PROTOCOL OR HEADER INDICATION – METHODS TO REQUEST QUALITY OF SERVICE FOR IP-APPLICATIONS

Gerald Eichler[1] *Ralf Widera*[2]
T-Nova Deutsche Telekom Innovationsgesellschaft mbH, Technologiezentrum Darmstadt
Am Kavalleriesand 3, D-64295 Darmstadt, Germany,
[1]e-mail: Gerald.Eichler@telekom.de, [2]e-mail: Ralf.Widera@telekom.de

ABSTRACT

The modern internet-based multi-media world is looking for reliable IP transport mechanisms to provide applications with their required quality. Besides signalling protocols, simple indication methods allow to build a quality of service framework including access and core network components which is scalable and very flexible to handle.

Starting from architectural bases, possible access *indicators* are analysed and evaluated. A light Differentiated Services solution using limited *network services* in combination with *profiling* forms the base to carry traffic over wide area networks and to support interaction of different access methods.

The theoretical results will be verified within the European AQUILA project which is part of the EU's IST research programme.

1 INTRODUCTION

Usually the user or the application implicitly knows best about their transmission quality of service (QoS) requirements. However, once an IP packet is delivered to the network, any background information is lost.

There are different ways for a user or application to communicate requirements towards the network. Both static and dynamic methods, as well as header-based and protocol-based solutions will be taken into consideration for this paper.

Not only technical reasons require a certain QoS, but also business reasons fixed in service level agreements (SLA) enforce the classification of IP packets. The performance and efficiency of appropriate packet handling benefit from the multiplex gain of aggregation within large scale networks.

2 ARCHITECTURAL BASES

QoS is expected end to end from host to host. Usually the access network and the core network follow different construction norms and are administrated by different parties. Within diverse access networks various ways to ask for QoS are imaginable.

There is a component which forms the borderline between access and core that is named Edge Router (ER). To bring access and core together the following solution is proposed.

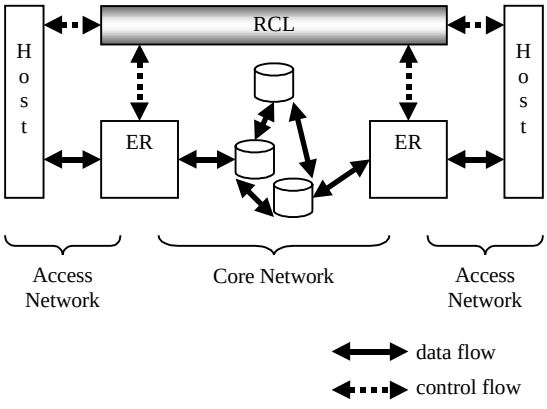


Figure 1. Access and Core Network.

While the core network handles all packets based on a limited number of service classes autonomously, various methods within the access domain are foreseen. The set of methods is extendable on-

demand. Rules for the combination of different methods have to be defined carefully.

An extra layer which is called Resource Control Layer (RCL) is introduced, see Figure 1 to allow a smooth intercommunication of hosts which share a QoS aware application. The RCL takes care of the following QoS sensitive functions:

- admission control,
- resource pool allocation,
- host communication,
- proxies,
- mapping database,
- Edge Router triggering.

The functions of the RCL will be implemented by means of several additional components. Implementation issues are presented in Chapter 5.1.

3 INDICATION AND PROTOCOLS AT THE ACCESS NETWORK

There are different kinds of methods for a host to request QoS. One group is based on layer 3 and 4 packet header information. This can be seen as in-band (implicit) signalling. Some of these indicators relate to the host, namely address information, others to the application/session like protocol number or port numbers. Beside this the use of the type of service (ToS) field allows dynamic QoS requests.

The other group uses out-of-band (explicit) signalling by means of a signalling protocol. The Resource Reservation Protocol (RSVP) is the most typical example.

A granularity classification describes the flexibility to change QoS requirements. It can be done statically on a per host base or dynamically per application/session or, even more flexible, changed per IP packet. The latter is of interest for applications that rely on different micro-flows.

Definition 1. *An **indicator** is an information element which implies a message about a requested QoS handling of an IP packet or flow.*

Indicators are not self-describing but need an interpretation on defined rules.

3.1 Header-based Indicators

The IP header [1] and higher level 4 headers include fields which can be analysed for QoS handling decisions. Principally each defined bit pattern of an IP packet could be traced, but some fields of the IP header are more meaningful. They are marked grey in Figure 2.

