QUALITY OF SERVICE IN INFORMATION NETWORKS

Augusto Casaca
IST/INESC, R. Alves Redol, 1000-029, Lisboa, Portugal.

Abstract: This article introduces the problems concerned with the provision of end-to-end quality of service in IP networks, which are the basis of information networks, describes the existing solutions for that provision and presents some of the current research items on the subject.

Key words: Information networks, IP networks, Integrated Services, Differentiated Services, Multiprotocol Label Switching, UMTS.

1. QUALITY OF SERVICE IN IP NETWORKS

Information networks transport, in an integrated way, different types of traffic, from classical data traffic, which has flexible Quality of Service (QoS) requirements, to real-time interactive traffic, which requires QoS guarantees from the network.

Most of the solutions for the transport of information in this type of networks assume that the networks run the Internet Protocol (IP), which provides a best-effort service. The best-effort service does not provide any guarantees on the end-to-end values of the QoS parameters, i.e. delay, jitter and packet loss. However, the best-effort concept results into a simple network structure and, therefore, not expensive.

The best-effort service is adequate for the transport of classical bursty data traffic, whose main objective is to guarantee that all the packets, sooner or later, reach the destination without errors. This is achieved by running the Transmission Control Protocol (TCP) over IP. Services like e-mail and file

transfer are good examples of this case. The problem occurs when real-time interactive services, such as voice and video, run over IP. In this case, the achievement of an end-to-end delay and jitter smaller than a certain value is key to achieve a good QoS. This means that the best-effort paradigm needs to evolve within IP networks, so that new network models capable of efficiently transporting all the types of traffic can be deployed.

The end-to-end QoS in a network results from the concatenation of the distinct QoS values in each of the network domains. In reality, these QoS values depend on the QoS characteristics of the different routers and links, which form the network. The QoS is basically characterised by the transfer delay, jitter and probability of packet loss, all relative to the traffic traversing the network.

The end-to-end delay is caused by the store-and-forward mechanism in the routers and by the propagation delay in the links. Jitter, which is defined as the end-to-end delay variation for the distinct packets, is caused by the different time that each packet remains in the router buffers. Packet loss basically results from congestion in routers, which implies the discard of packets.

The evolution of the best-effort paradigm to improve the end-to-end QoS in an IP network can be achieved by doing resource allocation at the router level, by intervening in the routing mechanism and by traffic engineering in the network. All these actions can be performed simultaneously in a network or, alternatively, only some of them can be implemented, depending on the QoS objectives. In the following text we will analyse these different mechanisms.

The router structure in traditional best-effort networks, which is shown in figure 1, is very simple.

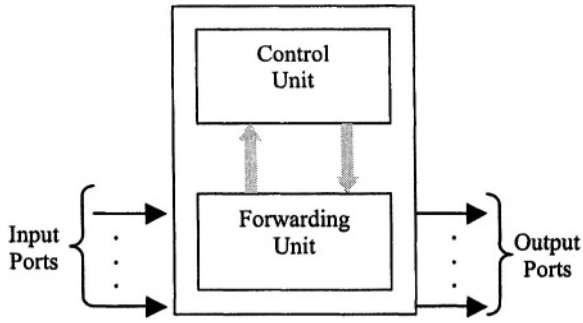

Figure 1. Best-effort router

The input ports accept packets coming from other routers and the output ports forward packets to other routers along the established routes. The forwarding unit sends each packet to the appropriate output port based on the IP destination address of the packet. For this purpose there is a routing table, which maps the destination address into the output port. The control unit is in charge of managing the forwarding unit. The routing protocol runs in the control unit.

To improve the QoS capabilities of the router, different mechanisms need to be implemented, which will result into a more complex structure for the router. These mechanisms are the following: classification, policing, marking, management of queues and scheduling [1].

Each traffic class, which requires bounded values for the end-to-end delay, jitter and packet loss, independent of the remaining traffic, needs a separate queue in the router. When a packet arrives at the router it needs to be classified and inserted into the respective queue. Also, after classifying a packet, it must be decided if there are enough resources in the queue to accept the packet. The policing mechanism is in charge of this action. A decision can also be taken in order to accept the packet conditionally, i.e. to mark the packet and discard it later in case of necessity. Each queue must have its own policy for packet discard depending on the characteristics of the traffic served by the queue. This is done by the queue management mechanism. Finally, a scheduling mechanism is required to decide on the frequency of insertion of packets into the output port that serves several queues.

Each of the referred mechanisms results into a new functional block in the router. QoS-capable routers are definitely more complex than best-effort routers, but must be able to inter-operate with them, because according to the Internet philosophy, incremental changes in one part of the network should be done without impact in the remaining parts of the network.

These QoS-capable routers are required for the new IP network models, namely Integrated Services (IntServ) and Differentiated Services (DiffServ), which need to allocate resources in the network routers for the distinct types of traffic classes. These network models will be explained later in this article.

