Dynamic QoS Configuration of a DVB-RCS Satellite Terminal for SIPbased Applications

Baptiste Jacquemin, Pascal Berthou, Thierry Gayraud, Cedric Baudoin

Dynamic QoS Configuration of a DVB-RCS Satellite Terminal for SIP-based Applications

B. Jacquemin[1, 2], P. Berthou[1, 2] and T. Gayraud[1, 2]
[1]CNRS ; LAAS ; 7 avenue du colonel Roche, F-31077 Toulouse, France
[2]University of Toulouse ; UPS, INSA, INP, ISAE ; LAAS ; F-31077 Toulouse, France
Phone +33 561336923, Fax +33 561336411, {jacquemin, berthou, gayraud}@laas.fr
C. Baudoin[3]
[3]Thales Alenia Space, 26 av. JF Champollion, BP 31037 Toulouse Cedex 1; France
Phone +33 534356817, Fax+33 534355560, cedric.baudoin@thalesaleniaspace.com

Abstract

Satellite systems were long-dedicated mainly to broadcast services as television whereas Internet access was offered by terrestrial networks as DSL technologies. However, the democratization of DVB-S terminal, the standardization of a return channel via satellite (DVB-RCS) and the significant breakthroughs leading to DVB-S2 standard have allowed them to represent an interesting alternative to terrestrial networks in remote areas. But, to be competitive, DVB-S2/RCS systems have to offer an appropriate support to the variety of current and future multimedia applications (VoIP, videoconferencing) by providing QoS guarantees in spite of its specific characteristics: a long transmission delay and a limited bandwidth.
In this context, this paper considers a DiffServ architecture, compliant with the SatLabs recommandations, in which three classes are considered at the IP layer to differentiate the traffic: EF, dedicated to real-time applications with strong time-constraints, AF used for non real-time applications and BE. Each Diffserv classe is mapped on a specific MAC queue to take benefit of the DAMA (*Demand Assignment Multiple Access*) mechanisms which allow the ST to dynamically request capacity to the NCC (*Network Control Center*).
The goal of this work is to configure dynamically the DiffServ classes of service of the Linux-based ST for interactive applications, as VoIP, by using the session descriptors contained in SIP messages. For that, a complete QoS architecture based on the communication between an enhanced SIP proxy and a QoS Server and/or an ARC (*Access Resource Controller*) has been designed, developed and evaluated on a satellite network emulation testbed (PLATINE), implemented in the SatIP6 and SatSIX IST projects.

1. Introduction

1.1 DVB-S2/RCS network architecture

The scenario, shown in Figure 1, gives an overview of the satellite network architecture in a mesh topology. The main components of the network are a geostationary satellite with onboard switching capabilities, Satellite Terminals (ST) which provide single PC or LANs with the access to the network, and a Gateway (GW) which allows the connection with Internet core networks. In this configuration, the GW plays the same role than the STs; it is thus represented by a ST allowing access to Internet. The uplink access from each ST is managed through DVB-RCS interfaces. STs and GW are boundary devices between the satellite and terrestrial links and play an important role in access to satellite resources and hence in QoS provisioning. Both devices implement IP routing and have an IP interface on the satellite segment, as IP serves as a common denominator between the satellite and terrestrial networks.
The return link access scheme in DVB-S/RCS systems is MF-TDMA (Multi-Frequency Time Division Multiple Access). The return link is segmented into portions of time and frequency (“superframes”), each of which is divided into timeslots (”bursts”) of either fixed or variable durations during which STs are able to transmit MPEG2-TS packets or ATM cells.
The entire satellite system control, especially STs’ synchronization and resource allocation, is performed by the Network Control Center (NCC). It periodically broadcasts a signaling frame, the TBTP (Terminal Burst Time Plan), which updates the timeslot allocation within a superframe between every competing ST.

This allocation can be dynamically modified depending on Satellite Terminals' requests thanks to a bandwidth on demand protocol called Demand Assignment Multiple Access (DAMA). It supplements the STs with the ability to frequently request capacities that fit their current respective traffic load to the NCC.

