Toward Feasibility and Scalability of Session Initiation and Dynamic QoS Provisioning in Policy-enabled Networks

Kamel Haddadou[1], Yacine M. Ghamri-Doudane[2], Marc Girod-Genet[3], Ahmed Meddahi[4], Laurent Bernard[3], Gilles Vanwormhoudt[4], Hossam Afifi[3], and Nazim Agoulmine[1]

[1] LIP6 Laboratory, Pierre & Marie Curie University, 8, rue du Capitaine Scott,
75015 Paris, France
{Kamel.Haddadou, Nazim.Agoulmine}@lip6.fr
[2] Computer Engineering Institute (IIE-CNAM), 18 Allée Jean Rostand,
91025 Evry Cedex, France
Ghamri@iie.cnam.fr
[3] GET/INT, Network and Telecommunication Service Department, 9 rue Charles Fourier,
91011 Evry Cedex, France
{Marc.Girod_Genet, Laurent.Bernard, Hossam.afifi}@int-evry.fr
[4] ENIC Telecom LILLE 1, rue Guglielmo MARCONI,
59658 Villeneuve d'Ascq Cedex, France
{Meddahi, Vanwormhoudt}@enic.fr

Abstract. In this paper we implement and evaluate a new solution for the multimedia session setup with resource allocation in policy-enabled networks that we developed in [1]. Our proposal is based on the use of Session Initiation Protocol (SIP) in the framework of Policy-Based Management (PBM)[1]. We specifically evaluate the feasibility and the scalability of such solution in a real environment through experimentation on a test-bed. The latter integrates all the components from session initiation to QoS policy enforcement into network elements. Our results demonstrate both of the relevance and the efficiency of such solution.

Keyword: SIP, PBM, Dynamic QoS Provisioning & Signaling.

1 Introduction

A major challenge in emerging multi-service, QoS-capable telecommunication networks is the deployment of high-quality multimedia applications. Both of network operators and end users are willing to offer and use multimedia communications with a large range of QoS-guarantees. To achieve this aim, an efficient control and management of network resources are submitted to be the key issues in the telecommuni-

[1] This work is undertaken within the MMQoS Project founded under the RNRT/French Ministry of Industry program.

cations world. So, we argue that a combination of end-to-end signaling and Policy-Based Management (PBM) [2] is required to enable proper multimedia sessions.

PBM aims to facilitate the management activity as it allows network administrators to define high-level objectives of network management schemes based on a set of policies. This latter is a set of pre-defined rules controlling network resources. Rules, established by the network administrator, include actions to be triggered when a set of conditions is fulfilled. PBM approach allows in its turn the translation of these high-level rules to a set of low-level device-compliant configuration commands.

As for signaling, SIP [3] is gaining increasing momentum as a protocol that enables set up of multimedia sessions. It has been adopted by both the Internet Engineering Task Force (IETF) for IP-based networks, as well as the 3rd Generation Partnership Project (3GPP) for next generation mobile networks.

Increasing the session establishment flexibility, while dynamically controlling the access to the resources, makes it thus easier to guarantee QoS in multi-service networks. In our work referred in [1], we proposed a novel approach that integrates session establishment with dynamic QoS control. We specifically transfer parts of the network management and control mechanisms to the user's terminal.

In the current work, we analyze the feasibility and the scalability of this approach taking into account session-oriented QoS parameters, mainly the delay of session establishment, as required in ITU-T recommendations [4].

We first integrate session establishment with dynamic resource allocation. In other words, we relate how to link session setup using SIP to QoS management using PBM. We then transfer numerous signaling and management functions to the user's terminal.

The rest of this paper is organized as follows: Section 2 introduces SIP and PBM. Section 3 presents our solution and some main related features. In Section 4, we describe the test-bed on which we implemented our solution. The empirical results targeting several feasibility and scalability testing scenarios are analyzed in Section 5. Finally, Section 6 concludes the paper and presents some future works.