The Internet routing is based on the shortest-path algorithm. Based on the IP address of the destination, this algorithm establishes a route between source and destination by using the shortest-path according to a well defined metric, for example, the number of routers to be traversed or the cost of the different routes. The algorithm is very simple, but it might cause an over-utilization of certain routes, leaving others free, when the network is highly loaded. This over-utilization results in extra delays and, in some cases, packet losses. An alternative is to use QoS-based routing, which originates

multiple routing trees, in which each tree uses different combinations of parameters as the metric. This allows having different routes for the same source-destination pair according to the characteristics of the traffic. For example, one route could have delay as the metric and other route could have cost. The first one would be more appropriate for interactive traffic and the second one for bursty data traffic.

Finally, traffic engineering allows the network operator to explicitly indicate the use of certain routes in the network, also with the aim of achieving route diversification for the different traffic classes. Although traffic engineering uses techniques, which are different from the ones employed by QoS-based routing, if used in a network, can achieve by itself some of the objectives of QoS-based routing.

2. RESOURCE ALLOCATION MECHANISMS IN ROUTERS

As seen in the previous chapter, QoS-capable routers require the implementation of a number of additional mechanisms besides the ones provided in best-effort routers, namely classification, policing, marking, management of queues and scheduling.

2.1 Classification of packets

The selection of the input queue where to insert a packet arriving to a router depends on the packet class. The classification of the packet is based on n bits existing in the packet header. These n bits constitute the classification key and, therefore, up to 2^n classes can be defined.

Some complex classification schemes can consider several fields in the packet header to perform the classification, e.g. source address, destination address and TCP/UDP ports. However, the normal case only considers a single field in the header. In IP version 4 (IPv4) it is the TOS byte [2], in IP version 6 (IPv6) it is the TC byte [3]. To further simplify the classification scheme the semantics adopted for both versions of IP follows the one defined for the IP Differentiated Services (DiffServ) model [4]. This is one of the new models for IP networks having in view an improvement of the best-effort model as it will be studied in chapter 4. In the DiffServ model, the field equivalent to the TOS (IPv4) and TC (IPv6) is called the DiffServ field. It is one byte long and its structure is indicated in figure 2.

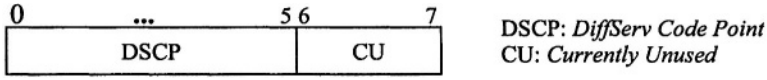

Figure 2. The DiffServ field

The 6 bits of the DSCP permit to define up to 64 different classes.

2.2 Policing and marking

Every class puts some limits on the timing characteristics of packet arrival. This consists on limiting the maximum allowed arrival rate and the maximum number of packets that can arrive within a certain time interval. The router polices the arrival of packets and can do one of two actions for the packets that do not respect the timing limits (out-of-profile packets), either eliminates all the out-of-profile packets, or marks them and lets them go into one of the router queues. The marking of packets allows that, in case of being necessary to drop packets in the queue, the marked ones might be selected to be the first ones to be discarded. The marking indication is given by a bit in the packet header.

The action of policing requires that the router is able to measure the timing characteristics of packet arrival so that it can decide whether the packets are in-profile or out-of-profile. These measurements are usually done by using the *token bucket* technique.

The best way to explain the *token bucket* technique is to symbolically consider that we have a bucket and tokens that are inserted or extracted from the bucket. The tokens are inserted into the bucket at the rate of x tokens/s and a token is removed from the bucket whenever a packet arrives at the router. The bucket has a capacity of k tokens. When a packet arrives, if there is at least one token to be extracted from the bucket, the packet is considered to be in-profile, but if the bucket is empty, the packet is considered out-of-profile. This technique allows the acceptance of bursty traffic up to a certain limit on the duration of the burst. The policing action can be followed by marking or not, this depending on the router implementation and also on the classification of the packet.

2.3 Management of queues

The router queue manager is responsible for the establishment and maintenance of the queues in the router.

The functions of the queue manager are: i) to insert a packet into the queue related to the packet class if the queue is not full; ii) to discard the packet if the queue is full; iii) to extract a packet from the queue when requested by the scheduler; iv) optionally, to perform an active management of the queue by monitoring the queue filling level and try to keep that filling level within acceptable limits, either by discarding or by marking packets.

An active management of the queues, although optional, is a recommended practice, as it allows accepting some traffic bursts without losing packets and can also diminish the packet delay in the router. There are several techniques to actively manage the router queues. We will mention some of the most relevant ones, namely, *Random Early Detection (RED), Weighted RED (WRED)* and *Adaptive RED (ARED).*

It is known that the best solution to control the filling level of a queue shared by different flows of packets is to statistically generate feedback signals, whose intensity is a function of the average filling level of the queue [5].

The RED technique [6] utilizes the average filling level of the queue, as a parameter for a random function, which decides whether the mechanisms that avoid the queue overload must be activated. For a queue occupancy up to a certain threshold (*min*), all the packets remain in the queue. For a filling level above *min,* the probability of discarding packets rises linearly until a maximum filling level (*max*). Above *max* all the packets are discarded. The average filling level is recalculated whenever a packet arrives.

The WRED technique uses an algorithm that is an evolution of RED by "weighting" packets differently according to their marking. The RED algorithm still applies, but now the values of *min* and *max* depend on the packet being marked or not. For marked packets the values of *min* and *max* are lower than for unmarked ones, therefore, there is a more aggressive discard policy for the marked packets.