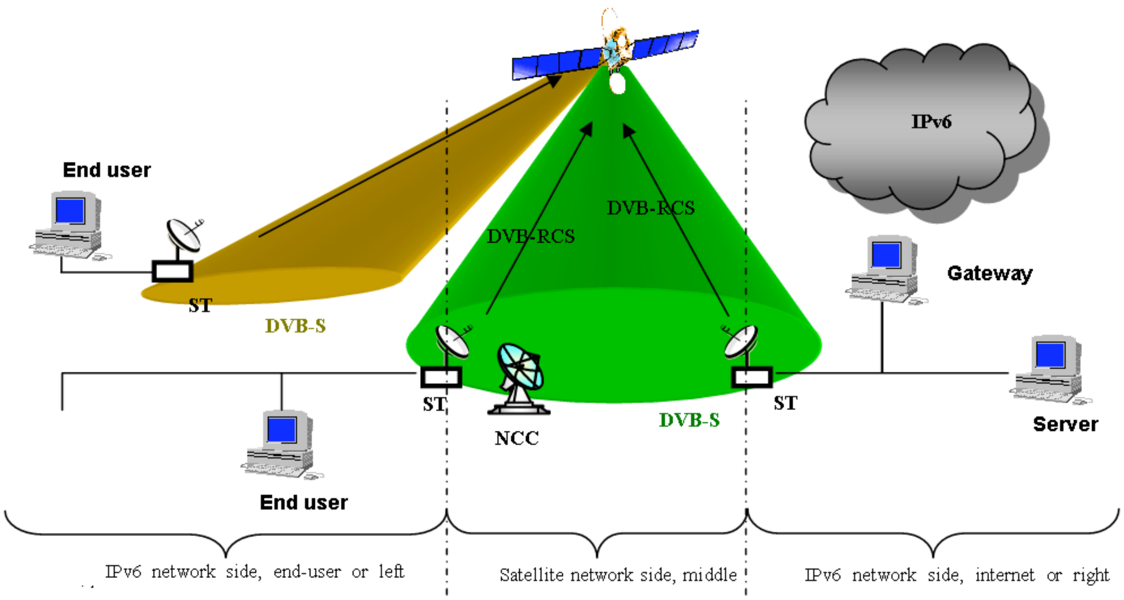


Figure 1 : DVB-S/RCS architecture

1.2 QoS in DVB-S2/RCS satellite systems

In accordance with SatLabs recommendations [1], our QoS architecture, developed under the SATSIX project [2], is based on a DiffServ architecture and on a mapping between layer 3 service classes and layer 2 request classes.
Three categories are then used at the IP layer: an EF class, an AF class divided into three subclasses (AF31, AF32 and AF33) and a BE class. Regarding QoS at the MAC level, three request classes are used: DVB-RT for real-time traffic, DVB-nRT for priority but non real-time traffic and BE for the rest, each class being associated with a specific logical channel. The EF class is mapped on the DVB-RT queue that has priority to use CRA capacity allocated for the whole duration of ST connection and has the ability to make RBDC request (but we will see that some mechanisms may also allow the dynamic modification of CRA). Then, AF classes are mapped to the DVB-nRT queue also associated to RBDC request but with a lower priority and finally, the BE class is mapped on the BE MAC queue associated to RBDC or VBDC request with the lowest priority. However, DVB-nRT and BE queues can use the statically allocated timeslots (CRA) when the RT queue is empty or does not use the whole fixed capacity.

1.3 QoS recommendations for VoIP or videoconferencing

As this study focuses on interactive applications such as VoIP and videoconferencing, their time-constraints have to be precisely analyzed and respected taking into account the long propagation delay (ST->satellite->ST) of a satellite system which is equivalent to 250 ms.
The ITU-T has then proposed a set of recommendations [1] concerning the parameters that those applications have to respect to work properly:

- A one-way delay lower than 400 ms
- A jitter lower than 1 ms
- A loss rate lower than 3%

As the propagation delay of a satellite system is very high, the first parameter to study is then the one-way delay of the video and audio flows and so, the rest of this paper will focus on this parameter.

2. Contributions

In this part, the several tools developed to configure dynamically the QoS of a DVB-RCS satellite terminal will be presented and as the QoS is more important for time-constrained applications, we will focus this study on SIP-based applications, SIP being one of the most important protocols able to control interactive session such as VoIP, videoconferencing, etc…

2.1.1 QoS-aware SIP proxy

The aim of this tool is to automate the resources reservation and make it transparent to users unable to choose the most appropriate classes of service to the different flows (audio and video for instance). In order to do that, the Session Initiation Protocol (SIP) [4] has been chosen because of its growing success in both public (open source code) and proprietary domains. The mechanism that we developed allows then configuring the QoS transparently by the analysis of SIP messages exchanged by applications. This analysis is made by an enhanced SIP Proxy (or QoS-aware SIP Proxy).

