposes a mechanism to share views of subscribed watchers among domains and defines the related required extensions of the SIP event framework [11]. After mutual authentication, federated domains trust each other and exchange access control lists to update their views of authorized watchers. This technique can be used to enable the previously presented common `NOTIFY` optimization.

These solutions have the relevant advantage of not imposing modifications to the existing IMS infrastructure. Further improvements could be obtained by relaxing the compliance constraint: for instance, the use of Transmission Control Protocol (TCP) instead of UDP would eliminate `OK` confirmations for inter-domain `NOTIFY` dialogs; the adoption of the Message Session Relay Protocol (MSRP), with MSRP relays, would aggregate multiple `NOTIFY` messages, at the expense of increased notification delays [9].

A very crucial technical point is that all of the above proposals have not been applied to the IMS-based PS yet. In addition, they reduce inter-domain `NOTIFY` exchange, but batched notification and MSRP achieve this goal by increasing delays for watchers. Moreover, all these solutions introduce longer delays due to the required relay of `NOTIFY` messages through both the presentity and watcher PSs. Finally, they do not address service differentiation at all, whereas other recent proposals in other IMS-related areas are starting to work on that, for instance, [12] suggests to prioritize emergency calls over regular ones by differentiating call session initialization at the IMS core.

Enterprise PS Enhancements for Service Differentiation and User Mobility

Notwithstanding some seminal research work, several enterprise requirements for PS are still largely unexplored. Service differentiation is

The key component in our proposal is the inter-domain optimization module that is deployed at each IMS domain and interacts with the local PS to implement PS optimizations together with differentiated quality levels.