Finally, the ARED technique is also based on an algorithm derived from RED. In this case, the RED parameters are modified based on the history of occupancy of the queue. ARED adjusts the aggressiveness of the probability of packet dropping based on the more recent values of the average filling level of the queue. This provides a more controlled environment for the management of the queue occupancy.

2.4 Scheduling

Scheduling is the mechanism that decides when packets are extracted from the queues to be sent to a router output port. There are different degrees of complexity for the implementation of schedulers. The simplest ones have the only objective of serving queues in a certain sequence, without caring about the output rate of each queue. The more complex schedulers have the objective of guaranteeing a minimum rate for certain queues and continuously adapt its serving sequence for this purpose.

The simplest schedulers are the *Strict Priority* schedulers. The queues are ordered by decreasing priority and a queue with a certain priority is only served if the queues with higher priority are empty. To avoid that the queues with less priority are never served, the upstream routers must have mechanisms of policing to assure that the higher priority queues are never working at full capacity. If the scheduler is busy and a packet arrives at a higher priority queue, the scheduler completes the present transmission and only then serves the higher priority queue. This a useful mechanism for services that require a low delay. The maximum delay value depends on the output link speed and on the maximum length of the packet.

Another simple scheduling mechanism is the *Round Robin.* The scheduler serves the queues in a cyclic order, transmitting one packet before serving the next one. It jumps over empty queues. In *Round Robin* it is difficult to define limits for delays, but it assures that all the queues are served within a certain time.

The *Strict Priority* and *Round Robin* mechanisms do not take into consideration the number of bits transmitted each time a queue is served. As the packets have variable length, these two mechanisms cannot be used to control average rates for the different traffic classes. The control of the rates requires that the service discipline of the scheduler adapts dynamically to the number of bits transmitted from each queue.

The *Deficit Round Robin (DRR)* scheduling mechanism [7] is a variant of the *Round Robin.* It considers the number of bytes transmitted from a certain queue, compares that number with the number of bytes that should have been transmitted (to achieve a certain rate) and takes that difference as a deficit. This deficit is used to modify the service duration of the queue the next time it is served.

Weighted Fair Queueing (WFQ) [8] is also a variant of *Round Robin.* It continuously recalculates the scheduling sequence to determine the queue that has more urgency in being served to meet its rate target. It also gives different weights to each queue. In WFQ and DRR the average rates are only achieved after the transmission of many packets.

3. THE INTEGRATED SERVICES MODEL

The Integrated Services (IntServ) model was the first network model to be considered to improve the IP best-effort network towards the support of real-time services. This model is defined in [9]. Integrated Services is explicitly defined as an Internet service model that includes best-effort service, real-time service and controlled link sharing. Link sharing means to divide the traffic into different classes and assign to each of them a minimum percentage of the link bandwidth under conditions of overload, while allowing unused bandwidth to be available at other times.

Besides the best-effort service, there are two other classes of service supported: Guaranteed Service [10] and Controlled Load Service [11]. The Guaranteed Service (GS) is for real-time applications with strict requirements for bandwidth and delay. The Controlled Load (CL) service is for applications that require a performance equivalent to the one offered by a best-effort network with a low traffic load.

The IntServ model requires the processing of the traffic in every router along an end-to-end path and also requires a signalling protocol to indicate the requests from each flow. A flow is defined as a set of packets from a source to one or more receivers for which a common QoS is required. This might apply to packets that have the same source/ destination addresses and port numbers.

The IntServ model consists of a sequence of network elements (hosts, links and routers) that, altogether, supply a transit service of IP packets between a traffic source and its receivers. If there is a network element without QoS control it will not contribute to the IntServ. Before sending a new flow of packets into the network, there must be an admission control process in every network element along the end-to-end path. The flow admission is based on the characterisation of the traffic made by the source.

The IntServ applications are classified in real-time tolerant, real-time intolerant and elastic. As suggested by the name, tolerant real-time applications do not require strict network guarantees concerning delay and jitter. In elastic applications the packet delay and jitter in the network are not so important.

The GS service provides firm bounds on end-to-end delays and it is appropriate for intolerant real-time applications. An application indicates its expected traffic profile to the network, which evaluates the end-to-end maximum delay value that can guarantee and gives that indication to the application. The application decides whether that delay value is adequate and, in the affirmative case, proceeds by sending the flow of packets.

The CL service is defined by the IETF as a service similar to the best-effort service in a lightly loaded network. This service is adequate for real-

time tolerant and elastic applications. Of course, many of the elastic applications can also be adequately served by the best-effort service.

The signalling protocol is a key element in the IntServ model, as it is used for doing resource reservation in the network routers. The signalling protocol makes resource reservation in two steps. The first one is admission control and the second one is configuration of the network elements to support the characteristics of the flow. The Resource Reservation protocol (RSVP) [12] has been selected as the signalling protocol for IntServ.

As schematically shown in figure 3, sources emit PATH messages to the receivers. Each PATH message contains two objects, *Sender_Tspec* and *Adspec,* respectively. The first object is the traffic descriptor and the second one describes the properties of the data path, including the availability of specific QoS control characteristics. The *Adspec* object can be modified in each router to reflect the network characteristics. The receivers reply with RESV messages to the source. A RESV message carries the object *Flowspec,* which contains the QoS expected by the receiver and to be applied to the source traffic.