2 Background

As our solution is based on SIP and PBM integration, these technologies are described briefly in the following sub-sections:

2.1 The Session Initiation Protocol (SIP)

As it was standardized by the IETF, SIP [3] can be considered as an application-layered signaling protocol. Its main role is to set-up sessions or associations between two (or occasionally more) Internet users or systems. The sessions that are initiated with SIP can be used to exchange various types of media. Specifically, SIP sessions are commonly used for handling voice media over packet networks. SIP is a client/server-oriented protocol with two types of messages: requests and responses. Messages are encoded in textual format using a structure similar to the HyperText Transfer Protocol (HTTP). It defines several messages to request action from the server.

One of these messages is the INVITE message that is sent to invite another participant to a session, the other is the BYE message for closing a session, and finally the ACK message to confirm session establishment. The response contains a status code to indicate the success or the failure of the request (e.g. 200 OK for establishment success). Message's body can also contain media or session description. Hence, when establishing a session, SIP exchanges also media attributes in order to share a common set of capabilities.

In SIP based network architecture, the previous messages are exchanged between four main types of entities playing different roles: User Agent, Proxy Server, Redirect Server and Registrar server. User Agents (UA) are SIP endpoints that send (caller) or receive (callee) signaling messages. A UA is divided in two components, the first acts as Client (UAC) and initiates the sessions; the second acts as a server (UAS) which is responsible for replying the session initiators. A UA communicates with another one directly or via intermediate proxy and redirect servers.

Proxy servers are application-layer routers that forward SIP requests and responses. Redirect servers receive requests and then return the location of the targeted SIP UA or location of the server where this user might be found. Registrars keep track of participant information (correspondence between SIP and IP address, access rights…). Proxy or redirect servers use registrars to determine routing or participant policies.

2.2 PBM Architecture and COPS Protocol

The end to end negotiation process, offered by SIP, only ensures capability exchanges. To allow Quality of Service (QoS) provisioning in an IP backbone, capability exchange is not sufficient and resource reservation mechanisms have also to be considered. In the current research initiatives, that are undertaken in order to simplify network resource management, the PBM [2] approach is the one which gains more interests due to the important number of advantages that it offers.

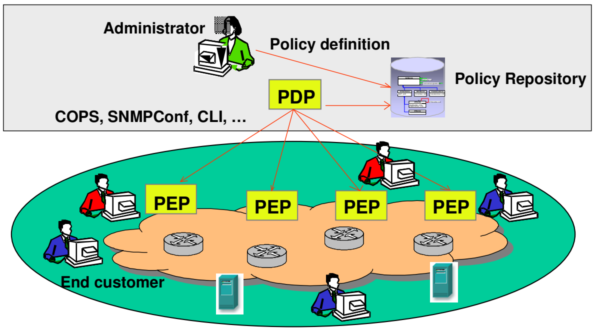

Figure 1. PBM components.

The IETF Resource Allocation Protocol (RAP) Working Group has specified a scalable and secure framework for policy definition and administration [5]. This framework introduces a set of components to enable policy rules definition, saving and enforcing: the Policy Enforcement Point (PEP), the Policy Decision Point (PDP) and the Policy Repository (Figure 1). PEP components are policy decision enforcers located in network and system equipments. The PDP is the component responsible for high-level decision-making process. This process consists of retrieving and interpreting policies, and implementing the decision in the network through the set of PEPs. The policy repository contains policy rules that are used by the PDP. To describe policies and network information, the IETF has adopted the Common Information Model (CIM) [6], which is a neutral scheme implementation describing overall management information.

In order to exchange management information and/or decisions, the PDP interacts with each PEP using one of the several protocols specified or extended for this purpose. Among them, the Common Open Policy Service (COPS) protocol [5] is the one which was designed specifically by the IETF to realize this interaction.

2.2.1 COPS Protocol

COPS is a client/server protocol allowing the exchange of policy information between a PEP and its corresponding PDP. This exchange is realized through three main messages: the request (REQ), decision (DEC) and report (RPT) messages. Hence, after a connection establishment between the PEP and its serving PDP, the PEP transmits requests for decisions to the PDP using the REQ message. In response to a REQ, a decision message (DEC) is sent by the PDP. Then, the PEP reports the outcomes to the PDP via the RPT message.