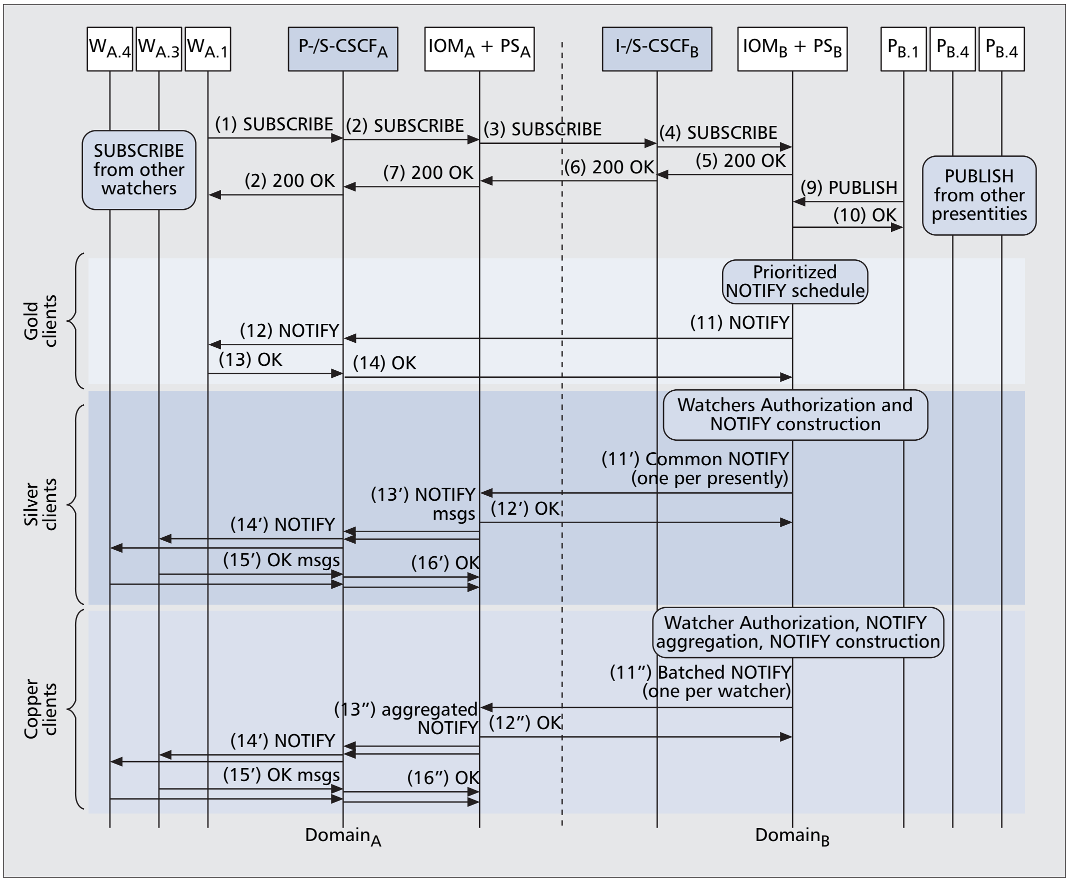


Figure 2. *Message flow for differentiated PS enhancements.*

unsupported: all watchers are uniformly handled at the PS and even if they specify diversified maximum NOTIFY delays, the consequent PS operations would complicate resource management and accounting, especially for wide-scale providers. Issues related to enterprise-user mobility, for example, inter-domain roaming and triangulation, have not been addressed at all. Last, but very relevant, to the best of our knowledge, none of the optimizations in the previous section have been implemented and validated over real multi-domain IMS benchmarks.

Our research work aims to overcome the above limitations. First, our solution defines three main classes of service (gold, silver, and copper) and differentiates notification processing to grant prompter notifications to higher priority classes: enterprise users can obtain service-quality differentiated-service provisioning according to internal organization roles; for instance, a possible mapping is workers to copper, managers to silver, and top managers to gold. In short, the primary idea is to exclude privileged clients from traffic reduction optimization techniques that tend to delay notifications. Second, our PS proposal can overcome message losses, which may occur during user roaming and temporary disconnections: it exploits original and lightweight heuristics for mobility prediction to foresee roaming events and masks potential losses through local buffering at IMS clients. Third, our solution implements all the optimization techniques already sketched, by adding them to the scenario of service differentiation and mobility enhancements. Last, but very relevant, our proposal was designed and implemented at the application-layer, without modifying the core IMS components, thus achieving full compliance with standard IMS-based PS and potentially also leveraging the adoption of our solution in already-deployed network environments.

The key component in our proposal is the inter-domain optimization module (IOM) that is deployed at each IMS domain and interacts with the local PS to implement PS optimizations together with differentiated quality levels. The IOM consists of a client and a server part. The IOM client acts as a local proxy for external PSs and mediates all watcher-to-PS and inter-domain PS-to-PS interactions. It accepts watcher subscriptions in the watcher home domain and reroutes them to external PSs by also adding a record-route header field in order to receive successive messages; afterwards, it receives and redistributes aggregated notifications. The IOM server handles subscription requests from external IOM clients: it determines the watcher class and subscribes to the local PS on its behalf; it captures notifications and delivers them after aggregation. To maintain the required level of trust between domains, IOMs are federated: each IOM keeps a list of authorized IOM peers and exploits transport-layer security for mutual authentication and data encryption.

Service differentiation is enforced by our IOM on a per-session base, depending on watcher indications in their SUBSCRIBE message headers. The IOM associates gold watchers with the notify-message management that grants the shortest notification delays. Gold watchers are excluded from traffic optimizations: the IOM does not participate in their service routing because they interact directly with external PSs; in addition, the PS schedules gold `NOTIFY` messages with the highest priority. Silver watchers, instead, are served by applying the common `NOTIFY` technique: for each `PUBLISH` message, the IOM server interacts with the PS to gather the list of all subscribed and authorized watchers and sends them possibly aggregated `NOTIFY` messages. Finally, copper watchers receive batched notification management: for each watcher, the IOM server batches its notifications and delivers an aggregated `NOTIFY` message, either at specified intervals or when the message size exceeds the maximum UDP payload length, so as to avoid message split.

Our solution differentiates PS management by working only at PS end-hosts for the sake of full compliance; when novel techniques for signaling-traffic differentiation will be introduced in the IMS standard specifications and will be available in IMS core implementations, our IOM will be easily extended to take advantage of those new features, with the consequent further improvement of its performance.

Figure 2 details message flows corresponding to our gold, silver, and copper PS enhancements. Continuous black lines represent the IMS session-signaling protocol. Subscription and publish phases (steps 1–10) are the same for all management scenarios. The notification phase, instead, is differentiated per class: the gold watcher interactions with the external PS are direct (steps 11–14), whereas silver and copper ones are mediated by the IOM (steps 11′–16′ and 11″–16″). S-CSCF_B may not take part in PS notification; however, in our experiments, we included it in the backward `NOTIFY` path because it enables accounting, a crucial function in the enterprise service scenario under consideration.

