

QoS REQUIREMENTS FOR MULTIMEDIA SERVICES

Editors: José Ignacio Moreno Novella¹, Francisco Javier González Castaño²

Contributors: Rafael Asorey Cacheda², Daniel Castro García³, Antonio Cuevas¹, Francisco Javier González Castaño², Javier Herrero Sánchez³, Georgios Koltsidas⁴, Vincenzo Mancuso⁵, José Ignacio Moreno Novella¹, Seounghoon Oh⁶, Antonio Pantò⁷

¹UC3M - Universidad Carlos III de Madrid, Spain

²UVI - Universidad de Vigo, Spain

³INFOGLOBAL, Spain

⁴AUTh - Aristotle University of Thessaloniki, Greece

⁵UToV - Università degli Studi di Roma “Tor Vergata”, Italy

⁶RWTH - Rheinisch -Westfälische Technische Hochschule Aachen, Germany

⁷CNIT - University of Catania, Italy

3.1 Introduction

Internet development and an ever-increasing demand for bandwidth are boosting the market for satellite solutions. Technological progress leading to new satellite capabilities and the availability of bandwidth at lower cost is

enabling this growing role of satellites in the Internet world. Satellite solutions are being used for both broadcast/multicast applications and point-to-point services. End-user access combines multicast and point-to-point services while content distribution to the “*edge*” of the Internet (i.e., to service providers’ points-of-presence serving access local loops) is a true multicast application.

Geostationary Earth Orbit (GEO) satellites and *Low Earth Orbit* (LEO) constellations essentially play a complementary role, in order to provide this complete range of services. Due to the large amount of capacity they provide and their low-latency characteristics, LEO systems are very well suited for point-to-point high-quality services while GEO solutions are very efficient for both broadcast/multicast offerings and access services including a significant percentage of multicast data. To support the different services it is important to consider their *Quality of Service* (QoS) requirements.

This Chapter mainly describes QoS requirements for multimedia services based on international standards. Section 3.2 shows a classification of applications according to error and delay tolerance, as well as performance characterization of traditional and multimedia applications. This work is based on the ITU G.1010 [1] standard that has been adopted by other standardization bodies like 3GPP. Section 3.3 presents main QoS support models over IP networks, while Section 3.4 shows main concepts for the transmission of multimedia and broadcast services over satellite networks. Finally, Section 3.5 presents experimental results of application performance over a real platform; the main interest here is to present QoS results on classical and emerging applications.

3.2 Services QoS requirements

Nowadays it is very important to support QoS in telecommunication systems, considering the requirements that should be met when a service is provided. This task should take into consideration that a user is not interested in the way a particular service is provided, but in the service quality level he/she finally delectates.

QoS refers to the capability of a telecommunication system to provide better service to selected traffic over heterogeneous networks (technologies or domains). The primary goal of QoS is to provide priority, including dedicated bandwidth, controlled jitter and latency (required by some real-time and interactive traffic), and improved loss characteristics. Moreover, it is important to assure that providing priority for one or more flows does not cause the failure of other flows. On intuitive level, QoS represents a certain type of requirements to be guaranteed to the users (e.g., how fast data can be transferred, how much the receiver has to wait, how correct the received data is likely to be, how much data is likely to be lost, etc.).

QoS requirements for multimedia traffic have been covered by different standardization groups, like ITU, ETSI or 3GPP. The main work provided by

ITU is in Recommendations Y.1541 [2], F.700 [3], and G.1010 [1]. Applications have been classified in eight groups, according to the error tolerance and delay, as summarized in Figure 3.1 [1],[4].

Error tolerant	Conversational voice and video	Voice/video messaging	Streaming audio and video	Fax
Error intolerant	Command/control (e.g., Telnet, interactive games)	Transactions (e.g., E-commerce, WWW browsing, Email access)	Messaging, Downloads (e.g., FTP, still image)	Background (e.g., Email arrival)
	Interactive (delay <<1 sec)	Responsive (delay ~2 sec)	Timely (delay ~10 sec)	Non-critical (delay >>10 sec)

Fig. 3.1: End-user QoS categories mapping. This figure is reproduced with the kind permission of ITU.

Referring to the above Figure, it is possible to consider the following values on the ordinate axis for what concerns the error rates:

- Error tolerant applications
 - Conversational voice/video *Frame Erasure Rate* (FER) < 3%
 - Voice/video messaging FER < 3%
 - Streaming audio/video FER < 1%
 - Fax *Bit Error Rate* (BER) < 10^{-6}
- Error intolerant applications
 - Information loss = 0.

The ETSI *Broadband Satellite Multimedia* (BSM) [5] working group provides technical reports and standards establishing a framework to specify QoS requirements for broadband satellite networks based on the Internet protocol suite. These standards (following those developed in ETSI and other bodies) identify how Internet quality-related standards can be adapted, translated or made transparent to satellite transmission protocols and equipment. Some of the results of this standardization work have been the definition of the protocol stack architecture shown in Chapter 1 (Section 1.5), where lower layers depend on satellite system implementation (*satellite-dependent layers*) and higher layers are those typical of the Internet protocol stack (*satellite-independent layers*).

The traffic classes established by BSM are based on ITU-T, Tiphon, 3GPP, and UMTS decisions, with adaptation to the satellite environment. In particular, the BSM standards deal with variable link layer conditions, high asymmetry and higher delay that are characteristics of satellite networks. The aim is to enable the satellite network and the *Internet Service Provider* (ISP) to ensure acceptable QoS levels and to relate these issues to the BSM architecture for broadband systems.

In UMTS and, by extension, in satellite networks, four basic *service classes* (layer 7) are defined [4]: *conversational*, *streaming*, *interactive* and *background*. It is interesting to note that there is no strict one-to-one mapping between these service classes and the namesake traffic classes (layer 2) [6]: an interactive application can very well use a bearer of the conversational traffic class, if the application/service or the user has tight requirements on delay. In the following sub-Sections the performance requirements for all four service classes are investigated from the user perspective.

Note that the delay values in the Tables of the following sub-Sections represent one-way delay (i.e., from originating entity to terminating entity).

3.2.1 Performance requirements for conversational services

The most common service in this category is real-time conversation, such as telephony speech. *Voice over IP* (VoIP) and video conferencing also belong to this category, with increasing relevance as the Internet is rapidly evolving. This is the only class whose characteristics are strictly determined by human perception (senses). Thus, this scheme has the most stringent QoS requirements: the transfer time should be low and, at the same time, the temporal relation of information entities of the stream should be preserved. The limit for acceptable transfer delay is very strict (failure to provide low transfer delays will result in unacceptable lack of quality). However, there are loose requirements on FER, due to the human perception. For real-time conversation, the fundamental QoS characteristics are:

- Preserving the temporal relation of information entities in the same stream;
- Conversational pattern (stringent and low delay).

Some application examples based on conversational services are: conversational voice, videophone, interactive games, two-way control telemetry and Telnet. Table 3.1 summarizes these applications providing the explicit requirements for each of them [1],[4].