Initially, the COPS protocol was designed mainly for resource allocation in an Internet backbone. In order to make such a reservation, two models within the COPS protocol were proposed: the Outsourcing and Provisioning models. In the former policy-requests are triggered by particular events when in the latter policies are installed in the PEP before the PEP decides how to treat the event. For this second model, a specific COPS extension, called COPS PR [7], have been designed.

In order, to facilitate the resource allocation process and render the PBM approach more dynamic, the authors in [8] proposed using the COPS protocol to unify both QoS signaling and resource allocation. The idea behind this is that each end-system will encompass a specific PEP that have to initiate resource reservation requests. Requests are handled by the PDP which in its turn takes decisions according to current resource usage and customer's Service Level Agreement (SLA).

In our work, both QoS signaling by end-system and dynamic resource provisioning in edge routers are considered. In this case, each time a new reservation request is accepted by the PDP, the latter both provisions the concerned edge router accordingly (using COPS PR) and informs the end-system on its positive decision. For end-system QoS signaling, our proposed COPS extension is called COPS usage for QoS Parameter Signaling (QPS). More details on COPS QPS operations can be found on [9].

3 Dynamic QoS Setup using SIP with PBM

We are now recalling our new solution for the dynamic setup of QoS for multimedia applications in the context of PBM [1].

3.1 Functional Architecture

As shown in Figure 2, our solution is based on the integration of SIP proxy and QPS PEP. This environment allows relocating networks entities inside the terminal and then can be seen as a network extension. In [1] we demonstrate the advantage of this integration by comparing it with other proposals based on the combination of session initiation and QoS provisioning. Actually, one can note that the integration of the SIP proxy and the QPS PEP allows a complete control of session-setup by the network. Indeed, the SIP Proxy participates to every step of SIP exchange. Then, it knows all the information concerning the session. Furthermore, its implementation near to the QPS PEP automates the interactions between application's needs and network-resource allocation. As part of network management functionalities (QPS PEP and SIP Proxy) are deported into the user's terminal, these functionalities have to be ran within a secured environment (execution environment secured by smart card [1][10], for example). How to secure such an environment is not the target of our paper and will not be described.

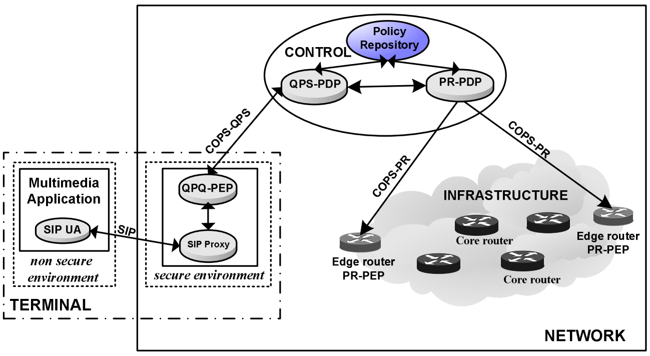

Figure 2. Functional Architecture

3.2 Session Establishment Sequence Diagram

The complete process of QoS-enabled session establishment process is detailed in this section (Figure 3). At the beginning, the UAC initiates a session by sending an INVITE message to its correspondent UAS. This message is intercepted by the client's SIP proxy. After having recovered the media information concerning the SIP session initiator, the SIP proxy forwards the message to the UAS. When the UAS replies with a 200 OK, the SIP proxy recovers the media characteristics concerning UAS. At this stage, it has all media information concerning the media session. The

SIP proxy is then responsible of translating this information into QoS parameters (bandwidth, delay, jitter and packet loss) and sending them to the QPS PEP. Using these QoS parameters, the QPS PEP sends a QoS request to the PDP. First of all the PDP consults the user's SLA and the resource availability before generating the appropriate decisions towards the concerned edge-routers and the caller's terminal.