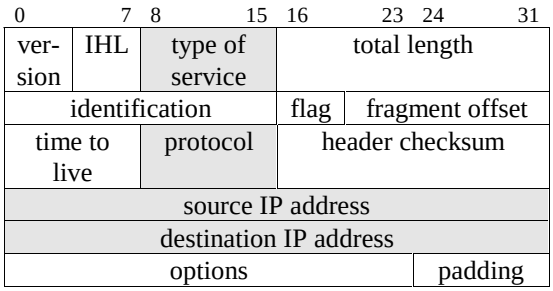


Figure 2. IPv4 Packet Header.

3.1.1 Address-based QoS Indication

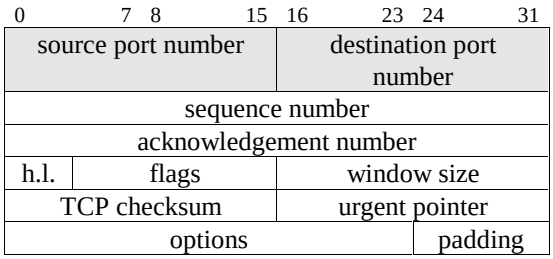
Address-based QoS indication identifies packets from a certain source and/or to a certain destination. This is practically applicable if no address translation is performed. Besides single addresses there are also ranges of addresses e.g., addresses of a subnetmask, will be taken into account.

3.1.2 Protocol Type-based QoS Indication

A transmission protocol, which is represented by an extra header, is added to IP packets on layer 4. This protocol is indicated within the IP header protocol field. Dependent on the protocol, typically TCP, UDP or RTP, QoS mechanisms can be applied.

3.1.3 Port Number-based QoS Indication

The TCP header (Figure 3) and the UDP header (Figure 4) contain source and destination port numbers which typically specify the required service, e.g. TCP port 21 = FTP.



h.l. header length

Figure 3. TCP Packet Header.

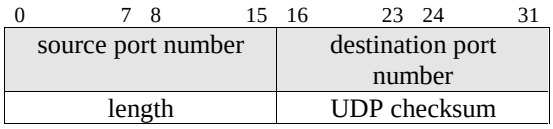
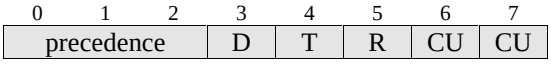


Figure 4. UDP Packet Header.

Instead of single port numbers port number ranges are sometimes required for the same service. Therefore, in addition to single port numbers, port number masks are proposed as indicators as they are known from firewall systems.

3.1.4 Host Marking

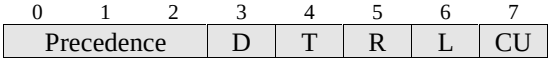
Traditionally, the ToS field (or ToS byte) was designed to carry information about eight precedence levels (3 bit code) as well as delay, throughput and reliability priority (1 bit indication each) as depicted in Figure 5 [3].



D delay
T throughput
R reliability
CU currently unused

Figure 5. Traditional use of the Type of Service Field

Bit 6 and 7 were left for further studies or local use. Another version of this interpretation specifies bit 6 as the "low cost bit" as depicted in Figure 6.



D delay
T throughput
R reliability
L low cost
CU currently unused

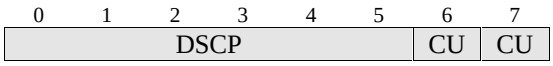
Figure 6. Type of Service Field with low cost bit.

The interpretation of the bit settings of the host is up to the RCL and ER. One possible exploitation is direct coding of network services (see Chapter 4) within the precedence bits of the ToS field.

As host marking allows different interpretations of the ToS bits, an interpretation scheme must be associated with applications using this method. It can be done in combination with the IP address that unambiguously identifies the host. Use of the remaining CU bit(s) is another possible solution to select one of multiple stored interpretation schemes.

3.1.5 Differentiated Services Model (DiffServ)

The traditional type of service approach has been changed several times and has come up with the Differentiated Services (DiffServ) model [2], [3]. The idea is to handle the first 6 bits together as Differentiated Services Codepoint (DSCP), see Figure 7. The ToS field is now called DS-Byte. A maximum of 2^6 = 64 classes will be supported. The DiffServ approach is a special case of host marking.



DSCP Differentiated Services Codepoint
CU currently unused

Figure 7. DiffServ Interpretation of the ToS Field.

3.2 Indication by Signalling Protocols

A more specific request is possibly given by using signalling protocol-based methods. The Integrated Services (IntServ) model [4] using RSVP is still under strong discussion within the IETF for IP QoS requests.