With regard to our PS mobility extensions, we introduced a lightweight client stub (CS) that extends the IMS client with lightweight and decentralized predictions of client roaming (and possible network detachments) through local monitoring of wireless interfaces at the client side only [13]. Proactive prediction of user roaming is crucial to start early local management operations, which are required to mask client disconnections. In particular, when the CS predicts an IMS client disconnection, it starts to locally buffer `SUBSCRIBE` and `PUBLISH` messages from all mobile services running at the client. As soon as another wireless interface is available, it triggers the IMS client to promptly send a `REGISTER` message, flushes all PS messages in its buffer, and receives, and locally delivers, incoming notifications.

Finally, let us note that IOM interposition cannot solve completely the performance issues related to triangulation. In fact, according to the standard IMS-based PS specification, all communications toward a roaming watcher (connected to a foreign P-CSCF) must pass through its home S-CSCF [3, 4]. Therefore, to be IMS-compliant, our optimizations can be applied on the segment path between different S-CSCFs but not on the last segment path between S-CSCF and P-CSCF. This limitation, however, is less relevant in the wide-scale deployment scenarios addressed: large enterprises also can distribute geographically several IOMs, PSs, P-/S-CSCFs, to serve as local entry points for employees who are far from home.

PS Enhancement Implementation and Performance Results

We thoroughly tested and evaluated the scalability performance of our solution by validating it over our multi-domain IMS testbed deployment. Our IMS infrastructure components run on Linux boxes, each equipped with two 1.8 GHz CPUs and 2048 MB of RAM. The distributed testbed consists of two domains (domain A and domain B) that follow the IMS-based PS specifications. For each domain, each IMS core component (P-/I-/S-CSCF and HSS) runs on a single host, while IOM and PS execute together on a dedicated host (Fig. 1). We realized the IOM component as a new module for the OpenSER PS by adding some original features: the IOM server interacts with the OpenSER PS module to implement our enhancements and uses the PS user agent module to interact with external PS servers [14]. In addition, our enhanced PS is based on OpenIMSCore, an open and recognized implementation of the main IMS components. Finally, our CS was implemented by using standard Linux tools for wireless interface query and designed to integrate with the open-source University of Cape Town (UCT) client. Our prototype and multi-domain testbed are openly available for the IMS community.[1]

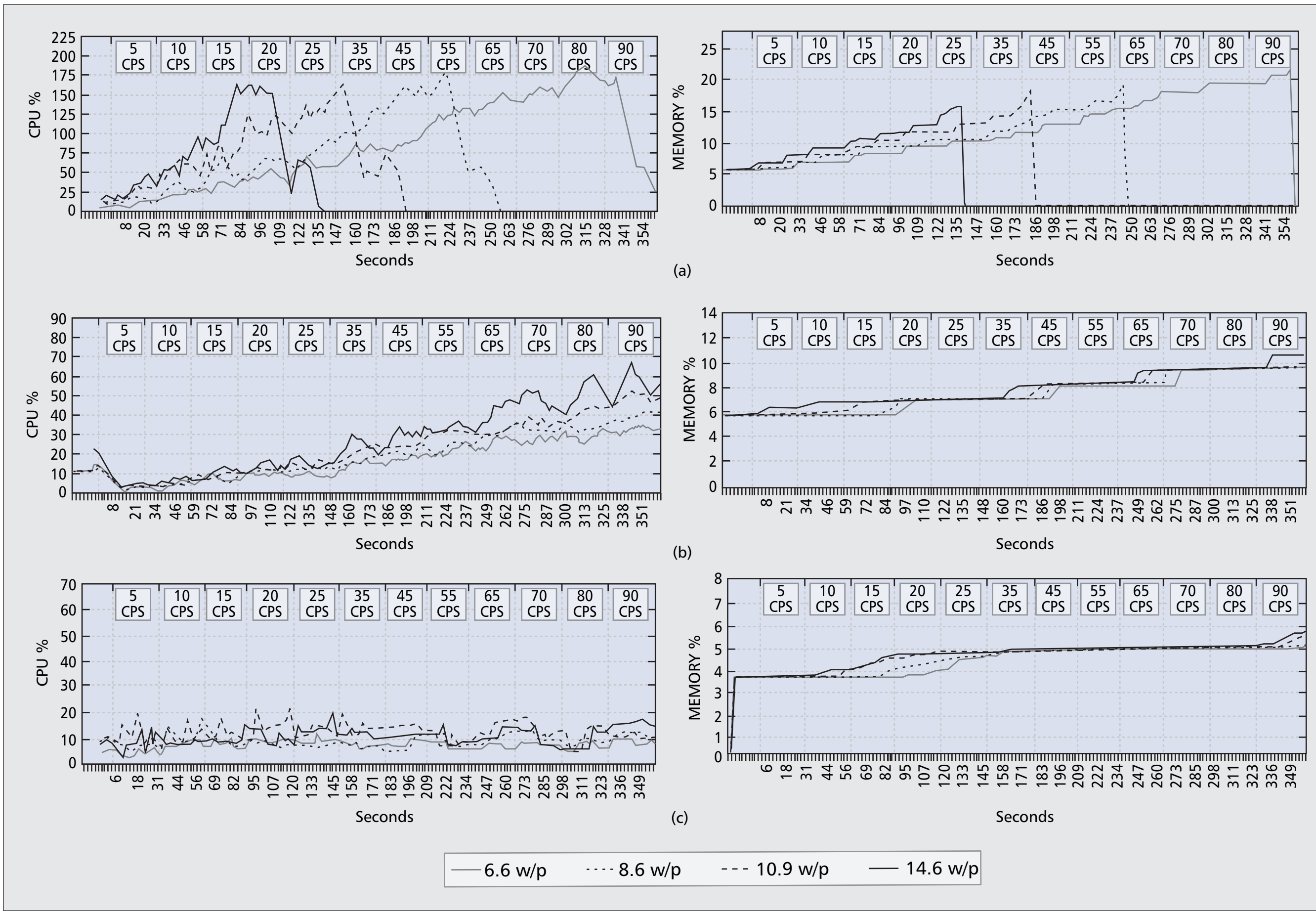
To test inter-domain system scalability, we employed the IMS Bench SIPp, an IMS traffic generator that conforms to the European Telecommunications Standards Institute (ETSI) TS 186 008 IMS/NGN Performance Benchmark specification. IMS Bench SIPp permits the definition of benchmark configuration scenarios that correspond to different IMS session phases (e.g., registration, subscription, etc.). In the experiments, we defined three main phases. In the first phase, IMS clients register with a constant arrival rate of 10 calls per second (cps); this phase lasts 15 s and registers 150 clients, equally distributed over the two domains. In the second phase, all watchers subscribe to the presentities: in our case, all watchers belong to domain A and all presentities to domain B; in addition, we tuned phase duration to obtain a different number of subscriptions for single presentity, that is, the watcher/presentity (w/p) factor. Finally, during the third phase, presentities inject `PUBLISH` messages.