The enhanced SIP Proxy intercepts the session descriptors included in SIP messages, obtains the characteristics of each media involved in the session and pass it to a QoS Agent communicator in charge of (automatically) reserving and releasing QoS for the session via the QoS Server described in the following part.

Besides traditional SIP communication, additional functionalities are thus required:

- An SDP analyser making the Proxy able to analyse the session descriptions
- A table of medias updated during the session establishment. The medias negotiated between the caller and the callee are identified by a call-ID.
- A table of correspondence between a media and its characteristics. For exemple, the codec name used for a video flow has to be translated into usable parameters like maximum throughput, maximum delay, maximum jitter and maximum loss. To solve this problem, we develop a web-service based solution detailed in [5].
- An SDP/Diffserv mapping
- A QoS module (the QoS Agent communicator) which realizes the resources reservation associated to each media using the QoS Server.

In the proposed architecture, a QoS-Aware SIP proxy is deployed in each user LAN, behind each ST or GW. This distributed architecture is well suited for an access or mesh topology based on a regenerative satellite because it addresses the following two concerns:

- Scalability concerning flow QoS management in user LAN;
- Session establishment delays: the number of round trips of session and QoS signalling on the satellite link are minimized.

2.1.2 QoS Server

The QoS Server is an entity located on each ST/GW that receives information characterizing a flow sent by the associated SIP proxy. The exchange of resources reservation messages and resources release messages between the enhanced SIP proxy and its QoS Server is performed through a TCP connection over IPv4/IPv6 to assure a reliable and ordered message delivery and the protocol used to communicate is based on XML.

The messages exchanged by the enhanced SIP Proxy with the QoS Server indicate the type of message (RSV, FREE), the source address, the source port, the destination address and the destination port which are required to the identification of a specific connection. Then, to configure the DiffServ class of service, the messages contain also the characteristics of the media (maximum bitrate, the maximum delay, the maximum jitter and the maximum loss) and the wished class of service. An example of a reservation message is shown on the Figure 2 : Reservation message

Thanks to those characteristics, the QoS Server will be able firstly to redirect the concerned flow to the class of service indicated in the reservation message and also to dynamically reconfigure the size of concerned classes of service. In the example given in the Figure 2, the audio flow has to be configured for the EF class (EF corresponds to the service identified by “1”) and its maximum bitrate is 256 kbps. The QoS Server will then configure consequently the DSCP header in each IP packets corresponding to this flow, and, in the same time, will increase the size of the EF queue by 256 kbps and decrease the size of BE queue by 256 kbps. All those configurations are made by using the Traffic Control (TC) tool integrated into most current Linux kernels.

```
<?xml version="1.0" encoding="UTF-8"?>
    <XMLQoSMessage>
          <Sender>SIPProxy</Sender>
          <Type type="RSV"/>
          <Connection
                 BitRate="256.0"
                 CallID="a6893dd52390eb0dd7d4105d8632d40f@2001
                        :660:6602:102:8190:1f07fb574:2adb«
                 IPDst="2001:660:6602:104:c8ad:2eaf:5808:3e44"
                 IPSrc="2001:660:6602:103:e489:a343:5f66:3acf"
                 IPVersion="IPv6"
                 MaxDelay="400"
                 MaxJitter="1"
                 MaxLoss="3"
                 MediaType="audio"
                 PortDst="2502"
                 PortSrc="14502"
                 Service="1"/>
    </XMLQoSMessage>
```

Figure 2 : Reservation message

2.1.3 Access Resource Controller (ARC)

Another entity used in the satellite system to realize the control admission procedure is called the Access Resource Controller (ARC) and, in this part, we will explain how this entity can be used to dynamically modify the quantity of CRA allocated to an ST after detailing in which kind of situation, this mechanism is necessary.