The appropriate handling of PATH/RESV messages, which are the basic RSVP signalling messages, requires RSVP aware routing entities within the core network. Because it is a soft-state protocol it is flexible concerning the adaptation to session requirements. But on the other hand the implementation requires much overhead which often results in performance or scalability problems. Compared to other signalling protocols RSVP is receiver and not sender oriented.

3.3 Organisation of Indicators

	Static indicators (Administrative issue)		Dynamic indicators (Operation issue)	
	Host/ Network sensitive	Session sensitive	During Session changeable	Per Packet changeable
Header-based (implicit)	▪ source IP address ▪ destination IP address ▪ IP address subnetwork	▪ protocol type ▪ source and destination port number ▪ port number ranges		▪ DiffServ ▪ host marking
Protocol-based (explicit)			▪ RSVP	

Table 1. Organisation of QoS Indicators.

Table 1 shows a systematic organisation related to header and protocol-based indication on the one hand and static and dynamic indicators at the other hand.

3.4 Exploitation of Indicators

Indicators are not self-describing. Their presence needs an interpretation rule. In case of the presence of more than one indicator, these rules must be

cascadable and have to form a well defined order. How to handle packets which meet a certain rule is lodged in a database.

These database entries will also associate more technical parameters which are necessary to calculate the resources and influence router queuing and scheduling settings, e.g.:

- bandwidth parameters,
- packet size,
- burstiness.

4 LIGHT DIFFSERV MODEL FOR THE CORE NETWORK

The core network must be able to carry packets with different QoS characteristics properly. Modern IP routers support a wide range of QoS features and allow flexible configurations with multiple queues and advanced dropping and scheduling mechanisms.

To be scalable, similar traffic will be aggregated within one network service. This meets the basic principle of IETF's DiffServ approach [3].

Definition 2. *A **network service** describes how customer's traffic is handled while passing a network. Traffic of similar characteristic is aggregated within one network service.*

4.1 Design of Network Services

There should be only a limited number of network services to guarantee efficient network operation, maintenance and monitoring. Depending on the network size, up to four network services are sufficient.

4.1.1 Network Service Characteristics

Network Service	Primary QoS Characteristic	Examples
PVO	delay sensitivity	▪ IP telephony ▪ network control
PMM	jitter sensitivity	▪ broadband streaming ▪ video conferencing
PMC	loss sensitivity	▪ SAP ▪ NetGames
STD	cost efficiency	▪ e-mail ▪ ftp

Table 2. Network Services' primary Characteristic.

Instead of communicating large sets of parameters which are often not independent, a network service should be characterised by its main focus. The following four network services form a practical example set:

- Premium Voice Service (PVO),
- Premium Multi-Media Service (PMM),
- Premium Mission Critical Service (PMC),
- Standard Service (STD).

Table 2 identifies the most critical quality criteria of the listed network services and gives two typical examples for applications.

Routing mechanisms have to be adapted carefully per network service to meet the expected QoS behaviour.

4.1.2 Admission Control and Profiling

As known from the DiffServ proposals, this approach provides only relative guarantees or better to say probabilities. To guarantee a sufficient performance a combination with admission control procedures is required which focuses on an adequate mixture of traffic of different network services on a large scale network.

As there is no exact knowledge of the traffic matrix traffic profiling is proposed. This is a "soft version" of admission control. If an agreed level of traffic is exceeded, packets will not be dropped immediately but marked as "out-of-profile".

Definition 3. ***Profiling** is a tagging mechanism which marks packets that exceed a certain agreed level of a defined parameter.*

As long as there is no congestion these packets are handled in the same manner as "in-profile" packets of the same network service. There can be several drop levels. If a congestion situation occurs at a routing entity, the packets with the highest drop level will be dropped first.

4.2 Coding of Network Services

Four network services and two profiling levels require 4 * 2 = 8 codepoints. The proposal is to use the three precedence bits for an implicit coding within each IP packet header. Figure 8 shows a possible coding scheme. Bit 0 becomes the profiling bit.

The rule, the higher the numeric precedence value the higher the importance, could not be completely considered, as the premium network services have different optimisation criteria which do not form a logical order.

The traditional ToS bits D, T, R and L, see Chapter 3.1.4, are untouched. This mechanism, sometimes called Class of Service (CoS) transparency, allows applications end-to-end to communicate about their QoS characteristics, independently from the network handling. This is an advantage compared to

the DiffServ approach where these bits are subject to per-hop modification.

0	1	2	3	4	5	6	7
precedence			D	T	R	CU	CU
1	1	1	PVO in-profile				
1	1	0	PMM in-profile				
1	0	1	PMC in-profile				
1	0	0	STD in-profile				
0	1	1	PVO out-of-profile				
0	1	0	PMM out-of-profile				
0	0	1	PMC out-of-profile				
0	0	0	STD out-of-profile				

Figure 8. ToS Field Coding of Network Services.