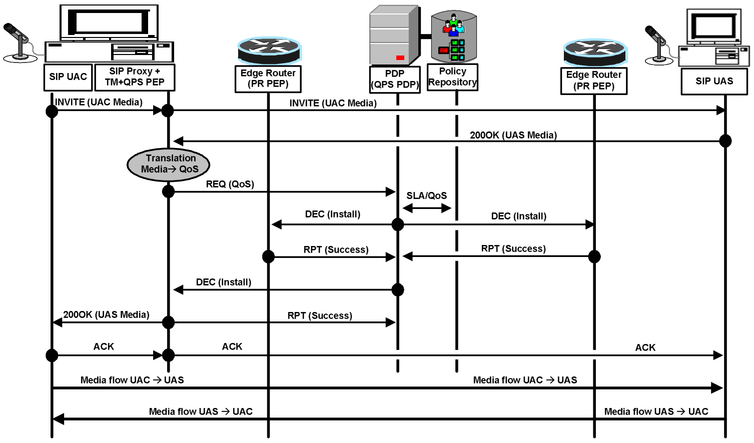

Figure 3. Session establishment phase with QoS setup.

Note that, each time a COPS QPS REQ message arrives to the PDP, the policy repository is accessed three times: the first is for the retrieval of the client's SLA, the second is to retrieve admission control policies and the last access allows retrieving topology information. This latter is realized in order to identify the edge-routers that will be crossed by media flows. The number of accesses realized after the reception of a COPS QPS REQ message depends on the structure of the policy repository that is a realization of the CIM model [6]. These accesses are optimized as we have three information to retrieve and we access these information directly. Indeed, the policy repository is realized as a Lightweight Directory Access Protocol (LDAP) [11] server. When using LDAP we have the ability to access information directly using their distinguished names (dn). Hence, in our architecture, we use, as distinguished names (dn), specific information related to terminal location (SubNet Address, Wireless Cell Identification or BSSID, …), that are sent as objects of the QPS REQ message. The use of dn accelerates substantially the delay of each information access [12].

The decision that is sent by the PDP to the crossed edge-routers resides on classifying and marking media-flow packets with the appropriate QoS class tag [13]. Once the successful enforcement of the decision is reported to the PDP, the latter sends the appropriate decision to the requesting QPS PEP. In the case of acceptance, the SIP 200 OK message is directed to UAC which terminates the session setup. The multimedia application can starts flow transmissions tacking advantage of the QoS-level assigned to it within the network.

4 Test-Bed

As we intend to verify the feasibility and scalability of our original solution, an integrated test-bed, containing all SIP and all PBM entities, is carried out. Unlike other experiments found in the literature [1], the COPS operations are not considered alone but all PBM architecture components are included to our test-bed. Indeed, the interactions between PDP and the policy repository, policy interpretation by the PDP, and real policy enforcement are implemented. The details of our test-bed building blocks are given bellow.

4.1. SIP Building Block

All SIP entities, listed above, have been implemented using the Java-based specifications called Jain-SIP [14].

- SIP User Agent Client: this component is the one that generates SIP requests and records call set-up delays (from the "INVITE" message up to the "200 OK"),
- SIP Proxy: in addition to its classical operations defined in SIP [2], it is in charge of the interconnection with the Translation Module (TM) (cf. Section 4.3),
- SIP User Agent Server: this component generates automatically the responses to UAC requests.

4.2. PBM and QoS Building Block

The set of PEPs, the PDP, and the COPS protocol have been implemented using Java. As denoted previously (cf. section 3.2), the policy repository is realized as a LDAP schema, where in the management information are modeled using CIM. The distinguished names (dn), in the LDAP schema, are chosen to correspond to objects transported by incoming COPS messages. This allows rapid and direct access to the management information that is needed to handle a new media flow.

In addition to PBM components, specific software, the Traffic Designer [15], is used for the enforcement of QoS decisions (traffic classification and packet marking) within edge routers.

4.3. Translation Module (TM) Building Block

- The TM building block is mainly responsible of converting the media information into QoS requirements. It interconnects SIP building block to PBM building block. More specifically, this building block allows converting and relaying information between the SIP proxy and the QPS PEP. In our test-bed, this is realized throughout a simple conversion table built thanks to some tests of the behavior of the used codecs. A more accurate mapping outwards the scope of this paper and will be a subject of a future work.