Conversational voice

Audio transfer delay requirements [3] depend on the level of interactivity of end-users. To preclude difficulties related to the dynamics of voice communications, ITU-T Recommendation G.114 specifies the following general limits

Medium	Application	Degree of symmetry	Data rate	Key performance parameters and target values		
				End-to-end one-way delay	Delay variation within a cell	Information loss
Audio	Conversational voice	Two-way	4-25 kbit/s	< 150 ms preferred < 400 ms limit	< 1 ms	< 3% FER
Video	Videophone	Two-way	32-384 kbit/s	< 150 ms preferred < 400 ms limit Lip-synch: < 100 ms		< 1% FER
Data	Telemetry-two-way control	Two-way	< 28.8 kbit/s	< 250 ms	NA	Zero
Data	Interactive games	Two-way		< 250 ms	NA	Zero
Data	Telnet	Two-way (asymmetric)		< 250 ms	NA	Zero

Table 3.1: End-user performance expectations - conversational services.

for one-way transmission delay (assuming that echo control has been applied) [7]:

- 0 to 150 ms: preferred range (below 30 ms the user does not notice any delay at all, whereas above 100 ms the user does not notice delay if echo cancellation is provided and there are no distortions in the link)
- 150 to 400 ms: acceptable range (but with increasing degradation)
- Above 400 ms: unacceptable range

We should remember here that there are three types of satellite systems: LEO, MEO and GEO. Due to their different distance to Earth's surface, the propagation delay for the transmitted signal (from Earth to the satellite and back to Earth) varies from 10 ms to 250 ms (see Section 1.2). This means that for LEO and MEO satellite systems the preferred range described above is achievable. However, a GEO system cannot achieve an end-to-end delay below 250 ms. This means that, according to the satellite system used, the network designer should be very careful when selecting operational modes. Other classes have looser requirements and they may be supported by GEO

satellites.

The human ear is highly intolerant to short-term delay variation (*jitter*), so it should be kept really low. It has been suggested that 1 ms is an adequate limit. However, the human ear is tolerant to moderate distortion of the speech signal. An acceptable performance is typically obtained with FER up to 3%. Finally, a connection for a conversation normally requires the allocation of symmetrical communication resources.

Videophone

Videophone requires a full-duplex system, carrying both video and audio, and it is intended for a conversational environment. Therefore, the same delay requirements of conversational voice will apply, i.e., no echo and minimal effect on conversational dynamics, with the added requirement that audio and video must be synchronized within certain limits to provide “*lip-synch*” (i.e., synchronization of the speaker’s lips with the words the end-user hears). In fact, it will be difficult to meet these requirements, due to the long delays incurred in video codecs. Human eye is tolerant to some information loss, so that some degree of packet loss is acceptable. It is expected that high performance video codecs will provide acceptable video quality with FER up to about 1%. In satellite networks, the same considerations for conversational voice hold in this case.

Interactive games

Interactive games are games that use the network to interact with other users or systems. Requirements for interactive games are very dependent on the specific game considered in terms of bandwidth and delay. Many interactive games try to exchange high volumes of data, but demand very short delays, and a delay of 250 ms is reasonable.

Two-way control telemetry

Telemetry is a technology that allows the remote measurement, operation and reporting of information of interest. Two-way control telemetry is included here as an example of a data service that does require real-time conversational performance. Two-way control implies very tight limits on allowable delay and a value of 250 ms is proposed, but a key difference with voice and video services is that information loss cannot be tolerated. It is well known that the satellite channel is error-prone and in order to achieve zero information loss we need sophisticated error control techniques to ensure it. Delay is a relative issue for this class of traffic. As far as a satellite network can meet the deadlines that a particular telemetry service imposes, it can support that service.

Telnet

Telnet (*TELEtype NETwork*) is a network protocol used on the Internet or local area network connections. In this context, Telnet refers to the program that provides the client part of the protocol. It allows a remote server access. Due to the interactivity of the program, Telnet needs a low delay to allow a user perception of interactivity. This application is included here with a requirement for a low delay in order to provide back instantaneous character echoes. By extension we could consider in the same service/application group any remote access applications like *rlogin* (*remote login*) or *ssh* (*secure shell*).

3.2.2 Performance requirements for interactive services

This second class comprises interactive services (i.e., a human or a machine request on-line data from a remote server). It is characterized by the request-response pattern of the end-user. An entity at the destination is usually expecting a response message within a certain period of time. The *Round Trip propagation Delay* (RTD) time is therefore one of the key attributes. Another characteristic is that the content of the packets must be transparently transferred (with a low BER). The resulting overall requirement for this communication scheme is to support interactive non-real-time services with low RTD.

For interactive traffic, the fundamental QoS characteristics are:

- The request-response pattern;
- Preserving payload content.

Some examples of this service type are: voice messaging and dictation, data, Web-browsing, high-priority transaction services (e-commerce) and e-mail (server access). The corresponding requirements are summarized in Table 3.2 [4].

Voice messaging and dictation

The requirements for information loss are essentially the same as for conversational voice, but, on the contrary, there is more tolerance to delay since there is no direct conversation involved. Therefore, the main task becomes to determine the delay that can be tolerated between the user, issuing a command to replay a voice message, and the actual start of the audio. There is no precise data on this, but a delay in the order of a few seconds is considered to be reasonable for this application.

Web-browsing

The main performance factor is the visualization response time, after a Web page has been requested. A value of 2-4 s per page is proposed. However, a decrease up to a target of 0.5 s would be desirable.

Medium	Application	Degree of symmetry	Data rate	Key performance parameters and target values		
				One-way delay	Delay variation	Information loss
Audio	Voice messaging	Primarily one-way	4-13 kbit/s	< 1 s (playback) < 2 s (record)	< 1 ms	< 3% FER
Data	Web-browsing - HTML	Primarily one-way		< 4 s/page	NA	Zero
Data	Transaction services - high priority e.g., e-commerce, ATM	Two-way		< 4 s	NA	Zero
Data	E-mail (server access)	Primarily one-way		< 4 s	NA	Zero

Table 3.2: End-user performance expectatives - interactive services.

3.2.3 Performance requirements for streaming services

This service class is mainly unidirectional with high continuous utilization (few idle/silent periods) and low time variation between information entities within a flow. However, there is no strict limit for delay and delay variation, since the stream is normally aligned at the destination. Additionally, there is no strict upper limit for the packet loss rate.

For real-time streams, the fundamental QoS characteristics are:

- Unidirectional continuous stream;
- Preserving time relation (variation) between information entities of the stream.

The resulting overall requirement for this communication scheme is to support real-time streaming services with continuous unidirectional data flows. Table 3.3 details some application examples and the corresponding limitations [4].

Note that Figure 3.1, Table 3.1, Table 3.2 and Table 3.3 derive from 3GPP TS 22.105 [4]. 3GPPTM TSs and TRs are the property of ARIB, ATIS, ETSI, CCSA, TTA and TTC who jointly own the copyright in them. They are subject to further modifications and are therefore provided “as is” for information purposes only. Further use is strictly prohibited.

Medium	Application	Degree of symmetry	Data rate	Key performance parameters and target values		
				Start-up delay	Transport delay variation	Packet loss at session layer
Audio	Speech, mixed speech and music, medium and high quality music	Primarily one-way	5-128 kbit/s	< 10 s	< 2 s	< 1% Packet loss ratio
Video	Movie clips, surveillance, real-time video	Primarily one-way	20-384 kbit/s	< 10 s	< 2 s	< 2% Packet loss ratio
Data	Bulk data transfer/retrieval, layout and synchronization information	Primarily one-way	< 384 kbit/s	< 10 s	NA	Zero
Data	Still image	Primarily one-way		< 10 s	NA	Zero

Table 3.3: End-user performance expectations - streaming services.