The limitation to eight codepoints implies another advantage. Many discussions focus on the exploitation of the Multi-Protocol Label Switching (MPLS) to carry IP traffic. MPLS is a transport encapsulation. An extra header located between the UDP/TCP and IP header, the 32-bit shim header, is added to each packet, see Figure 9.

0	19	20 22	23	24	31
label		exp	BoS	TTL	

label	MPLS label
exp.	experimental bis
BoS	bottom of stack
TTL	time to live

Figure 9. MPLS Shim Header.

The three experimental bits of the shim header can be used to carry the same coding for the network services on MPLS level as the ToS field of the IP header does.

4.3 Mapping onto Network Services

The decision which is the right network service for a single QoS request results from different factors and will be stored in a database. Basically they could be of technical or administrative nature. Table 3 collects technical and administrative aspects.

Technical factors	Administrative factors
▪ loss, delay, jitter ▪ robustness of the application ▪ data importance ▪ network load per path	▪ signed capacity per network service ▪ customer decision ▪ price model ▪ service level agreement

Table 3. Technical/Administrative Mapping Factors.

Practically, the mapping procedure onto one selected predefined network service results from the evaluation of active indicators. The separation into active and passive indicators is an administrative issue where only a subset of indicators is taken into consideration.

5 THE AQUILA PROJECT

An implementation of relevant proposed methods and solutions will be carried out in the AQUILA project [5], [6] which lasts from 2000 till 2002. AQUILA stands for "Adaptive Resource Control for QoS Using an IP-based Layered Architecture".

Two trial phases are planned during the project. Legacy and QoS aware applications will be supported. While the first trial phase is carried out on laboratory islands, the second will involve real users.

5.1 Additional Components at the Edge

To fulfil the tasks of the RCL mentioned in Chapter 2, additional functional components with well defined interfaces were designed. To get a rough feeling the End User Application Toolkit and the Admission Control Agent are shortly introduced [7].

5.1.1 End User Application Toolkit (EAT)

The EAT forms a platform independent middleware between end user applications and the network infrastructure in order to provide customers with QoS features. It will do both enrich legacy application by giving them access to QoS features and offer an API for QoS aware EAT-based applications. GUI support is planned.

5.1.2 Admission Control Agent (ACA)

The ACA is a complex entity which decides whether a QoS request can be fulfilled or not. It omits overload of the network and assigns resources in co-operation with other entities of the RCL to the customers. Its main components are a session manager, local admission control and an edge router adapter.

5.2 AQUILA Architecture

The AQUILA architecture has different QoS methods for the access and the core network in mind. The interaction of the components and methods is illustrated in Figure 10. At the edge a special ER is introduced. Different indication methods are proceeded be means of additional components.

The EAT in co-operation with the ACA will be able to handle header-based indicators or simple RSVP requests. Requests using the PATH/RESV mechanism are terminated and generated locally

within the access domain by the EAT within the AQUILA architecture.

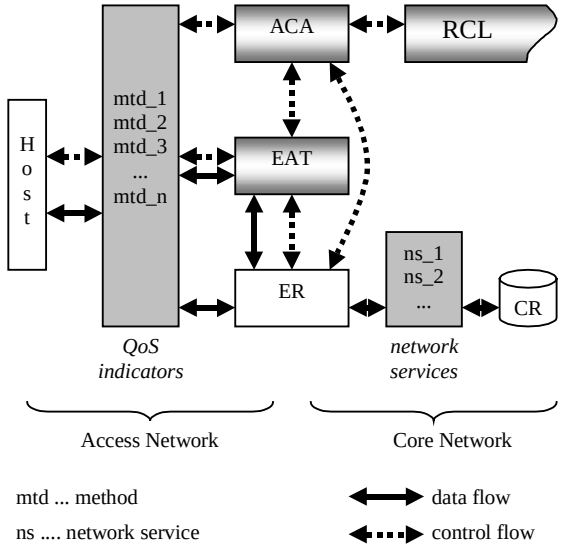


Figure 10. AQUILA Architecture.

Core Routers act autonomously based on their knowledge about network services. The RCL takes care of available resources.

Other protocols can be easily integrated in the concept as a new plug-in. Typically, a useful subset of such protocols will be exploited which allows the best mapping to service classes.