The reported experimental results focus on potential system bottlenecks for wide-scale deployment to point out three different and crucial aspects: the first one stresses the workload reduction that the adopted PS enhancements enable in inter-domain routing; the second one evaluates inter-domain SIP traffic reduction; the third one focuses on decreasing inter-domain delays by service differentiation, even in overload conditions.

Regarding the local workload in inter-domain

To be IMS-compliant, our optimizations can be applied on the segment path between different S-CSCFs but not on the last segment path between S-CSCF and P-CSCF. This limitation, however, is less relevant in the wide-scale deployment scenarios addressed.

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



■ **Figure 3.** *CPU and memory performance results at S-CSCF: a) no optimization; b) common NOTIFY; c) batched NOTIFY.*

PS routing, we report results about the most overloaded component: the external S-CSCF (S-$CSCF_B$ in Fig. 2). The same considerations also apply to other components, such as the local S-CSCF in the non-optimized PS scenario. Figure 3 reports CPU and memory utilization (from 0 to 200 percent, summing up the usage percentage of the two CPUs) for our benchmark phases; all presented measurements have exhibited a limited variance (under 5 percent for more than one hundred runs). In particular, the third phase was configured with 12 incremental steps with 30 s of duration, going from 5 cps to 90 cps according to a Poisson distribution. After a preliminary evaluation of the system scalability threshold, we have determined four w/p thresholds (i.e., 6.6, 8.6, 10.9, and 14.6 w/p) to increasingly stir up the system to its upper limit.

Without any PS enhancements, S-CSCF cannot terminate step 12 and collapses at 353 s, 244 s, 183 s, and 137 s for 6.6, 8.6, 10.9, and 14.6 w/p, respectively (Fig. 3a). The IOM activation can greatly improve system scalability; in particular, the common and batched `NOTIFY` enhancement (Figs. 3b and 3c) effectively can drop both CPU and memory usage at the S-CSCF by decreasing the number of `NOTIFY` messages exchanged. The bursty CPU trend in Fig. 3c is due to notification batching at the IOM; however, the CPU oscillation interval is modest and its peaks are always under 20 percent (much lower than the peaks usually due to common `NOTIFY` optimization).