This issue comes from the fact that on-demand resources (with RBDC capacity request for example) allocated to an ST when a new flow begins are not always sufficient to allow a proper functioning, particularly for time-constrained applications. In this case, we choose to increase the quantity of fixed capacity (CRA) to a given ST during a communication that needs this kind of service. As this study is focused on SIP applications, the same reservation message sent to the QoS Server by the SIP Proxy is transmitted to the ARC that realize the CAC procedure and modify the DAMA server to allocate a fixed capacity equivalent to the maximum bitrate of the used codec for the whole duration of the SIP session, if resources are available on concerned ST.

3. Evaluations

In this part, we will firstly describe the testbed on which our contributions have been developed and integrated and our experimentations realized. Then, the evaluations will be presented and analyzed precisely.

3.1 PLATINE: the satellite system emulation testbed

The Satellite network emulation testbed (PLATINE) has been initially developed in the SATIP6 IST project. It is compliant with the architecture adopted within the ETSI BSM [6] group and the DVB-RCS and DVB-S2 standards. It is able to emulate complex scenarios for next generation satellite network. Each network element involved in the satellite network is emulated in the platform on a dedicated node. In fact, 3 users' LANs of one or two nodes (standard Linux/Windows systems) are connected to the emulated satellite network with 3 ST/GW (standard Linux system) that implement an almost complete DVB-S2/RCS stack. The satellite core network is emulated thanks to the Satellite Emulator (SE) as link emulator and the Network Control Center (NCC), collocated with the GW, for bandwidth management (DAMA). 9 computers are used as described in Figure 3.

In order to have the most modular platform and so preserves room for future evolution (GSE), stringent requirements were fixed before the development phase.

The emulation testbed takes advantage of a linux system (Fedora Core 5) which natively supports IPv6 and a wide panel of IPv6 applications (Apache as HTTP Server, Mozilla as HTTP Client, Vsftpd as FTP Server, Gnomemeeting for Videoconferencing, VideoLanClient for Videostreaming), as well as advanced network and QoS features.
The different building blocs of the testbed are the following:

- The satellite carrier package is responsible for the different satellite carriers emulation on top of Ethernet (DVB-RCS, DVB-S2 and Signaling Channels) and the simulation of typical satellite bit errors and delay
- The DVB-S2/RCS package implements a framing structure compliant with the DVB-S2/RCS standards and fills DVB-RCS frames with upper layer packets (ATM or MPEG2-TS) coming from the ENCAP bloc layer. In order to achieve proper QoS, this layer manages synchronization and queues according to the authorizations a DAMA algorithm delivers.
- The DAMA package implements the DAMA algorithms used to manage the satellite resources allocation at layer 2 taking into account adaptive physical layer information.
- The ENCAP package implements AAL5 and ULE encapsulation schemes, and is in charge of the segmentation and reassembly functionalities (ATM or MPEG2-TS).
- The IP QoS Package implements common mechanisms to enable differentiation at this level. It mostly relies on QoS services offered by Linux kernel, retrieves incoming packets from IP network with their associated tag and forwards them to the lower layer.