Audio streaming

Audio streaming is expected to provide better quality than conventional telephony, thus the packet loss requirements will be correspondingly tighter. However, there are no conversational elements involved and the delay requirements can be relaxed.

One-way video

The main distinguishing feature of one-way video is the absence of conversational elements. Therefore, the delay requirements will be not so stringent.

Still image

Regarding still images, the human eye is tolerant to information loss. However, single bit errors can cause large disturbances in still image formats. Therefore, it is generally expected that there will be zero errors in the transmission of still images. Delay requirements are low.

3.2.4 Performance requirements for background services-applications

This service class applies when the end-user, typically a computer, sends and receives data files in background. It is a classical data communication scheme where the destination is not expecting data within a certain deadline. Hence, the propagation delay (like that of satellite systems) is not that important in this case. However, error control is very important, since errors should be kept to very low levels (in the satellite scenario such feature calls for adequate coding protection and retransmission schemes).

For background traffic, the fundamental QoS characteristics are:

- The destination is not expecting data before a certain deadline;
- Preserving payload content.

The resulting overall requirement for this communication scheme is to support non-real time services without any special requirement on delay. A background application has no delay constraint. In principle, an essentially error-free delivered information is the only requirement for applications in this category. However, there is still a delay constraint, since data is effectively useless if it is received too late. Examples of these applications are: fax, low priority transaction services, e-mail (server to server), *Short Message Service* (SMS), download of databases and measurement records.

Fax

Fax is not normally considered a real-time communication. Nevertheless, there is an expectation that a fax transmission will take less than 30 s.

Low priority transaction services

An example in this category is SMS. An acceptable delivery delay is 30 s. Table 3.4 compares the applications on the basis of the service class and the associated delay requirement [8].

3.3 IP QoS frameworks/models

Many factors influence the user-perceived quality of a telecommunication service, from *codecs* employed to the performance of the network. The constraints and requirements have been presented in the previous Section 3.2. In this Chapter we will analyze the mechanisms designed for IP networks to achieve QoS. This Section addresses the IP layer and as such we keep it very general, so that the satellite network can be one of the possible scenarios.

It is well known that IP networks were not designed to provide any

Service class	Conversational (delay \ll 1 s)	Interactive (delay \sim 1 s)	Streaming (delay < 10 s)	Background (delay > 10 s)
Error tolerant	Conversational voice and video	Voice messaging	Streaming audio and video	Fax
Error intolerant	Telnet interactive games	e-commerce Web browsing	FTP, still image, paging	e-mail arrival notification

Table 3.4: Application examples in terms of QoS.

This table is reproduced from “Radio Resource Management across Multiple Protocol Layers in Satellite Networks: A Tutorial Overview”, P. Barsocchi, N. Celandroni, F. Davoli, E. Ferro, G. Giambene, F. Castaño, A. Gotta, J. I. Moreno, P. Todorova, *International Journal of Satellite Communications and Networking*, Vol. 23, No. 5, pp. 265–305, September/October 2005. ISSN: 15442-0973. ©2005. Copyright John Wiley & Sons Limited. Reproduced with permission.

QoS guarantees. However, the applications traditionally using IP as a communication technology could perfectly cope with that lack; telephony or iterative applications over IP (that are nowadays beginning to be used) need transport networks with very strict QoS support. Such mechanisms vary from 100% guarantee solutions (employing techniques that can be assimilated to virtual circuit creation/provisioning) to other solutions not providing 100% guarantees. The over-provisioning approach is also considered but, of course, it cannot be applied in scarce-bandwidth radio access networks. Besides, offering different qualities for the data transport service will create new opportunities for providing several quality levels at different prices. We can conclude that, in the future, the IP data transport will be QoS-enabled.

The way to provide QoS in IP networks has been discussed for a long time. The most accepted solutions are IETF’s IntServ [9] and DiffServ [10]: both IntServ and DiffServ endow the routers with QoS mechanisms, such as queuing, scheduling and shaping, as illustrated in Figure 3.2. These mechanisms are implemented in the router interfaces. The main difference between IntServ and DiffServ lies in the level of detail used by the classifiers and in the need to keep state information.

The IntServ model provides end-to-end QoS guarantees by reserving per-flow resources (normally using the RSVP protocol [11]) in all the nodes along the path. While this architecture provides excellent QoS guarantees, it has scalability problems in the network core because of per-flow state maintenance and per-flow operation in routers. It is worth noting that RSVP is not the only IP reservation protocol, but RSVP is by far the most accepted one and has become an “integral” part of IntServ networks. There exist even some commercial RSVP-enabled routers.

RSVP identifies a communication session by the combination of destination address, transport-layer protocol type, and destination port number. In IPv6 those two last parameters may be replaced by the flow label. RSVP is used to reserve resources in the routers along the path between the sender and the receiver(s). RSVP also allows freeing these resources when they are no

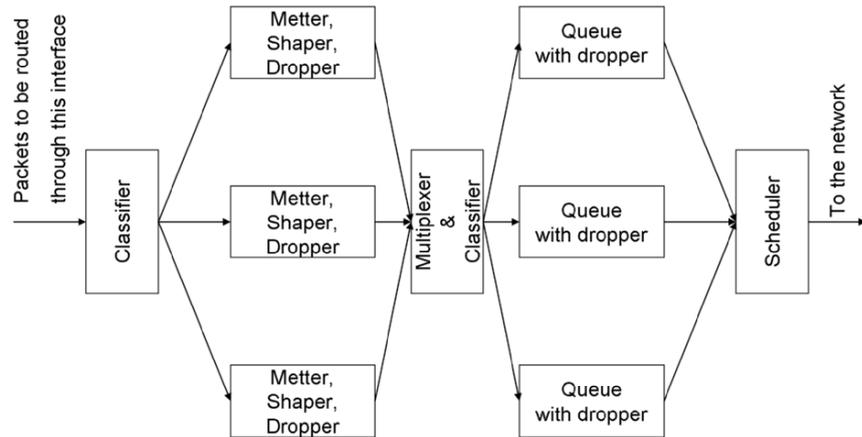


Fig. 3.2: QoS mechanisms in a router interface.

longer needed. Normally these reservations are to be policed and it is common to have an entity termed bandwidth broker (or, also, QoS broker) that takes the policy decision and communicates it to the routers. This entity will be studied later in this Section.

The primary messages used by RSVP are the “Path” message, which originates from the traffic sender, and the “Reservation” message, which originates from the traffic receiver(s). They are used in the resource reservation process. RSVP can also explicitly shut down the QoS sessions using RSVP teardown messages. Teardown messages can be initiated by an application in an end-system (sender or receiver) or a router as the result of state timeout. RSVP supports two types of teardown messages: “path-teardown” and “reservation-request teardown”. Path-teardown messages delete the path state (deleting the reservation state), travel toward all receivers downstream from the point of initiation, and are routed like path messages. Reservation-request teardown messages delete the reservation state, travel towards all matching senders upstream from the point of teardown initiation, and are routed like corresponding reservation-request messages.