Even at the saturation threshold without PS enhancements, the IOM continues to work properly and avoids the congestion of inter-domain IMS components at an acceptable overhead cost. In particular, the system with our enhancements saturates only at a `PUBLISH` rate of 250 cps with 14.6 w/p (not shown in figure). At this point, the CPU usage at PS is always below 120 percent for common `NOTIFY`, whereas it slightly increases to 140 percent for batched `NOTIFY`. However, due to its ability to better distribute the inter-domain network load over time, `NOTIFY` batching proved to be less prone to `PUBLISH` message peaks. In both cases, the IOM presents a limited overhead, and most CPU overhead is due to database access: 80 percent for 250*14.6 = 3650 accesses per second.

The second set of experimental results in Fig. 4 reports inter-domain traffic reductions by the IOM interposition; Figs. 4 and 5 are in a logarithmic scale for sake of readability. The results relate to the same experimental scenarios considered before, with the addition of all messages potentially sent to the S-CSCF reported as the first column (including the ones lost in the case of an S-CSCF crash due to overload). The w/p ratio increase does not produce any significant change to common `NOTIFY` optimization, whereas the number of batched `NOTIFY` messages slightly decreases because of favoring by higher w/p factors.

The third test shows how our service differentiation techniques can achieve low inter-domain delays. We evaluated the average delay between

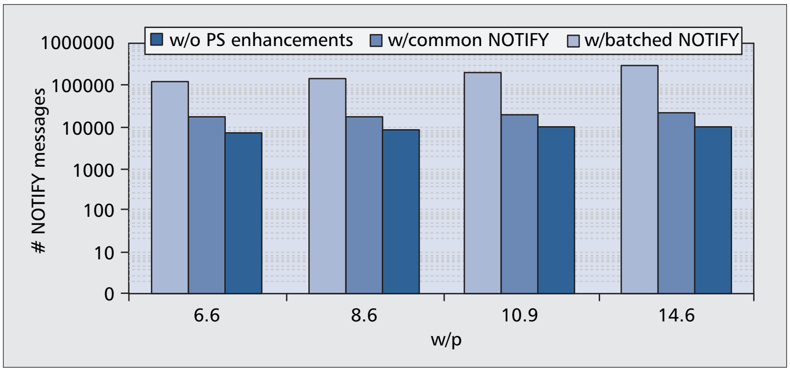
each PUBLISH and the corresponding NOTIFY message. Under the same workload conditions of the previous experiments, we collected delays for:
- All the clients without our PS enhancements
- All the clients of any service class with the IOM running, as reported in Fig. 5.

Under low load conditions (6.6 w/p), all delays are below 20 ms. However, without our PS enhancements, as load increases, these average delays increase linearly on a logarithmic scale. Our optimization has reported gold and silver inter-domain delays always below 12 ms and 18 ms, with a lower gold delay because of the higher processing priority at the PS. For the sake of readability, we omitted reporting copper delays that definitely are not comparable, that is, about 3490 ms, due both to the delay of 3.2 s and the batching period of 500 ms applied to copper clients. All collected measurements exhibited a limited variance: always below 10 ms w/o service differentiation and for the copper client and below 4 ms for the gold and silver client (for more than one hundred runs).

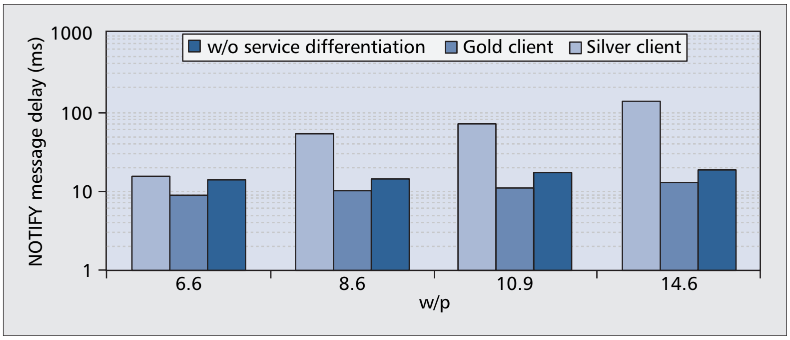
Conclusions and Future Work

Our work within the PS demonstrates the suitability of a service differentiation approach to enhance the standard IMS infrastructure for improved performance in inter-domain enterprise-service scenarios. The reported results are encouraging and confirm the good scalability of all of the proposed PS enhancements. Our work preserves full compliance with the standard and, thus, enables the deployment over already installed IMS-conformant networks. We are convinced that the availability of our developed prototype and inter-domain IMS testbed could significantly contribute to the widespread deployment of IMS-based technologies.