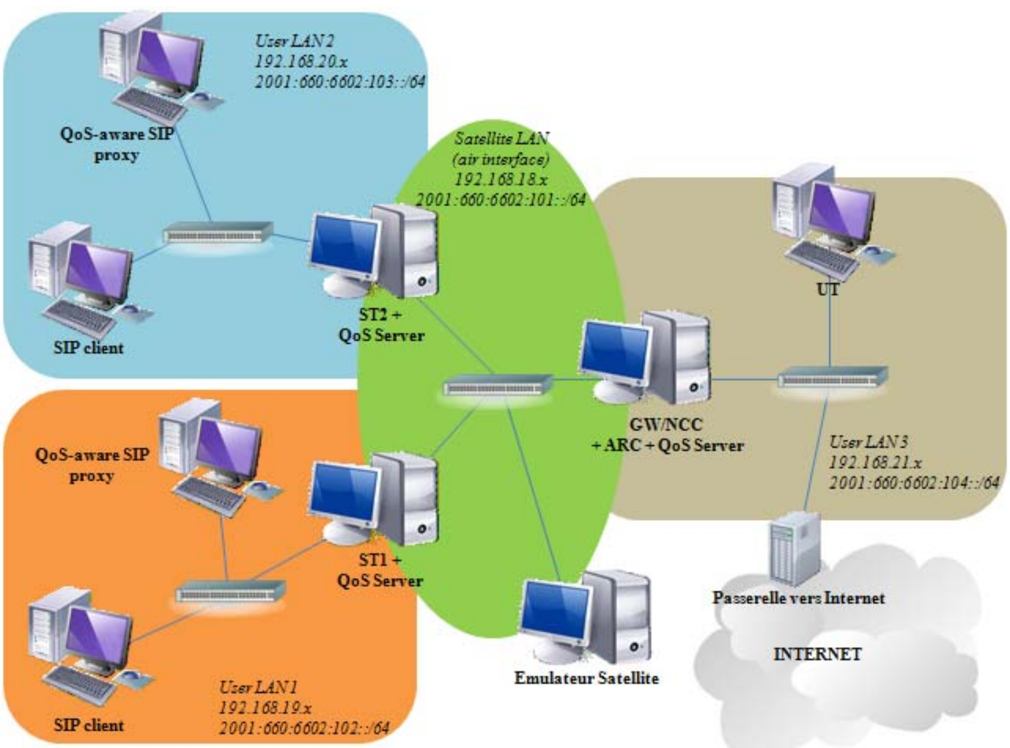


Figure 3 - Satellite emulation testbed

The QoS tools that we have implemented are then integrated into the satellite emulation testebed :

- A QoS Server, developed in C++, located on each ST/GW to mark packets and reconfigure the size of DiffServ classes
- An ARC, developed in C++, located on the GW/NCC to dynamically reconfigure the quantity of CRA allocated to an ST.
- A QoS-aware SIP proxy on the the User LANs 1 and 2 that intercepts SIP messages and send reservation/release messages to the QoS Server and the ARC.
- A SIP client (VisioSIP) located on the User LANs 1 and 2 to initiate the SIP sessions.

3.2 The experimental scenario

This section presents the experimental scenario of the different evaluations that have been realized during our test.

Firstly, the total bandwidth of the ST considered for the evaluations, the ST1, is 1 Mbps. At the IP level, three DiffServ classes are considered: EF, AF and BE, the AF and EF queues having an initial size of 0 kbps whereas the BE queue has the whole capacity : 1Mbps.

The scenario is then :

- At t = 10s, a videoconferencing SIP session (audio+video) is initiated.
- At t = 60s, a first UDP concurrent flow of 500 kbps starts.
- At t = 120s, a second UDP concurrent flow of 500 kbps starts. The ST1 is now overloaded.
- At t = 180s, a third UDP concurrent flow of 500 kbps starts.
- At t = 240s, the UDP flows stops.
- At t = 300s, the SIP session ends.

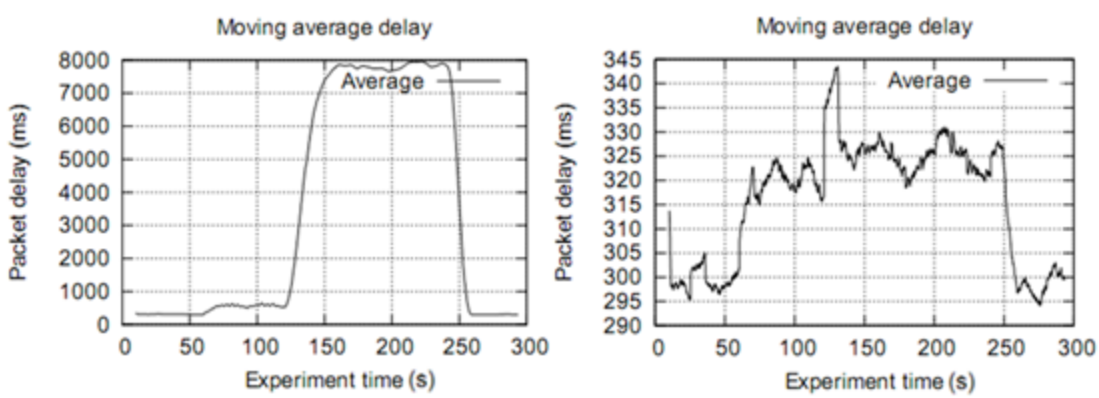
The three concurrent flows are configured to use the BE queue on the ST1 and the video and audio flows are configured to use the EF class when the QoS mechanisms are activated and the BE class when deactivated.

3.3 Evaluation of the interaction between the enhanced SIP Proxy and the QoS Server

In this evaluation, we consider that the CRA capacity allocated to the ST1 is sufficient to support both audio and video flows.