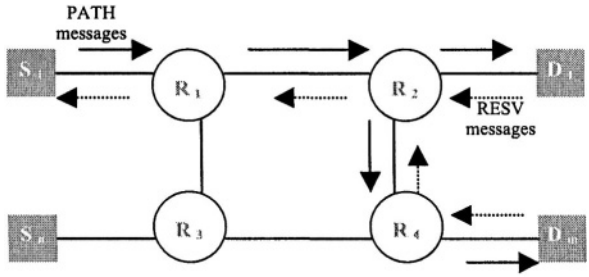

Figure 3. RSVP operation

To start a reservation, the source of the flow defines the *Sender_Tspec* and *Adspec* parameters and inserts them in a PATH message. At the receivers, *Sender_Tspec* and *Adspec* are used to determine the parameters to send back in the *Flowspec* object. In *Flowspec* it is indicated whether CL or GS is selected and it also carries the parameters required by the routers along the path, so that they can determine whether the request can be accepted. RSVP is appropriate for multicast operation.

All the routers along the path must do local measurements, followed by policing, so that the agreed bounds can be achieved. The resource reservation mechanism is independent of the routing algorithm. The RSVP messages circulate along the routes previously established by the routing algorithm.

4. THE DIFFERENTIATED SERVICES MODEL

The IntServ model is conceptually a good model to support both the real-time and non-real-time services in the Internet. However, in practice, this model is not scalable for the Internet. Its deployment would require to keep states in the routers for every flow and also to process these flows individually, which is very difficult to achieve. This was the main reason for the definition of another IP network model, the Differentiated Services (DiffServ) model [13]. DiffServ represents an incremental improvement of the best-effort service. It is a minimalist solution compared to IntServ, but it is scalable.

The DiffServ network structure is shown in figure 4. A network has edge and core routers. The edge routers map the customer's traffic into the core routers, whose main function is to transport packets to other routers until the egress edge router. The egress edge router communicates with the customer's terminal.

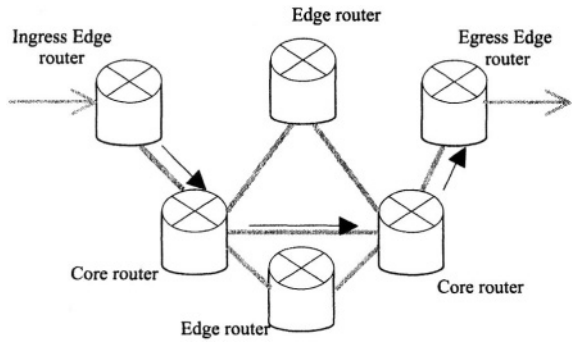

Figure 4. The DiffServ network model

The edge routers classify and police the customer's traffic before sending it to the network. The edge routers can refuse requests, therefore, transitory overloads can be solved. The more complex decisions are taken in the edge routers, simplifying the structure of the core routers, which implies that we can have faster core routers. Also we will have a smaller number of states than in IntServ as the packet context is established only from the DSCP field (see figure 2). The classification done in the edge routers allows that a large variety of traffic can be mapped into a small set of behaviours in the core network. In the DiffServ terminology, a collection of packets with the same DSCP is called DiffServ Behaviour Aggregate.

DiffServ introduces the concept of Per Hop Behaviour (PHB). Basically the PHB is the specific behaviour of the queue management and scheduling mechanisms in a network element. The concatenation of the different PHBs between an ingress and an egress edge router in the network defines the expected behaviour of the network and permits to define a Service Level Agreement with the customers.

DiffServ supports two distinct classes of PHBs besides best-effort. They are named Expedited Forwarding (EF) [14] and Assured Forwarding (AF) [15]. They are distinguished by the different coding values of the DSCP field. All bits with the value 0 in DSCP means a best-effort PHB.

EF PHB is defined by the code 101110 in the DSCP. This PHB is the most stringent one in DiffServ and is used for services that require low delay, low jitter and small packet loss. EF PHB requires co-ordination among the mechanisms of policing and scheduling along the path to be used by the EF packets. This service is sometimes also known as Premium service.

The AF PHB is less stringent than EF and is specified in terms of relative availability of bandwidth and characteristics of packet loss. It is adequate to support bursty traffic. In AF there are two types of context encoded in the DSCP: service class of the packet and precedence for the packet loss. The service class of the packet defines the router queue where it will be inserted. The loss precedence influences the weight allocated to the queue management algorithm, making this algorithm more or less aggressive towards packet discarding.

The first three bits of DSCP define the service class and the next two bits define the loss precedence. The sixth bit is fixed at 0. The standard defines four service classes and three loss precedence levels as shown in table 1. More classes and precedence levels can be defined for local use.

	Class 1	Class 2	Class 3	Class 4
Precedence 1 (low)	001 010	010 010	011 010	100 010
Precedence 2 (medium)	001 100	010 100	011 100	100 100
Precedence 3 (high)	001 110	010 110	011 110	100 110

Table 1. DiffServ classes and loss precedence levels

As the AF PHB is the one advised for the support of data applications, it is important to understand the interaction of this mechanism with TCP. Some authors claim that some improvements need to be done at the DiffServ

level in order that TCP performance is not diminished [16]. This is a subject that requires further study.