On the other hand, DiffServ requires no per-flow control in the core network and, consequently, routers do not maintain any per-flow state and operation; no reservation protocol exists. As a result, DiffServ is relatively scalable in the forwarding/data plane, but offers no strict QoS guarantees. The criterion to classify the packets in core routers relies on the *DiffServ Code Point* (DSCP) field in the packet header [12]. DSCP defines three classes of priority:

- *Best Effort* (BE): to provide the service in the same way as in the current Internet, where there are no QoS guarantees, IETF recommends that the DSCP value should be 000000 (bin).
- *Assured Forwarding* (AF): The AF group contains four independent classes, each with three different *drop precedence* values in the queues. There is no specified algorithm for each value, but the dropping probabilities must be increasing and the packets must be marked with AF DSCP value and must arrive to the destination in the proper order. In case of congestion, the dropping probability depends on the *drop precedence* value.
- *Expedited Forwarding* (EF): EF is designed as the best group. It should provide very small drop probability, latency and jitter. That is the reason why this service is sometimes regarded as a *Virtual Leased Line* (VLL). This *Per-Hop Behavior* (PHB) is predestined to handle real-time applications like video streams. When EF packets enter a DiffServ router, they should be handled in very short queues and quickly serviced to maintain lower latency, packet loss, and jitter. Throughput of the EF flow should be limited to the value that can be handled by each node. It is necessary to avoid the situation where the queue could overflow and cause flow degradation. IETF recommends that the EF class should be marked with the DSCP value 101110 (bin).

Routers should allocate enough resources for the high priority DSCPs, while the lower ones or the “classical” BE traffic (DSCP 0) may use spare resources. DiffServ networks require access control in the edge routers, so that only authorized users can inject packets with high priority DSCPs. Access control is enforced by the shapers. Depending on the type of edge routers, this access control can take place in different levels of detail. For instance, in edge routers connecting the core network to the users (*Access Routers*, ARs) this control follows a per-user and per-flow basis, since ARs will handle a small traffic load. However, for edge routers connecting the core network to the Internet or other domains, this access control can only proceed at a very rough level of detail.

Besides the QoS-enabled routers, another entity called QoS broker [13] is used to control and to manage the network. This entity, for scalability reasons, can be replicated in the network; moreover, the network can be hierarchically divided into several areas, as proposed in [14]. In a simplified way, the QoS broker manages and monitors the network resources in one particular domain of operation. It also monitors the edges for incoming and outgoing resource reservations/utilization. The information thereby acquired is used in conjunction with the policy system information to take admission control decisions and reconfigurations and to convey them to the routers. A QoS broker is then an entity that takes *Service Admission Control* decisions and performs network device configuration, according to a set of conditions imposed by the network administration entities (e.g., *Authentication, Authorization and Accounting*, AAA, System) with the goal of achieving end-to-end

QoS, also over different networks. The QoS broker may also be responsible for managing inter-domain communications with neighbor QoS brokers, so that QoS-enabled transport services are implemented in a coordinated way across various domains.

Since IntServ requires resource reservation, it is the most evident scenario to integrate a QoS broker. In IntServ a RSVP enabled router may consult the QoS broker (using the *Common Open Policy Service*, COPS, protocol) about the decision to take when receiving RSVP path or reservation messages. The decision taken by the QoS broker is normally conveyed in a COPS message and then enforced by the router. In DiffServ, the edge routers need to perform admission control and may also outsource the decision to the QoS broker. This process can take place when the DiffServ access router detects a new traffic; the level of detail to define new traffic may vary, as we just explained. QoS brokers functionally can go beyond taking policing decisions; generally they are also in charge of managing the network. The actual role of the QoS broker may adapt to the different scenarios and business models. For instance in the scenario described in Section 3.4.1, the “recovery provider” may consult a QoS broker before gathering data from the content providers and sending it to the satellite so that this broadcasts it.

Many existing approaches combine IntServ and DiffServ: IntServ in the access part of the network and DiffServ in the core network. Of course, solutions based on other paradigms also exist and are even complementary to these ones. For example, [15] proposes new routing schemes over high availability networks.

3.4 Broadcast and multicast services

In addition to DVB-S broadcast, satellite IP multicast for content distribution to the “*edge*” of the Internet and to corporate sites has numerous advantages over terrestrial technology. Satellites offer highly “*regular*” broadband data streams and a single transmission from a central operation center can be delivered to a high number of receiving sites. In addition to reducing costs, the single long hop of the satellite link replaces all the small hops of terrestrial content distribution and bypasses bottlenecks, thus improving QoS in many applications. Thus, satellite multicast for content distribution and satellite content delivery to mobile terminals (either broadcast or multicast) are interesting working areas. Clearly, reception is mainly possible when the satellite is in direct line-of-sight or attenuation is low. Hence, complementary terrestrial repeaters enhance the architecture by retransmitting the satellite signal.

When only a satellite signal is present (i.e., no terrestrial repeaters), satellite broadcasting systems may use time diversity to enhance availability. This technique broadcasts the same content twice, so that the two transmissions are uncorrelated with respect to mobile reception blockages. The receiver is

able to combine them to provide seamless reception.

In the case of satellite broadcasting to mobile terminals using mobile communication modulations, the *client* could switch between two content sources with different QoS levels: satellite (or terrestrial-repeated satellite) and terrestrial wireless networks (when neither satellite nor terrestrial repeaters are available). This handover between physically different access interfaces is problematic for example in the case of UMTS and WiFi (again, the latter would provide a higher QoS level, at least in terms of regularity, if a satellite gateway is present).

When terminals support dual network access, e.g., satellite and terrestrial (WiFi, UMTS, etc.) links, it is quite critical to select the appropriate network for each application depending on both the available resources and the kind of application involved. In general, interface selection can be network-initiated or terminal-initiated. In the first case, the network operator decides the appropriate access network for each application, whereas in the second case the terminal will decide the best path. All these procedures must be performed during application initialization as well as during handovers, and must consider available access technologies, user profile (SLA, user requirements, etc.), and QoS capabilities depending on the available resources. In the case of multicast and broadcast services, terminal-initiated interface selection seems the natural approach, since it would be too difficult for a network operator to select *individually* optimum interfaces for the large user populations involved.

Satellites have traditionally served point-to-point communications (such as intercontinental telephony circuits) and unidirectional TV broadcast. *Very Small Aperture Terminals*, VSATs (i.e., narrowband data terminals in transactional mode), appeared in the 90's. With some exceptions, the medium access technology at that time neither allowed broadband service provision nor massive terminal deployment, but 10-to-100 units at most. On the other hand, equipment manufacturers developed proprietary platforms that could not interoperate. A high terminal/service cost kept related services within corporate markets, beyond the possibilities of SMEs. This situation has radically changed in the last six years, due to technological advances such as multiple access protocols. On one hand, VSAT terminal manufacturers (Hughes, Gilat [16], etc.) have developed fully bidirectional equipment (still proprietary) to provide broadband services to large user communities and, in some cases (Starband [17], DirectWay), at an acceptable cost even for residential users. On the other hand, a bit of new manufactures offer interoperable equipment based on the DVB standard, i.e., specifically, MPE (*Multi Protocol Encapsulation*) and RCS (*Return Channel via Satellite*).

The advent in 1997 of the MPE standard for DVB IP data encapsulation implied that equipment manufacturers should no longer supply the whole communication chain thanks to interoperability. Traditional head-end manufacturers began to include IP data insertion equipment in their catalogues (Thomcast [18], Divicom, Rohde & Schwarz, etc.), and some new ones completely centered their efforts in this direction (Logic Innovations,

Tandberg [19], etc.). In general, they did not provide user terminals, due to the deep differences between professional and user markets in terms of quality goals, sales support, etc. For this reason, many PC peripheral manufacturers entered the competition with DVB-S boards and boxes (Adapteq, Terratec [20], Technotrend, etc.).