To evaluate our QoS mechanisms based on the interaction between our enhanced SIP Proxy and the QoS Server, we compare the cases where the videoconferencing SIP session starts without QoS (no enhanced SIP proxy and no QoS Server) or with QoS, that is to say that the QoS Server is marking the video and audio flows to go to the EF queue and that it reconfigures the size of the EF and BE queues according to the maximum throughputs of the flows transmitted by the enhanced SIP Proxy.

Moreover, we will make the analysis on the audio delays graphs presented on the Figure 4, but the same analysis could be applied to the video delays graphs that are similar.



(a) Without QoS **(b) With QoS**

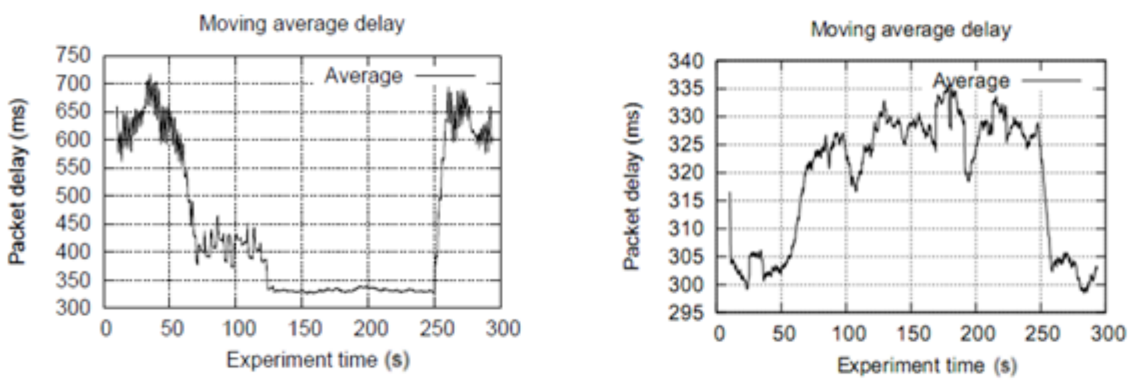
Figure 4 – Impact of the QoS mechanisms initiated by SIP on moving average delay.

These two series of delays' graphs show a real benefit of the IPv6 QoS usage and a fair separation of the classes of service can be observed on the Figure 4.b. Detail analysis of those graphs is now provided: First, concerning the comparison of the graphs with and without QoS, it can be observed a real improvement when the QoS architecture is running especially when background traffic (concurrent UDP flows) is high: The "moving average delay" graphs show that when two or three concurrent flows are running (between 120 and 240 s), a very high increase of average delay is experienced by the audio flow when the QoS is not set (until 8 seconds delay) while the average delay remains below 345 ms when the

QoS is set, which is compatible with audio conference requirements (< 400 ms). In the case of high load on the satellite return link, the impact of the QoS architecture is clearly shown here.
When no concurrent flows are running, delay for the audio flow is around 300ms in both cases (with and without QoS), cf. graphs between 0 and 60 seconds. This can be explained by the fact that all CRA resources, in this case, are used by the multimedia flows and no on-demand capacity is needed. When just one concurrent UDP flow of 500kbps is running the delay of VoIP application is increasing in both cases but very slightly when QoS is set (from 300 ms up to 320 ms on average) while it's increasing up to 500 ms on average in the case where no QoS is set. The capacity of the channel should be enough for both flows but the CRA capacity is not enough and on-demand RBDC bandwidth is required. So the audio flow experiences more delay when QoS is not set; this is due to the fact that all flows (audio, video and best-effort flows) are using the same MAC buffer and logical channel, and so the same delay is experienced by all packets in this buffer, implied by the capacity allocation scheme. When the QoS is set, a different MAC buffer and logical channel is used for high priority traffic (audio and video packets) and is served first compared to the low priority MAC buffer. Consequently, the audio flow is protected and the delay is increasing very slightly: it's experiencing an end-to-end delay compatible with audio conference application requirements (under 400ms).
Secondly, concerning the classes of service separation, we can notice on the second graph (b) with QoS that the impact on high priority classes of service of concurrent flows is rather low, and does not degrade the overall quality for end-to-end users whatever the load of the ST1: the delay remains below 400ms which is acceptable for interactive audio conference applications.