5.3 Merging of Methods

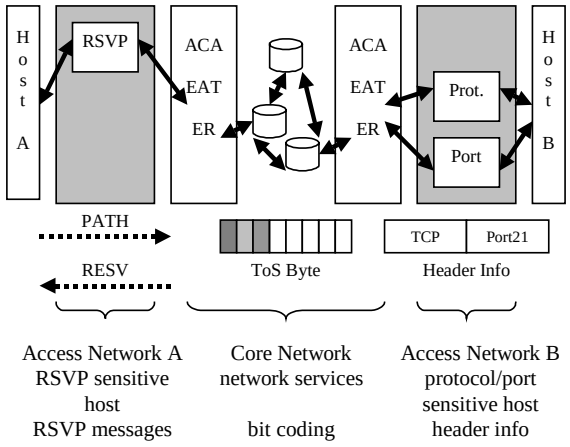


Figure 11. Mixed QoS Indication Handling.

Independently of the chosen method, the indication message will be terminated or generated locally within the access domain, using the infrastructure of the resource control layer (ACA, EAT). This approach implies the chance of inter-communication between hosts, using different QoS indication methods.

In the example, given in Figure 11, the RSVP aware host A shares an application with the protocol type and port sensitive host B.

6 SUMMARY AND TRENDS

The proposed architecture allows a flexible indication of IP packets at the access and classification of IP traffic to be mapped onto a limited set of network services by the interpretation of indicators. No extra protocol per session is needed. Admission control and profiling avoid traffic congestion in the core network.

As the internet grows uncontrollably and works connectionlessly the general introduction of a single signalling protocol is unlikely. Light solutions which cover many existing QoS access mechanisms and provide them with access to a single common core network are one possible solution to fill this gap.

The proposed solution does not touch the core network, except for the fact that QoS aware routers are expected. This offers the chance to generate distributed virtual private networks on top of large provider's networks.

LIST OF ABREVIATIONS

ACA	Admission Control Agent
API	Application Programming Interface
AQUILA	Adaptive Resource Control for QoS Using an IP-based Layered Architecture
BoS	Bottom of Stack
CoS	Class of Service
CR	Core Router
DiffServ	Differentiated Services (IETF)
DSCP	Differentiated Services Code Point
EAT	End user Application Toolkit
ER	Edge Router
GUI	Graphical User Interface
IETF	Internet Engineering Task Force
IntServ	Integrated Services (IETF)
IP	Internet Protocol
Ipv4	Internet Protocol version 4
IST	Information Society Technologies
MPLS	Multi-Protocol Label Switching
PMC	Premium Mission Critical service
PMM	Premium Multi-Media Service
PVO	Premium Voice service
QoS	Quality of Service
RCL	Resource Control Layer
RSVP	Resource Reservation Protocol
RTP	Real Time Protocol
SLA	Service Level Agreement
STD	Standard service
TCP	Transport Control Protocol
ToS	Type of Service
TTL	Time To Live
UDP	User Datagram Protocol

ACKNOWLEDGMENTS

The AQUILA project IST-1999-10077 [5] is a common research EU project within the IST programme [6]. Manufacturers, ISPs and operators, content provider as well as universities and research institutes came together to work on this 3-year research project in the "Next Generation Networks" cluster.

Many thanks to our project partners for the fruitful co-operation. These are Siemens AG - D, Q-Systems Association – GR as manufacturers; Telekom Austria AG – A, Telekomunikacja Polska S.A. – PL, Elisa Communications – FIN, Deutsche Telekom – D from the ISP and operator side; Bertelsmann mediaSystems – D as content provider and from the universities and research institutes: Dresden University of Technology – D, National Technical University of Athens – GR, Salzburg Research Forschungsgesellschaft m.b.H. – A, Warsaw University of Technology – PL and Consorzio di Ricerca sulle Telecomunicazioni – I.

REFERENCES

[1] RFC791, "Internet Protocol - DARPA Internet Program Protocol", September 1981

[2] RFC2474, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers ", Dezember 1998

[3] IETF DiffServ Working Group, URL: http://www.ietf.org/html.charters/diffserv-charter.html, April 2000.

[4] IETF IntServ Working Group, URL: http://www.ietf.org/html.charters/intserv-charter.html, July 2000.

[5] AQUILA project consortium, "Adaptive Resource Control for QoS Using an IP-based Layered Architecture", AQUILA homepage, URL: http://www-st.inf.tu-dresden.de/aquila/, Dresden, September 2000.

[6] The European Commission, "Information Society Technologies Programme (IST)", IST homepage, URL: http://www.cordis.lu/ist/, Brussels, May 1999.

[7] M. Winter et. al., "System architecture and specification for the first trial", AQUILA deliverable D1201, Munich, June 2000.