MPE stimulated the entrance of satellite IP services into the mass market. For applications requiring interactivity (bidirectionality), the services initially relied on auxiliary terrestrial technologies for the return channel, wired (POTS, ISDN or Frame Relay) or wireless ones (GSM, GPRS or similar). There was a clear lack of a satellite technology to eliminate this terrestrial dependence. In 1999, the DVB-RCS standard covered this gap. Despite of some initial interoperation problems (usually leading to the election of the same supplier for the whole communications chain), the standard has matured in the last years. Several operators have selected it (Satlynx, Hispasat [21], etc.).

In the last two years the new protocol *DOCSIS for Satellite* (or DOCSIS-S) is emerging as an alternative to DVB-RCS, based on the well-known DOCSIS standard for cable networks and mostly promoted by American vendors and providers (Viasat [22] and WildBlue [23]). Compared with DVB-RCS, DOCSIS-S exploits the economies of scale of silicon designs for cable infrastructure, and takes advantage of a huge selection of *Operations and Business Support Systems* platforms from the cable market. However, DOCSIS-S is still a “vendor-promoted protocol”, not a real standard; thus interoperability and availability of suppliers are an issue.

These new protocols enable new multimedia application scenarios based on multicast and broadcast distribution. One of these applications is distance learning with or without interactivity. In it, a teacher provides a lesson to a number of remote students using multicast video and audio streaming and additional aids such as a digital blackboard, slides, etc. When interactivity (return channel) is available, students may send questions to the teacher either by chat or by their own microphone and webcam, so that the other students may follow the question and the response. In this case, because of the delay induced by the satellite itself (500 ms for a GEO system), the media access protocol for the return channel (100 - 300 ms) and the video codecs (100 - 1000 ms), a voice handshake similar to a “walkie-talkie” must be implemented in order for the teacher and the student not to interfere. Also, when there is a large audience, the application must provide specific controls so that the teacher may act as moderator, granting or denying participation to the students. At present, distance learning systems (Centra [24]) and services (Hughes [25], Gilat [16]) are commercially available and widely deployed.

Another common multicast application not requiring real-time operation that largely benefits from a return channel when available is massive content distribution, where a central station delivers common multimedia contents to a large population of remote clients (with a reception acknowledge mechanism when interactivity is provided). The typical data losses and corruptions are

avoided by *a*) adding redundant information to the data to be transmitted at the application level by means of convolutional coding, polynomial protection and interleaving, and *b*) implementing a return channel so that each remote client may inform the central station about the missing parts of the media content after receiving them and correcting the errors. Then, the central station re-sends those pieces of data grouped in overlapped parts, to avoid repeating the same datagrams. Massive content distribution software solutions are available from Kencast [26] and Tandberg [19], among others.

The DVB-RCS standard enables other innovative application scenarios for satellite content delivery to mobile terminals, such as *Delayed Real-Time* (DRT) services with QoS support for GEO satellite distribution systems. We describe them in the next sub-Section.

3.4.1 Delayed real-time service over GEO satellite distribution systems

The distribution of multimedia contents via satellite, even though it is one of the very first services envisioned by the satellite communication community, still represents a hot topic for satellite networks. There are many types of multimedia communication services; in this sub-Section, we address the class of DRT services, whose importance arises in the field of QoS-aware real-time communications.

DRT services fall in the category of streaming services whose requirements are discussed in sub-Section 3.2.3. DRT services have been conceived as an extension of unidirectional real-time broadcast and multicast services. So far, there are no standard architectures to support DRT, but diverse applications have been proposed in order to cope with given QoS requirements by means of specific application layer mechanisms. Instead of limiting DRT support to a mere application layer implementation, this Section presents an architecture that exploits both application and transport layer features. The proposed architecture assumes that DVB-RCS is deployed over a GEO satellite system. Nonetheless, it can be easily extended and adapted to any other layer 2 protocol stack suitable for broadcast and multicast applications, allowing customers to interact in real-time with the multimedia distribution farm (e.g., WiMAX or UMTS technologies).

A DRT service recovers from data losses and corruptions by using a buffer and, in turn, by introducing an artificial delay at the beginning of the play-out phase. A real-time return channel is fundamental, since the receiver must initiate a data recovery procedure after a data loss has been detected. In that case, additional resources can be invoked over the distribution channel, if available. The maximum possible duration for each recovery phase is determined by the length of the adopted buffer, and can be modulated by the choice of the codec (or codecs) for the multimedia streaming.

It is worth noting that multiple retransmissions could be requested at the same time by different users (e.g., by users belonging to the same multicast

group), and, therefore, different retransmissions could partially overlap. Accordingly, retransmissions are executed in multicast and are requested through dynamically joining and pruning the multicast retransmission group. As a consequence, it is possible to design an architecture where a legacy satellite broadcast service is endowed with a specific multicast recovery algorithm able to mitigate the impact of network/satellite disruptions. This is the case of link failures due to user mobility and related shadowing effects. The reference scenario (Figure 3.3) is composed of three main elements:

- The *Content Provider* (we assume to have N content providers in the network);
- The *Recovery Service Provider* (just one in the network);
- The *users* (specifically, N groups of users, one group for each active content provider).

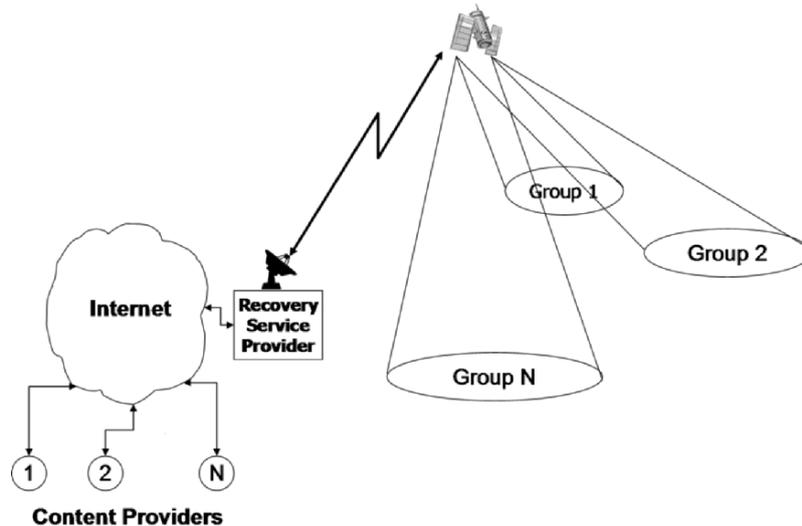


Fig. 3.3: DRT service architecture.

The *Content Providers* are the primary sources for video applications, i.e., they generate the real-time data. We can suppose that a content provider is located just before the satellite hop or, more generally, that the Internet spreads between them.

The *Recovery Service Provider* consists of a streaming proxy that has access to satellite resources and manages the retransmission priority. In fact, retransmission requests can be listed according to a priority that is related to the time constraints of the recovery phase, but also to the type of service and the customer class the service pertains to. It is worth noting that

retransmission requests can be rearranged in time by the proxy, based on a metric that quantifies the importance of a data segment for a requesting customer, so that a simple FIFO scheduling of retransmissions is far to be optimal in terms of fairness, throughput and user satisfaction degree.