The DiffServ model is simple and, therefore, attractive for deployment in the Internet. However, the mapping of a large number of flows into a limited number of PHBs requires techniques that are very dependent on the network topology and QoS characteristics of the routers, namely the classification, queue management and scheduling mechanisms.

5. INTEGRATED SERVICES OVER DIFFSERV NETWORKS

The IntServ model supports the delivery of end-to-end QoS to applications in an IP network. An important factor, however, has not allowed a large deployment of IntServ in the Internet. It has to do with the requirement for per-flow state and per-flow processing, which raises scalability problems.

On the other hand, the IntServ model is supported over different network elements. A DiffServ network can be viewed as one of these network elements, which exist in the end-to-end path between IntServ customers. As we know, the main benefit of DiffServ is to eliminate the need of per-flow state and per-flow processing and, therefore, making it a scalable model. In this context, IntServ and DiffServ can be used together to create a global end-to-end solution. In this global solution it is possible to have IntServ signalling between the hosts and the ingress router to the DiffServ network so that the router can indicate to the host whether there is enough network capacity to transport the packets related to the service. This capacity is provisioned during the configuration of the DiffServ network. The state information is only treated at the IntServ level.

The IntServ/DiffServ network configuration is shown in figure 5 [17].

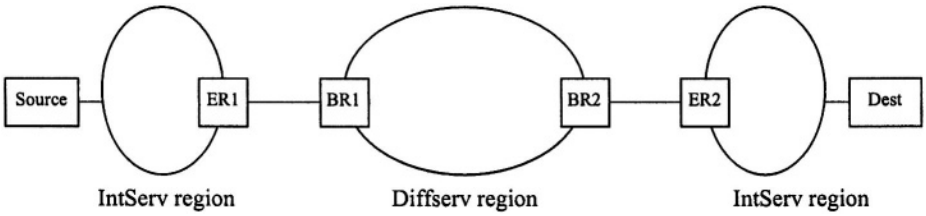

Figure 5. Reference IntServ/DiffServ configuration

The model distinguishes between edge routers (ER) and border routers (BR). Edge routers are egress/ ingress routers in the IntServ regions. Border

routers are ingress/ egress routers in the DiffServ regions. The border routers are the ones that map the DiffServ ingress traffic into the network core routers (not represented in the figure). The RSVP signalling generated by the hosts is carried across the DiffServ regions. The signalling messages may be processed or not by the DiffServ routers. If the DiffServ region is RSVP-unaware, the border routers act as simple DiffServ routers, doing no processing of the RSVP messages. Edge routers do the admission control to the DiffServ region. If the DiffServ region is RSVP-aware, the border routers participate in RSVP signalling and do admission control for the DiffServ region.

This model to support QoS in an IP network is an attractive compromise, but some additional work still needs to be done, mainly concerned with the mapping of IntServ services to the services provided by the DiffServ regions, with the need for the deployment of equipments, named bandwidth brokers, that can provide resources in a DiffServ region in a dynamic and efficient way and for the support of multicast sessions with this network model [18].

6. MULTIPROTOCOL LABEL SWITCHING

Multiprotocol Label Switching (MPLS) provides traffic control and connection-oriented support to IP networks. These capabilities allow the provision of a basic connection-oriented mechanism to support QoS, ease the provision of traffic engineering in the network and also support the provision of Virtual Private Networks at the IP level [19].

MPLS must be clearly distinguished from the IP network models (IntServ, DiffServ) previously defined. The IntServ and DiffServ models are defined at the IP level, whereas the MPLS protocol runs below the IP level. MPLS configures the network to transport IP packets in an efficient way.

MPLS was preceded by other technologies, namely IP Switching from Ipsilon, ARIS from IBM, Tag Switching from Cisco and CSR from Toshiba. These different technologies had aims similar to MPLS and now they have been superseded by the MPLS standard defined at IETF [20].

IP packets are partitioned into a set of the so-called Forwarding Equivalent Classes (FEC). As defined in the standard, a particular router will consider two packets to be in the same FEC if there is some address prefix X in that router's routing tables such that X is the longest match for each packet's destination address. All packets which belong to a certain FEC and which travel from a particular node will follow the same path in the network. In MPLS, the assignment of a certain packet to a FEC is done at the network entry. The FEC is encoded as a label, which is appended to the packet

header. This label is used in the network to switch the packets in the different routers which are MPLS-capable. These MPLS-capable routers are named Label Switching Routers (LSR) and have switching tables that operate using the packet label as an index to a table entry, which determines the next hop and a new label. MPLS simplifies the forwarding of packets in the network and allows explicitly sending a packet along a certain existing route. This latter technique is known as traffic engineering.