The encouraging results already obtained are motivating future work in two primary directions. On the one hand, we are working on the IOM client to also limit PS subscription traffic by aggregating multiple watcher SUBSCRIBE messages and by exchanging lists of new or updated subscriptions. On the other hand, we are using our PS enhancements to implement a novel IMS-based location-aware agenda for mobile users willing to track their mutual positions and activities while moving in our campus wireless infrastructure.



■ **Figure 4.** *Number of interdomain exchanged messages.*



■ **Figure 5.** *NOTIFY message delay w/o service differentiation.*

References

[1] R. Shacham *et al.*, "Composition for Enhanced SIP Presence," *IEEE ISCC*, 2007.
[2] G. Camarillo and M. A. García-Martín, *The 3G IP Multimedia Subsystem (IMS)*, 2nd ed., Wiley, 2006.
[3] 3GPP TS 23.141, "Presence Service: Architecture and Functional Description," v. 8.0.0, Mar. 2008.
[4] 3GPP2 X.S0027-003-0, "Presence Stage 3," v. 1.0, Feb. 2008.
[5] OMA-TS-Presence_SIMPLE-V1_1-20080128-C, "Presence SIMPLE Specification," Jan. 2008.
[6] H. Khlifi and J.-C. Gregoire, "IMS for Enterprises," *IEEE Commun. Mag.*, vol. 45, no. 7, Jul. 2007, pp. 68–75.
[7] D. S. Tonesi *et al.*, "Evaluation Of Signaling Loads in 3GPP Networks," *IEEE Wireless Commun.*, vol. 15, no. 1, Feb. 2008, pp. 92–100.
[8] P. Agrawal *et al.*, "IP Multimedia Subsystems in 3GPP and 3GPP2: Overview and Scalability Issues," *IEEE Commun. Mag.*, vol. 46, no. 1, Jan. 2008, pp. 138–45.
[9] A. Houri *et al.*, "Presence Interdomain Scaling Analysis for SIP/SIMPLE," IETF Internet draft, Feb. 2008; draft-ietf-simple-interdomain-scaling-analysis-04.
[10] V. K. Singh *et al.*, "Presence Traffic Optimization Techniques," tech. rep., Columbia Univ., Oct. 2006.
[11] J. Rosenberg *et al.*, "Optimizing Federated Presence with View Sharing," IETF Internet draft, Feb. 2008; draft-ietf-simple-view-sharing.
[12] M. El Barachi *et al.*, "Context-Aware Signaling for Call Differentiation in IMS-Based 3G Networks," IEEE ISCC, 2007.
[13] P. Bellavista *et al.*, "Context-Aware Handoff Middleware for Transparent Service Continuity in Wireless Networks," *Pervasive Mobile Comp. J.*, vol. 3, no. 4, Aug. 2007, pp. 439–66.
[14] OpenSER Project; http://www.openser.org/

Biographies

PAOLO BELLAVISTA [SM] (paolo.bellavista@unibo.it) graduated from the University of Bologna, Italy, where he received his Ph.D. degree in computer science engineering in 2001. He is now an associate professor of computer engineering at the University of Bologna. His research activities span from mobile agent-based middleware solutions and pervasive wireless computing to location-/context-aware services and adaptive multimedia. He is a senior member of ACM. He is an editorial board member of *IEEE Communications Magazine* and *IEEE Transactions on Services Computing*.

ANTONIO CORRADI [M] (antonio.corradi@unibo.it) graduated from the University of Bologna, Italy, and received an M.S. in electrical engineering from Cornell University, New York. He is a full professor of computer engineering at the University of Bologna. His research interests include distributed and parallel systems and solutions, middleware for pervasive and heterogeneous computing, infrastructure support for context-aware multimodal services, network management, and mobile agent platforms. He is member of the ACM and the Italian Association for Computing (AICA).

LUCA FOSCHINI [M] (luca.foschini@unibo.it) graduated from the University of Bologna, Italy, where he received a Ph.D. degree in computer science engineering in 2007. He is now a research fellow in computer engineering at the University of Bologna. His interests include distributed systems and solutions for pervasive computing environments, system and service management, context-aware services and adaptive multimedia, and mobile agent-based middleware solutions. He is a member of AICA.