The user is actually to be considered as a set of customers (*Group 1*, *Group 2*, ... , *Group N*) located behind the satellite link, whose applications share some common bandwidth resources. Optimizing the usage of those resources is one of the goals of the envisaged architecture.

3.4.2 Scenario characterization and results

Every content provider sends a multimedia stream over the satellite link using a guaranteed bandwidth. According to Figure 3.3, there are N content providers and, therefore, N statically allocated channels. Data are sent to the streaming application after a playout delay (e.g., D seconds). Each receiver uses a local proxy buffer to store at most D seconds of streaming data, i.e., data to be played within D seconds. This “*elastic buffer*”, that empties at constant rate and fills at variable rate, permits to continue the playout during the satellite channel outage, as long as sufficient information has been previously stored in the buffer. When a channel outage happens, the receiver (i.e., the proxy located at the receiver group) leaves a blank space in the application buffer and, when the channel is again available, sends a retransmission request to a *Recovery Service Provider* (RSP), in order to fill the hole in the elastic buffer. All the retransmissions use a shared channel, e.g., the $(N+1)$ -th channel. We propose that, in this “*recovery*” channel, content providers retransmit the packets using a transport protocol with the *Additive Increase Multiplicative Decrease* (AIMD) scheme [27],[28]. In particular, the number of packets a sender can put on the network is limited by a *congestion window* ($cwnd$) that is managed as follows:

- Slowly (additively) increase the $cwnd$ size as long as there is no congestion. Typically, the $cwnd$ is increased by one packet for each window sent without a packet drop (in practice, $cwnd = cwnd + \alpha/cwnd$ as each ACK returns, with $\alpha = 1$).
- Quickly (multiplicatively) decrease the $cwnd$ size as soon as congestion is detected. Typically, $cwnd$ is halved for each window containing a packet loss ($cwnd = \beta/cwnd$, with $\beta = 0.5$).

In this way, the available bandwidth is fairly shared. After receiving a retransmission request, the RSP (which acts like a proxy for on-demand services) classifies the request according to the run-time estimated urgency. The urgency is calculated from the information requested and the time available for recovery purposes. Correspondingly, every user communicates a time interval and two timestamps conveyed by the retransmission request:

$$\Delta T, [t_0, t_1] \quad (3.1)$$

where t_0 is the time when the broadcast connection became unavailable for the requesting receiver, t_1 is the time when the data to be retransmitted should be used by the multimedia player, and ΔT is the data window that is requested, i.e., the “room” to be filled in the receiver buffer, in playout seconds.

The RSP assigns a proper bandwidth to each retransmission, which is calculated from the corresponding urgency. The policy that determines the urgency of a request is based on both the difference ($t_1 - t_{current}$) and ΔT , i.e., the intervals available to start and to complete the recovery procedure. This means that the urgency of a retransmission may change during the retransmission itself, so that bandwidth assignments have to be dynamically adjusted. Possibly, a policy function might run on the retransmissions codec, trying to accommodate multiple requests in the same channel.

Once the codec has been selected for a retransmission, the amount B of data to be sent is determined, and the following formula is used to compute the AIMD transmission parameters α and β :

$$B = r(\alpha, \beta) * (t_1 - t_{current}) \quad (3.2)$$

where B is the amount of data to send at time t_1 and r is the rate to be achieved by means of an opportune choice of α and β .

A formula is shown in [29] that correlates the AIMD mean sending rate r with the control parameters, α and β , the loss rate p , the mean *Round Trip Time*, RTT, the mean timeout value, T_0 , and the number b of packets each ACK acknowledges:

$$r(\alpha, \beta) = \frac{1}{TD_{\alpha, \beta} + TO_{\alpha, \beta}} \quad (3.3)$$

where:

$$TD_{\alpha, \beta} = RTT \sqrt{\frac{2b(1-\beta)}{\alpha(1+\beta)}} p \quad (3.4)$$

$$TO_{\alpha, \beta} = T_0 \min \left(1, 3 \sqrt{\frac{(1-\beta^2)b}{2\alpha}} p \right) p(1+32p^2) \quad (3.5)$$

Thus, from the bandwidth value, the proxy calculates α and β parameters of the AIMD transport scheme, which will be communicated to every content provider that has to retransmit data.

Here we modeled the link with a good-bad process with exponentially distributed permanence times for both good and bad states. Real-time broadcast applications are always on, with a fixed bandwidth usage. Also the bandwidth available for retransmission is fixed and guaranteed by the distribution systems, and the playout delay of each receiving application is the same for all users. Furthermore, we represent each multicast group with a single user that acts as the worst-case user, so that the good-bad process actually refers to the time distribution of periods in which link failure occurs or not, for an entire multicast group. This assumption simplifies the simulative analysis while preserving the correctness of results; in fact, in our system, overlapping retransmission requests sum and turn into a single multicast retransmission. Finally, no codec adaptation has been considered.

As for the transport protocol, we have tested UDP-like retransmissions (the evaluation of TCP and AIMD-like protocols will be considered in a future study). However, preliminary results obtained with UDP, justify the study of connected transport protocols to enhance system performance.

As a reference, let us consider a scenario with $N = 10$ *Content Providers* generating an aggregate of 20 Mbit/s (each *Content Provider* generates at a fixed, but different rate of about 2 Mbit/s, to avoid synchronization effects), and a 6 Mbit/s bandwidth is guaranteed for recovery. The playout delay of users is 20 seconds, and the transport protocol is UDP. The average duration of the bad state of each link has been set to 5 s; we have obtained the results by changing the average duration of the good state and by collecting simulation results over 200000 seconds.

Figure 3.4 shows the aggregate amount of retransmitted data when the adopted retransmission priority is proportional to the bandwidth of the real-time stream. Curves are normalized to the aggregate number of bytes requested by users. The lower curve in the Figure represents data retransmitted for retransmissions that the system was able to complete. It is clear that a great number of retransmissions is stopped due to lack of resources as soon as the link error probability exceeds 0.2. Furthermore, for error probability greater than 0.1, the number of unrecoverable bytes increases (due to outage periods longer than the playout delay, which are now more frequent).

For the same scenario, Figure 3.5 depicts the aggregate delivered data and the amount of data lost due to link failures during the retransmission procedure. Lost data are normalized to retransmitted data and not to requested data, to give a correct measure of the needs of a connected transport protocol during the recovery procedure. Note that system performance is not satisfactory even with values of the link failure probability as small as 0.1, which is not so much for users.

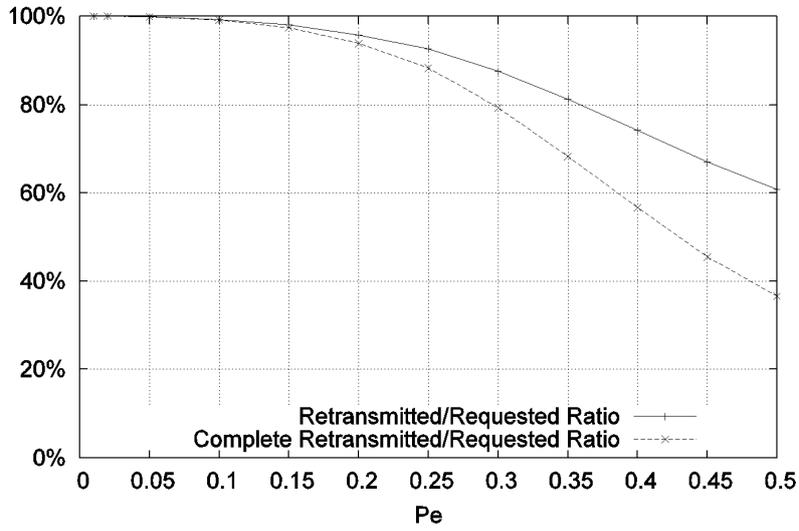


Fig. 3.4: Retransmitted data using a retransmission priority proportional to the required bandwidth.