The MPLS label is a 32-bit field as shown in figure 6. The first 20 bits define the label value, which is defined at the network entry depending on the FEC to which the packet belongs. The label value has only local significance. It is changed by the LSRs in the switching process. The experimental bits are reserved for local use, the stack bit is used when labels are stacked and the Time to Live (TTL) field establishes a limit for the number of hops. The TTL field is important because the usual TTL function is encoded in the IP header, but the LSR only examines the MPLS label and not the IP header. By inserting TTL bits in the label, the TTL function can be supported in MPLS. If MPLS runs over a connection-oriented layer 2 technology, such as ATM or Frame Relay, the label value is inserted in the VPI/VCI field of ATM or in the DLCI field of Frame Relay.

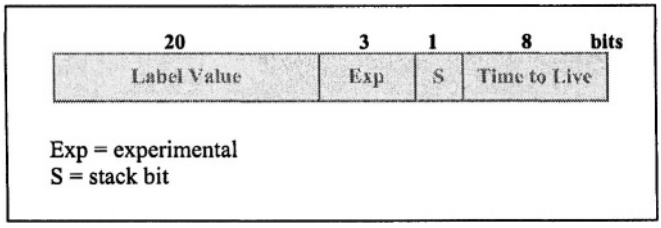

Figure 6. MPLS label format

The operation of MPLS can be described as follows. Initially, a path must be established in the network to send the packets of a given FEC. This path is known as Label Switched Path (LSP). The establishment of the LSP can take into consideration the resource allocation to be done in the network routers having in view the support for QoS provision. To establish this path, two protocols are used. The first one is the routing protocol, typically OSPF, which is used to exchange reachability and routing information. The second one is used to determine which route to use and which label values must be utilised in adjacent LSRs. This latter protocol can be the Label Distribution Protocol (LDP) or an enhanced version of RSVP (RSVP-TE). Alternatively, instead of using LDP or RSVP-TE, an explicit route can be provisioned by a network operator, which will assign the adequate label values.

When a packet enters the MPLS domain, the LSR assigns the packet to a certain FEC, and implicitly to an LSP, and inserts the MPLS label into the packet. The next action is to forward the packet. Within the MPLS domain, when an LSR receives a packet, the switching table is accessed, the label is substituted by a new one and the packet is forwarded to the next hop. Finally the egress LSR removes the label, examines the IP header and forwards the packet to the destination.

MPLS can be used to efficiently support the transport of packets in a DiffServ network [21]. At the ingress of a DiffServ network the IP packets are classified and marked with a DSCP, which corresponds to their Behaviour Aggregate. At each router the DSCP is used to select the respective PHB. RFC 3270 specifies how to support the DiffServ Behaviour Aggregates whose corresponding PHBs are currently defined over an MPLS network. It specifies the support of DiffServ for both IPv4 and IPv6 traffic, but only for unicast operations. The support of multicast operations is currently under study.

7. QUALITY OF SERVICE IN THIRD GENERATION WIRELESS NETWORKS

Third Generation wireless networks, also known in Europe as Universal Mobile Telecommunications System (UMTS), are a good example of information networks. Whereas second generation wireless networks were optimized for the communication of voice, third generation networks focus on the communication of information, including all the types of services. This requirement to transmit information in all its forms implies that the circuit switched based network architecture of second generation networks has to include also a packet switched part in its evolution towards a third generation network architecture.

The UMTS network architecture has been defined by 3GPP (Third Generation Partnership Project). 3GPP has planned the evolution of the network according to a series of releases. The first one to be implemented is known as Release 99 [22]. A simplified view of the UMTS architecture, according to Release 99, is shown in figure 7.

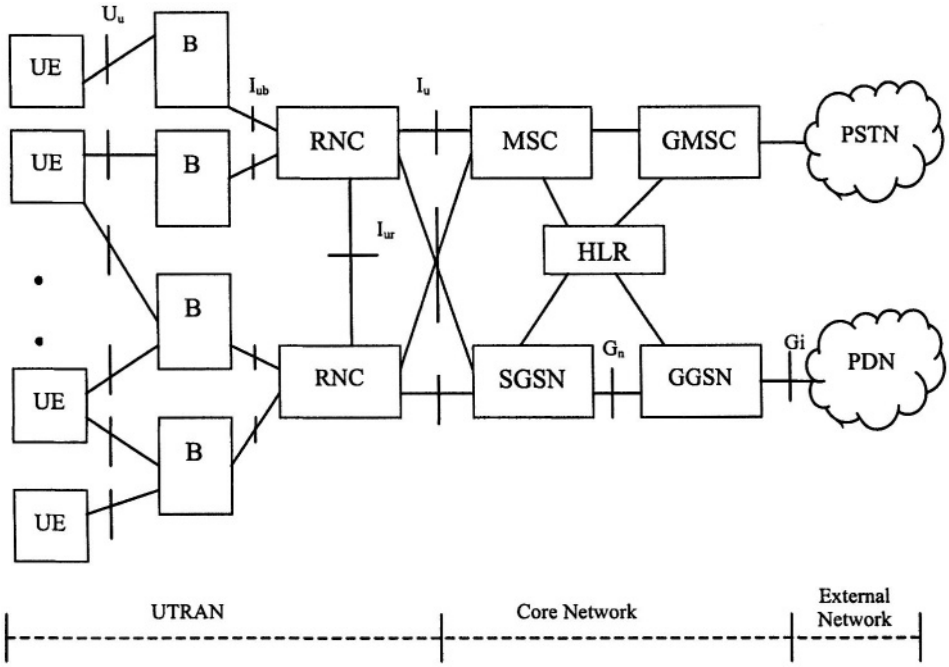

Figure 7. UMTS network architecture

The structure of a UMTS network consists of two main levels: radio access network and core network. They are separated by the Iu interface.