5 Tests and Measures

Our aim is to ensure that the delays of the session establishment due to our proposition remain always under the recommended ITU-T delay limit of 6s [4]. This limit is given for session setup involving less than four network-control entities. This is the case for our test-bed (Figure 4). Indeed, COPS QPS and SIP signaling messages cross only three network control entities: a SIP Proxy, a QPS PEP and a QPS PDP. The test-bed, previously presented, is then used to measure the observed delays introduced by each of our building blocks.

5.1. Scenario

The established test-bed, shown in Figure 4, allows us to set up multiple feasibility and scalability testing scenarios. Demonstrating the feasibility and scalability of our solution consists of analyzing the behavior of SIP and COPS, and verifying that the delay bound defined by the ITU-T is never exceeded. The feasibility testing scenarios reside on measuring this delay on both local and distant configurations.

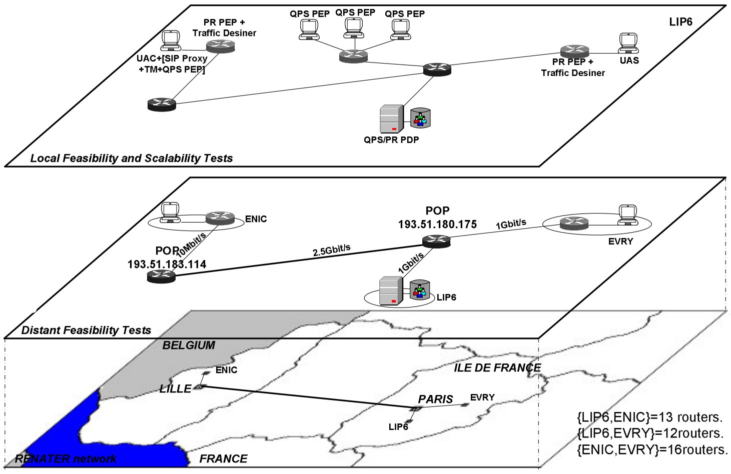

Figure 4. Test-bed

In the local configuration, all the test-bed components are in the same LAN. In contrast, for the distant feasibility tests, the test-bed components are deported on 3 different LANs interconnected over the public Internet (i.e. over the inter-university IP network called RENATER [16]). Hence:

- The initiator of the multimedia communication (SIP UAC + SIP Proxy + TM + QPS PEP) is connected through an edge router (PR PEP) to the RENATER Network. These components are located in ENIC premises in the city of Lille.
- The responder (UAS) is also connected through an edge router (PR PEP) to the RENATER Network. These components are located in IIE premises in Evry.
- The PDP is located in a third location which is the LIP6 laboratory in Paris.

For the distant feasibility test, the distance, in terms of number of router interconnecting each pair of sites, is also depicted in Figure 4. For this scenario, let's consider an important number of users relying QoS requests. As one SIP proxy is dedicated to each user and not shared, the only bottleneck of our solution becomes the PDP. The scalability testing scenarios consist then of measuring the session establishment delay when the PDP is overloaded. The idea is to measure the effect of increasing the QoS-request rate. This rate is increased by the periodical request generation by three additional terminals hosting a set of virtual PEPs (Figure 4).

Let's note that 30 iterative session-establishment demands are initiated by the SIP UAC in order to compute a statistically acceptable delay estimate. This is realized for each testing scenario.

5.2. Local Feasibility Tests

From Table 1, our attention is focused on the small overall measured delays, compared to the 6s ITU-T bound. Furthermore, the delay's standard deviation is quite low, confirming the accuracy of these experiments. This is not very surprising as these tests are down locally on the same sub-network and no concurrent QoS requests are handled in the same time by the PDP.

Table 1. Local feasibility tests: delay measurements

Delay (ms)	Min	Max	Mean	SD	CV (%)[3]
SIP	409	502	446	25.05	6
TM	2	18	6	3.32	55
PBM	137	424	238	59.81	25
Overall	571	859	690	63.86	9

Table 1 also shows that the simultaneously hold delays for PBM are 50% smaller than those of SIP. This is due to the format and the size of COPS and SIP messages that influence their treatment delays. Indeed, the SIP messages are of text type (said HTTP-like) and need a parser to be interpreted. Quite the opposite, the COPS messages handle objects and are treated by fastest API. To show the size difference between COPS and SIP messages, we can note that the size of COPS REQ is about 356 octets and that the size of SIP INVITE message is about 930 octets.