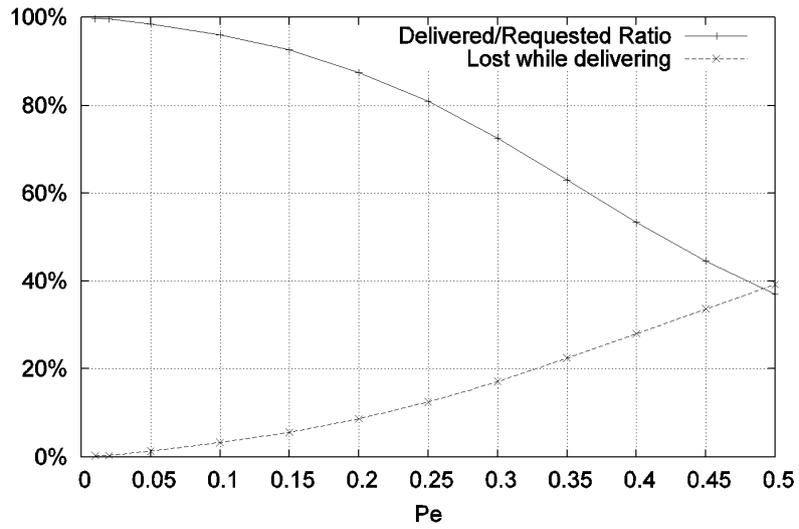


Fig. 3.5: Delivered and lost retransmitted data using a retransmission priority proportional to the required bandwidth.

3.5 Experimental results on QoS

Many of the application QoS requirement studies have been done in current-day Internet networks, for instance many of the considerations shown in Section 3.2. The aim of this Section is to describe the work carried out in a *Next-Generation Network* (NGN) prototype to characterize the application QoS requirements in such a kind of network. Results refer to real experiments on application behavior.

The test bed was an “*all IPv6*” network; Figure 3.6 depicts the network architecture. Two access technologies, one wired (Ethernet) and one wireless (IEEE 802.11), were employed. This can represent a subset of all the access technologies a future network operator may offer to its customers. Users, employing the appropriate devices could connect to any of the two networks. In the test bed, wireless connectivity is assured using commercial “SMC WLAN” cards with prism driver. Satellite links were not available in our test bed due to the complexity and high costs in using these links for experiments. We however believe that the obtained results may provide good insights also for general networks (including satellite links) in particular for what concerns the characterization of application behavior in NGNs with features such as mobility or QoS, using IP as convergence layer.

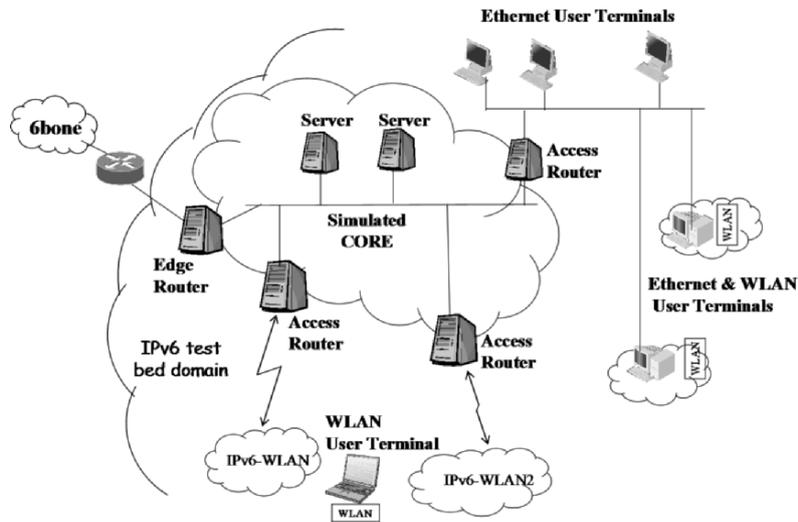


Fig. 3.6: NGN prototype test bed.

Our network was divided in 2 parts: (i) an “*access part*” where the users connect to via either Ethernet or WLAN (i.e., WiFi); (ii) the core network. The latter is connected to the “*6 bone*” (IPv6 Internet) via an *Edge Router*

and to the “*access part*” and the users’ terminals via the *Access Routers*. The core network hosts two servers supporting different functionalities of NGNs. These functionalities include aspects that should be present in next-generation commercial mobile networks, such as user authentication and accounting; mobility and QoS management were also controlled by these servers. All the nodes (including the routers) are general-purpose machines (Pentium III and IV PCs). All run Red Hat 7.2 with Linux-2.4.16 kernels. More details about the test bed can be found in [30].

QoS is based on DiffServ with access control. This access control is performed on the *Access Routers* on a per flow and per user basis. The *Access Router* outsources the admission decision to the QoS broker, an entity located in the core network able to take this decision and configure the routers with appropriate parameters.

The test bed here described is composed of general-purposes machines and it is just a mere representation of what a next network infrastructure may be, but we believe that the results obtained in it can provide us early and valuable hints about the applications specific QoS requirements when using NGNs.

We performed on-site real measurements of end-user performance perception and application characterization under different situations that can be present in NGNs, as detailed in [30] and [31].

The tested applications correspond to conversational services and interactive services. All of them were IPv6 applications. Conversational services were provided by *Robust Audio Tool* (RAT) for conversational voice and Quake 2 and Tetrisnet for games. Again, for interactive services we employed RAT (for audio streaming) and VideoLan for video streaming. Conversational and interactive services characterization was already described in Section 3.2; the added value of this Section is to show experimental studies obtained in an NGN prototype and check the differences.

Two kinds of tests were performed: the first was intended to characterize application behavior in terms of bandwidth needs (including burstiness); the second one experimented with user tolerance to delay, jitter and packet loss. We will show and analyze the results; the tests methodology is further detailed in [30].

For the first type of tests, *ethereal* [32], a network analyzer software, was used to capture the packets and *tcpstat* was adopted to analyze the application traffic. Two parameters were evaluated: packet size and packets per second. Mean, min, max, deviation and deviation/mean values were calculated for those two parameters. First, the results are presented and then some conclusions drawn. Audio stream has constant packet size and very small variation in packet rate. For video stream we have a nearly constant packet size and a small variation in packet rate. For conversational applications the results are as follows:

- Conversational voice presents a constant packet size, but also a high variation in packet rate.

- The Tetrisnet game generated a very low traffic, but with great variation in packet size and rate.
- Quake 2 generated more traffic and also had remarkable variations in packet size and a small variation in packet rate.