The Universal Terrestrial Radio Access Network (UTRAN) consists of a set of Base stations, known as nodes B, and a set of Radio Network Controllers (RNC). Each RNC controls a number of nodes B. Iub is the interface between a node B and an RNC. The RNCs may communicate between themselves via the Iur interface. The radio access part is comprised between the User Equipment (UE) and the nodes B (interface Uu). The RNC is the switching and control element of the UTRAN. Each RNC is respectively connected, via the Iu interface, to the Mobile services Switching Centre (MSC) and Serving GPRS Support Node (SGSN), which are two elements of the Core network.

The Core network consists of a circuit switched domain and a packet switched domain. The main elements in the circuit switched domain are the MSC and the Gateway MSC (GMSC). The MSC is responsible for the circuit switched connection management activities. The GMSC takes care of the connections to other PSTN networks. In the packet switched part, there are also two main elements, the SGSN and the Gateway GPRS Support

Node (GGSN), separated by the Gn interface. The SGSN supports packet communication towards the access network and is responsible for mobility management related issues. The GGSN maintains the connections towards other packet data networks, such as the Internet, via the Gi interface. The Home Location Register (HLR) contains the addressing and identity information for both the circuit and packet switched domains of the core network.

The problem of QoS provision in UMTS is particularly relevant for mobile packet switched based services, which constitute the main novelty introduced in UMTS networks compared to the previous generation of circuit switched wireless networks. The Core network circuit switched domain uses signalling protocols inherited from GSM. The Core network packet switched domain can be seen as an IP backbone internal to the operator network.

The end-to-end services are carried over the network using bearers. A bearer is a service providing QoS between two defined points. As the radio access network and core network have their own QoS properties, the QoS needs to be treated separately in each of these levels. The end-to-end QoS is the global result, which takes into account the distinct levels of the network.

In UMTS a specific medium access control protocol is used on the radio bearers, which link the UEs to the base stations. From the base stations to the core network, the transport of packets is done over ATM. In the core network, the information is encapsulated in IP; here, the QoS is treated according to the DiffServ model. The layer 2 protocols in the core network, which will transport the IP packets, are not standardized, although, in practice, ATM might be one of the main choices of network operators for this purpose.

In UMTS there is one additional feature, which consists in the UEs having the ability to negotiate the QoS parameters for a radio bearer. The negotiation is always initiated by the application in the UE and the network checks whether it can provide the required resources or if it rejects the request.

After the deployment of release'99, new releases are foreseen to upgrade UMTS networks in the future [23] [24]. The upgrade of the UMTS network aims, in a first phase, to evolve the whole core network into a packet switched architecture based on IP. This means that we will have voice over IP in the core network after the first phase of evolution is accomplished. The final aim is to have an "All-IP" network including the radio part. Therefore, we would have an end-to-end IP network to support the applications. Of course, this network would need to consider all the aspects covered in the previous chapters of the paper to achieve a satisfactory QoS for all types of services. Although this is the aim, it might still take some time to achieve it,

due to the characteristics of the air interface, where the bandwidth availability is at a premium, which requires optimization of the mechanisms to provide QoS.

8. CONCLUSIONS

The problem of provisioning QoS in information networks is not completely solved yet. As seen in the previous chapters, the evolution of an IP best-effort network into a network that can provide QoS guarantees is not an easy task. Some significant steps have already been given, but research continues active in this field. As described next, the use of signalling protocols, the evolution towards IPv6 and the convergence of IP with existing networks are good examples of current research work in this area.

As we know, resource allocation in the network elements is required to comply with bounds in the values of the different QoS parameters. Resource allocation can be done by provisioning the network, but provisioning is neither flexible nor dynamic. Network operation would be more effective if a dynamic and flexible solution based on signalling could be implemented. One of the protocols that is often referred for this purpose is RSVP. Some extensions have been proposed to RSVP to provide additional features, namely security, more scalability and new interfaces. One well-known extension is the so-called RSVP-TE, which is used in MPLS to establish explicitly routed LSPs. Other protocols have also been proposed, such as YESSIR and Boomerang [25]. All these signalling protocols apply to the intra-domain level. If we wish to consider also inter-domain signalling, which is the global scenario, other signalling protocols need to be considered. BGRP is a signalling protocol for inter-domain aggregated resource reservation for unicast traffic [26]. Other inter-domain protocols under study are SICAP [27] and DARIS [28]. The comparative efficiency of all these protocols to serve the different types of services is under evaluation [29].

Currently, IP networks use IPv4. A new version of the protocol (IPv6) is ready since about ten years ago. Although the main new feature of IPv6 is a larger IP addressing space (128 bits instead of 32 bits), there are also new fields in the IP header that can be used to facilitate the QoS support. However, the introduction of IPv6 in the existing networks has not been done yet at a large scale. The best strategy of introducing IPv6 in the running networks is still under discussion as well as the best way of taking advantage of its new features [30] [31].