3.4 Evaluation of the ARC

In this evaluation we consider the same scenario but no CRA capacity is allocated to the ST here, so all the capacity is given with RBDC requests. To evaluate our proposition based on the ARC, we will then compare two cases where QoS is set (EF marking of audio and video flows and reconfiguration of DiffServ classes) but in the first case, no dynamic reconfiguration of CRA capacity is done whereas in the second case, the interaction between the enhanced SIP Proxy and the ARC is possible.
The Figure 5 shows the results of this evaluation for audio graphs which is similar to the video graph.



(a) Without CRA **(b) With dynamic reconfiguration of CRA**

Figure 5 – Impact of the dynamic reconfiguration of CRA on moving average delay.

These experiments show the impact of the DAMA algorithm on the interactive applications. On Figure 5.a, the delay experienced by the audio stream is initially between 600 and 700 ms (this is the same values for video stream) and decreases when concurrent flows start. The first noticeable thing is that the DAMA algorithm implemented on PLATINE works fine with audio and video streams. The delay stays stable, around 650 ms but remains not compatible with the ITU-T recommendation on audio and video flows. The second noticeable thing is the delay diminution that occurs each time a new concurrent flow starts (at t=60 and t=120s) until the total capacity of the ST be reached. This can be explained because the videoconferencing application takes benefit from the RBDC requests made for background traffic as video and audio traffic have a better priority.
The first conclusion is that, even if RBDC requests works fine, it is not sufficient to ensure a proper QoS to audio and video flows because the ITU-T recommendations (delay < 400 ms) are respected only in the

case where the ST is overloaded and we cannot assume that an ST be permanently overloaded. It is then necessary to modify dynamically the CRA capacity allocated to a given ST when it is possible of course and the Figure 5.b presents the graph of moving average delay when the enhanced SIP proxy realizes a demand to the ARC to increase the CRA capacity during the whole SIP session. We can notice that, in this case, the delay is always lower than 350 ms, whatever the ST's load and whatever the initial configuration of CRA capacity.

4. Conclusion and Future Work

In this paper, we have then presented new mechanisms to automate the QoS configuration of videoconferencing SIP session on a DiffServ ST by allowing communication between an enhanced SIP Proxy, a QoS Server and the ARC. The evaluations, realized on a satellite emulation testbed, show that those mechanisms bring important features as it allows configuring dynamically the resource by using the SIP signalization to initiate QoS reservation/release and it ensures that, if the system resources are sufficient and if the SIP user is authorized to use this kind of service, the session will occur with QoS guarantees whatever the load and the initial configuration of the ST.
Our architecture is all the more interesting as SIP has been chosen as call control protocol by IMS (IP Multimedia Subsystem) [7] which is the most currently studied NGN (Next Generation Network) architecture and so, it indicates that the QoS mechanisms described in this paper could be adapted to the IMS architecture by adding the new features of our enhanced SIP Proxy to the SIP P-CSCF and by allowing an interaction with PEP/PDP that have quite similar role than QoS Server/ARC. It would be then very interesting to study how our architecture could be adapted to the IMS one and how a DVB-S2/RCS satellite system could be integrated to the IMS architecture. It could allow to configure a real end-to-end QoS to users of SIP services.

5. References

[1] SatLabs System Recommendations – Quality of Service Specifications, June 2008.

[2] IST SATSIX project, http://www.ist-satsix.org/

[3] ITU-T-Rec. G.1010, “End-user Multimedia QoS Categories”, Technical Report, International Telecommunication Union, 2001.

[4] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley and E. Schooler “SIP: Session Initiation Protocol”, IETF RFC 3261, June 2002.

[5] M. Gineste, B. Jacquemin, P. Berthou, A. El Fatni, T. Gayraud and C. Baudoin, “A Web Services Based Resource Signaling Scheme in Multimedia Satellite Systems”, 4th Advanced Satellite Mobile Systems Conference (ASMS'2008), August 2008.

[6] TR102157 v1.1.1. ETSI TC SES; Broadband Satellite Multimedia; IP Interworking over satellite; Performance, Availability and Quality of Service, July 2003

[7] G. Camarillo, M.A. Garcia-Martin. The 3G IP Multimedia Subsystem. West Sussex: Wiley, 2004