As a general conclusion, interactive applications have a higher bandwidth variation since they depend on user behavior: there is silence suppression, thus when the user does not talk no packets are sent. Moreover, Quake 2 bandwidth consumption depends on user activity: the more it interacts the larger the packets are, because more information needs to be sent (packets are sent at a rather constant rate). The bandwidth of the streaming application does not depend on user behavior, but only on the nature of scenes and audio. Obviously, the employed codecs play a fundamental role in determining application bandwidth consumption.

The results are as expected and similar to the ones obtained in the current Internet. However, there are some remarkable aspects worth to mention. For instance, mobility and overhead. Mobility in NGNs will be based on *Mobile IP* (MIPv6). This means adding, to the basic IP header the IP home address header and, also, generally the IPv6 routing header. For conversational applications with only audio, the payload is small and, as such, the ratio payload/overhead becomes very small. We also found NGNs specific results when dealing with applications adaptability. In NGNs, the users will roam between several access technologies with different performance characteristics. Applications should be able to cope with this heterogeneity adapting themselves, for instance in “layered” video, sending only detailed layers when the available bandwidth is high, for instance in downlink satellite links.

As aforementioned, the second type of tests evaluated user-perceived quality. NIST Net [33] is the software that can alter network conditions. It was employed to generate packet loss, delay and jitter in the test-bed network. Since NIST Net works only on IPv4 networks and the test-bed infrastructure was pure IPv6, a tunnel was set up. Table 3.5 presents the results. These results were as expected: conversational applications (Tetrisnet, Quake 2, and VoIP) have more strict requisites for delay and jitter. Tetrisnet is an exception, since it is an interactive application, but interaction speed is rather small (in the order of a second) so that delay requirements are very loose.

Application	Packet loss (%)	Delay/Direction (ms)	Jitter/Direction (ms)
Audio Stream	2	> 500	100
Quake 2	15	100	150
VoIP	10	150	50
Tetrisnet	20	> 500	> 500

Table 3.5: QoS requirements as measured in the NGN prototype.

The obtained requirements are similar to those presented in Section 3.2 for nowadays networks. The specific aspects of NGNs can be found mainly in the fact that network QoS is priced and tailored for the users. As such, we found that low profile users, “paying” less for the transport service where much more tolerant with their requirements. Besides, for some users, more than having better QoS, the important aspect was the unique NGN ability of supporting all kinds of applications and having seamless inter-technology handovers with the capability of taking the best profit from the available access technologies.

3.6 Conclusions

This Chapter stressed on the importance of providing QoS for data transport. Some applications have stringent QoS requirements, mainly related to delay and jitter. Satellite networks may suffer from too high delays so QoS aspects should be considered very carefully. On the other side, satellite networks are very well suited for multicast and broadcast transmissions as well as for DRT services. For about 6 years now, satellite networks are also a commercial solution for completely different scenarios: unicast bidirectional services like broadband Internet access. These scenarios, requiring strong QoS requirements, need a careful analysis and the implementation of mechanisms to support QoS as discussed in the next Chapters of this book.

References

- [1] ITU-T Recommendation G.1010: "End-user multimedia QoS categories", URL: <http://www.itu-t.org>.
- [2] ITU-T Recommendation Y.1541: "Network performance objectives for IP-based services", URL: <http://www.itu-t.org>.
- [3] ITU-T Recommendation F.700: "Framework Recommendation for audio-visual/multimedia services", URL: <http://www.itu-t.org>.
- [4] 3GPP, "Technical Specification Group Services and System Aspects Service aspects; Services and Service Capabilities", TS 22.105 V6.0.0 (2002-09) (Release 6), URL: <http://www.3gpp.org>.
- [5] ETSI, "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia (BSM) services and architectures; Functional architecture for IP interworking with BSM networks", TS 102 292, V1.1.1 (2004-02).
- [6] 3GPP, "QoS Concept and Architecture", TS 23.107, URL: <http://www.3gpp.org>.
- [7] ITU-T Recommendation G.114: "One-way transmission time", URL: <http://www.itu-t.org>.
- [8] P. Barsocchi, N. Celandroni, F. Davoli, E. Ferro, G. Giambene, F. Castaño, A. Gotta, J. I. Moreno, P. Todorova, "Radio Resource Management across Multiple Protocol Layers in Satellite Networks: A Tutorial Overview", *International Journal of Satellite Communications and Networking*, Vol. 23, No. 5, pp. 265-305, September/October 2005. ISSN: 15442-0973.
- [9] R. Braden *et al.*, "Integrated Services in the Internet Architecture: an Overview", IETF RFC 1633, June 1994.
- [10] S. Blake *et al.*, "An Architecture for Differentiated Services", IETF RFC 2475, December 1998.
- [11] R. Braden *et al.*, "Resource ReSerVation Protocol (RSVP) - Version 1 Functional Specification", IETF RFC 2205, September 1997.
- [12] K. Nichols *et al.*, "A Two-Bit Differentiated Services Architecture for the Internet", IETF RFC 2638, July 1999.
- [13] K. Nahrstedt *et al.*, "The QoS Broker", *IEEE Multimedia*, Vol. 2, No. 1, pp. 53-67, Spring 1995.
- [14] G. Cortese *et al.*, "CADENUS: Creation and Deployment of End-User Services in Premium IP Networks", *IEEE Communication Magazine*, Vol. 41, No. 1, pp. 54-60, January 2003.

- [15] G. Schollmeier *et al.*, “Providing Sustainable QoS in Next-Generation Networks”, *IEEE Communications Magazine*, Vol. 42, No. 6, pp. 102-107, June 2004.
- [16] Gilat Satellite Networks, URL: www.gilat.com.
- [17] StarBand, URL: <http://www.starband.com/>.
- [18] Thales Broadcast and Multimedia, URL: <http://www.thomcast.com/>.
- [19] Tandberg, URL: <http://www.tandbergtv.com>.
- [20] TerraTec Electronic GmbH, URL: <http://www.terratec.net/>.
- [21] Hispasat, URL: <http://www.hispasat.com>.
- [22] Viasat, URL: <http://www.viasat.com>.
- [23] Wildblue, URL: <http://www.wildblue.com>.
- [24] Centra, URL: <http://www.saba.com/centra-saba/>.
- [25] Hughes, URL: <http://www.hughes.com>.
- [26] Kencast, URL: <http://www.kencast.com>.
- [27] E. Altman, C. Barakat, V. Manuel Ramos, “Analysis of AIMD Protocols over Paths with Variable Delay”, *INFOCOM 2004*.
- [28] L. Cai, X. Shen, J. W. Mark, J. Pan, “A QoS-Aware AIMD Protocol for Time-Sensitive Applications in Wireless/Wired Networks”, in *Proc. of IEEE Infocom’05*, Miami, Florida, March 13-17, 2005.
- [29] Y. R. Yang, S. S. Lam, “General AIMD Congestion Control”, University of Texas, *Tech. Rep.* TR-2000-09, May 2000.
- [30] P. Serrano *et al.*, “Medida y análisis del tráfico multimedia en redes móviles de cuarta generación”, *Telecom*, I+D 2004, Madrid.
- [31] A. Cuevas *et al.*, “Usability and Evaluation of a Deployed 4G Network Prototype”, *Journal of Communications and Networks* (ISSN: 1229-2370), Vol. 7, No. 2, pp. 222-230, June 2005.
- [32] Ethereal: A Network Protocol Analyzer, URL: <http://www.ethereal.com/>.
- [33] NIST Net Home Page, URL: <http://snad.ncsl.nist.gov/itg/nistnet/>.