The support of the convergence of IP networks with other networks, such as the PSTN, is key to the success of information networks. This is an issue that has been under study in standardization bodies, namely at the ITU-T [32]. There is a need to coordinate the sharing of resources, which are done with different signalling protocols, in distinct operating domains.

Many other items related to the evolution of IP-based information networks are currently under study in several research projects, e.g. [33] and in standardization bodies, namely the IETF [34]. This study has a broad spectrum and extends from routing and transport to security issues in IP-based networks.

REFERENCES

[1] G. Armitage, Quality of Service in IP Networks, Macmillan Technical Publishing, 2000.
[2] P. Almquist, Type of Service in the Internet Protocol Suite, RFC 1349, IETF, July 1992.
[3] S. Dearing and R. Hinden, Internet Protocol Version 6 Specification, RFC 2460, IETF, December 1998.
[4] K. Nichols et al, Definition of the Differentiated Services Field in the IPv4 and IPv6 Headers, RFC 2474, IETF, December 1998.
[5] B. Braden et al, Recommendations on Queue Management and Congestion Avoidance in the Internet, RFC 2309, IETF, April 1998.
[6] S. Floyd and V. Jacobson, Random Early Detection Gateways for Congestion Avoidance, IEEE/ACM Transactions on Networking, no. 4, August 1993.
[7] M. Shreedar and G. Varghese, Efficient Fair Queueing Using Deficit Round Robin, ACM Sigcomm 95, October 1995.
[8] A. Demers et al, Analysis and Simulation of a Fair Queueing Algorithm, ACM Sigcomm89, September 1989.
[9] R. Braden et al, Integrated Services in the Internet Architecture: an Overview, RFC 1633, IETF, June 1994.
[10] S. Shenker et al, Specification of Guaranteed Quality of Service, RFC 2212, IETF, September 1997
[11] J. Wroclawski, Specification of the Controlled Load Service, RFC 2211, IETF, September 1997.
[12] J. Wroclawski, The Use of RSVP with IETF Integrated Services, RFC 2210, IETF, September 1997.
[13] S. Blake et al, An Architecture for Differentiated Services, RFC 2475, IETF, December 1998.
[14] V. Jacobson et al, An Expedited Forwarding PHB, RFC 2598, IETF, June 1999.
[15] J. Heinanen et al, Assured Forwarding PHB Group, RFC 2597, IETF, June 1999.

[16] P. Giacomazzi, L. Musumeci and G. Verticale, Transport of TCP/IP Traffic over Assured Forwarding IP-Differentiated Services, IEEE Network Magazine, Vol. 17, No.5, September/ October 2003.
[17] Y. Bernet et al, A Framework for Integrated Services Operation over Diffserv Networks, RFC 2998, IETF, November 2000.
[18] K Nichols et al, A two bit Differentiated Services Architecture for the Internet, RFC 2638, IETF, July 1999.
[19] William Stallings, MPLS, The Internet Protocol Journal, Volume 4, Number 3, September 2001.
[20] E. Rosen et al, Multiprotocol Label Switching, RFC 3031, IETF, January 2001.
[21] F. Le Faucheur et al, MPLS Support of Differentiated Services, RFC 3270, IETF, May 2002.
[22] 3GPP TS 23.002 V3.4.0, Network Architecture (Release 1999), December 2000.
[23] 3GPP TS 23.107, QoS Concept and Architecture (Release 4), June 2001.
[24] 3GPP TS 23.207, End-to-end QoS Concept and Architecture (Release 5), June 2001.
[25] J. Manner, Analysis of Existing Quality of Service Signalling Protocols, Internet-Draft, IETF, October 2003.
[26] P. Pan et al, BGRP: A Tree-Based Aggregation Protocol for Inter-domain Reservations, Journal of Communications and Networks, Vol. 2, No. 2, June 2000
[27] R. Sofia, R. Guerin, and P. Veiga. SICAP, A Shared-segment Inter-domain Control Aggregation Protocol, High Performance Switching and Routing Conference, Turin, Italy, June 2003.
[28] R. Bless, Dynamic Aggregation of Reservations for Internet Services, Proceedings of the Tenth International Conference on Telecommunication Systems - Modelling and Analysis, Volume One, Monterey, USA, October 2002.
[29] R. Sofia, R. Guerin, and P. Veiga. An Investigation of Inter-Domain Control Aggregation Procedures, International Conference on Networking Protocols, Paris, France, November 2002.
[30] M. Tatipamula, P. Grossetete and H. Esaki, IPv6 Integration and Coexistence Strategies for Next-Generation Networks, IEEE Communications Magazine, Vol. 42, No. 1, January 2004.
[31] Y. Adam et al, Deployment and Test of IPv6 Services in the VTHD Network, IEEE Communications Magazine, Vol. 42, No. 1, January 2004.
[32] N. Seitz, ITU-T QoS Standards for IP-Based Networks, IEEE Communications Magazine, Vol. 41, No. 6, June 2003.
[33] Euro NGI Network of Excelence, Design and Engineering of the Next Generation Internet; http://www.eurongi.org
[34] Internet Engineering Task Force; http://www.ietf.org/