We can notice finally that the translation module delays are negligible in comparison to the global session establishment delays.

5.3. Distant Feasibility Tests

Let's recall for these tests, the test-bed components are located in three different interconnected LANs throughout the public Internet. The same measures as the previous feasibility test are realized (Table 2).

[3] The coefficient of variation (CV) highlights the relative dispersion of the measured delays.

Table 2. Distant feasibility tests: delay measurements

Delay (ms)	Min	Max	Mean	SD	CV (%)
SIP	441	1429	552	223.33	41
TM	2	22	5	4.47	89
PBM	138	846	282	173,03	61
Overall	638	2099	838	337.22	40

When comparing the results showed in table 2 with the results obtained in the previous section, one can note the significant increase of the maximum delay and the standard deviation for both SIP and PBM. But, when looking at the minimum and the mean delays, we can remark a little increase compared to the previous test. This latter is not surprising due to the fact that these delays are mainly influenced by the computational overhead rather than the experienced network delays that are minimal. By detailing these results, we note that only 3 SIP and 2 PBM exchanges have a delay more than 700ms and 400ms respectively. These excessive delays are those causing the significant increase of the maximum delay and the standard deviation. They are due to small-transient delay-increase in the network. However, note that the obtained overall delays are not so high; even the maximum is 2.099s and the standard deviation is relatively small. In this way, our solution's overall delays stay clearly bellow the critical limit.

5.4. Scalability Tests

In order to analyze the scalability of our solution, the load of the PDP is progressively increased. In parallel to this increase of load, session establishment demands are initiated. The observed delays for each entity are then measured and analyzed.

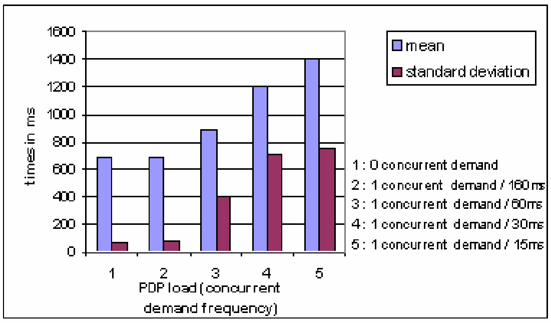

Figure 5. Overall delays in the scalability testing scenario

Figure 5 summarizes the results obtained in our scalability tests. It highlights the evolution of overall delays obtained for different PDP loads. In this curve, the obtained delays for loads that are smaller than "1 demand / 160ms" are skipped. This is justified by the fact that the corresponding overall-delay linearly grow with a small slope.

From Figure 5, one can note that for rates over 15 demands per second, both mean delay and standard deviation increase drastically. This rate corresponds to the limit after which the PDP load begins to influence clearly on its computational delays.

Table 3. Local scalability tests: delay measurements

Delay (ms)	Min	Max	Mean	SD	CV (%)
Policy Enforcement	10	41	28	9.11	33
Provisioning	128	2293	613	639.02	104
Resource Allocation	42	1693	367	479.04	131
SIP	359	458	415	18.57	4
TM	2	18	6	3.32	55
PBM	194	2752	980	744,03	76
Overall	610	3178	1400	746.39	53

Detailed delays obtained for the latter test, where 67 demands per second are generated by 3 different PCs emulating 300 connected users, are shown in table 3. This test confirms the non-surprising result related to the SIP operation delays. Indeed, these delays always remain below 500 ms, with a small standard deviation. This is naturally due to the fact that the bottleneck in our solution is not within the SIP part but concerns solely the PDP. However, the PBM operation delays are strongly increased compared to our feasibility tests. One can note that the provisioning and resource allocation delays are the parts that have the highest mean with a very important standard deviation. Indeed, the resource allocation and the provisioning procedures are both carried out by the PDP. The other observed delays remain stable. Let's however precise, that the maximum overall delay, in this latter experiment, is still under the 6s bound. These results consolidate the design philosophy adopted for our solution.

A load of 67 demands per second is quit high and the obtained results should give us a good estimate of the overall delays that would certainly be obtained for a realistic worst case. Indeed, this load can correspond to a worst case in a medium size network (eg. a corporation willing to use IP telephony between their different geographical sites). Future improvements of our structure particularly, the use of PDP replication, would lead us to estimate these delays for telecommunication WANs [17].

6 Conclusion

In this paper, we analyzed a novel architecture for multimedia session setup with QoS guarantees. This novel solution deals with the interoperation between SIP functionality and PBM components. Indeed, our proposal is to deport SIP proxy and a QoS Parameter Signaling (QPS) PEP within user's the terminal.

In order to measure the performances of our solution, a complete test-bed has been implemented. It includes all PBM, SIP and integration components. The experiments carried out allow us to highlight the properties of our solution in terms of feasibility and scalability. The feasibility of our solution allows us to find lower overall delay bound for session establishment that is significantly smaller than the upper delay bound recommended by the ITU. Scalability experiments allow us to demonstrate that the bottleneck of our architecture is the PDP. Undeniably, the slope of the PDP operation delay curve begins to drastically increase when the request rate exceeds 15 demands per second. However, tacking all our experiments, this delay always remains below the 6s ITU-T bound.

Note that the test-bed built is realistic for the case of multimedia communications within a corporate Intranet connecting multiple remote sites and aiming to control the resources of the corporate Virtual Private Network (VPN). In this case our solution gives very promising results.

To confirm these results for largest networks, improvements in our architecture (respectively, in our test-bed structure) are needed. These improvements are mainly dealing with PDP replication which will be the issue of our future works.

References

1. L. Bernard and H. Afifi, "QoS dynamique pour applications multimédia basée sur le couplage SIP-COPS et carte à puces," CFIP'03, (Paris, France), Oct. 2003.
2. D. C. Verma, "Policy-Based Networking–Architecture and Algorithms," New Riders Publishing, (Indianapolis, Indiana), Nov. 2000.
3. J. Rosenberg et al., "SIP: Session Initiation Protocol," RFC 3261, Jun. 2002.
4. ITU-T Recommendation No. E.721, "Network grade of service parameters and target values for circuit-switched services in the evolving ISDN," May 1999.
5. J. Boyle, et al, "The COPS (Common Open Policy Service) Protocol," RFC 2748, Jan. 2000.
6. CIM Specifications, "Common Information Model (CIM) Specification Version 2.8," DMTF Policy WG, Aug. 2003.
7. K. Chan, et *al*., "COPS Usage for Policy Provisioning (COPS-PR)," RFC 3084, Mar. 2001.
8. T.M.T. Nguyen, N. Boukhatem, Y.Ghamri-Doudane,G. Pujolle, "COPS-SLS : A Service Level Negociation Protocol for the Internet," IEEE Communication Magazine, May 2002.
9. Y.Ghamri-Doudane, "QoS Support and Management in Wireless Networks," PhD thesis, University of P. & M. Curie, Nov. 2003.
10. B. Zouari and H. Afifi, et al, "A novel authentification model based on secured IP smart card," IEEE International Conference on Communications, ICC'03, (Anchorage, Alaska), May 2003.
11. J. Hodges et R. Morgan, "Lightweight Directory Access Protocol (v3) : Technical Specification", RFC 3377, Septembre 2002.
12. X. Wang, H. Schulzrinne, D. kandlur, D. Verma, "Measurement and Analysis of LDAP Performance", International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS'2000), Santa Clara, CA, pp. 156--165, Janvier 2000.
13. K. Nichols, et al, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers," RFC 2474, Dec. 1998.
14. JAIN™ SIP API Specification: http://www.jcp.org
15. http://www.qosmos.net/EN/classification.htm
16. http://www.renater.fr/
17. ITU-T Recommendation No. E.500, "Traffic intensity measurement principles," Nov. 1998.