Functional dependencies and points of intersection between the mechanisms for providing guaranteed QoS

Vanya Topalova

Abstract: *Contemporary technology progress increase the requirements towards the communicational environment – performance, opportunities for transfer of multimedia streams in a real time, reliability etc. Two leading end goals emerge – scalability and reliability of the network solutions, according to which the data traffic in the computer networks is designed and developed. Furthermore, increased possibilities for achieving optimized network parameters as delay and bandwidth are requested. Existing architectures reflect the differences in the approaches for the design of the packet-router interaction scheme in order to achieve a guaranteed level of quality of service (QoS).*

This paper discusses the task for defining of the roles and the functional dependencies between the routers and the packets as a common problem in the flow of creation and improvement of QoS architectures.

Keywords: *Routers, Scalability, Quality of Service, QoS Architectures*

Introduction

The emergence of new applications, among which the most popular IP telephony, video conferences, audio and video streams are only small part, leads to requirements exceeding the possibilities of the best-effort service, implemented in the established IP-based network (Table 1). These applications are more critical than their predecessors toward parameters like delay, bandwidth, packet loss etc.

Table 1 Modern applications requirements

Applications	Requirements		Constraints
IP telephony, video conferences, audio and video flows	Standards	H.323 (ITU-T)	High prices, VoIP
		SIP (IETF)	Lower level of standardization
Audio streams transfer (IP telephony)	Human perception	sound	Maximum delay 100ms

The type of existing network architectures, mainly the achievable level of QoS they guarantee, is related to the satisfied part of the declared applied services. When the main purpose is to provide an assured quality of service, there are two core architectures [4] for IP-based computer networks:

- session-based architectures, for instance integrated services
- session independent architectures, such as differentiated services

The former include more flexible and powerful services, orientated toward a separate flow. They also achieve higher level of resource utilization. One of the most important characteristics of the second kind of architecture is the property scalability. However, both architecture types have some common aspects, outlined below, which are discussed in this paper.

First, routers have a key role in service providing. One of the most significant reasons for the importance of these devices in the communicational process is the greater than before need to guarantee a successful service delivery [3]. Another motive is the specificity of the algorithmic and technical realization of the covered network interfaces. Moreover, routing tables and protocols directly impact on the mechanisms for achieving a certain quality and the level of design complexity of the devices.

Second, new architectures development depends to some extent on the correct understanding of the term service and its implementation for a concrete applied scenario. Every service is characterized with a set of qualitative and quantitative parameters. The quantitative ones involve:

- reliability level, regarding the end service delivery and

- isolation, which describes the network ability to protect a separate flow against malicious attacks that can lead to network overloading.

Quantitative parameters (bandwidth, delay, jitter, packet loss and so on) can also be used to define in large sense the frames of the qualitative parameters between two end hosts.

The pointed out set of general features allows proposing classifications and comparisons in different axis, as the one examined below.

Existing QoS architecture models

In this section we propose an extended classification (Table 2) of the one presented in [3] for a packet switching network environment. Furthermore, we use this taxonomy to describe the traditional best effort service and the recently proposed services, leading to enhancement of today's Internet. Services are classified along one main property: the architecture model providing a predefined QoS. We then compare the existing solutions to implement these services, including the two relatively new QoS architecture models - Dynamic Packet State and Adaptive services.

Table 2 Network services taxonomy

Service model/type		Architecture granularity	Quality of service
Best-effort service		Packet	Connectivity, no delivery guarantees
Flow protection		Flow	Flow isolation
Integrated services (IntServ)	Guaranteed	Flow	Bandwidth and delay guarantees
	Controlled-Load	Flow	Lower bandwidth guarantees
Differentiated services (DiffServ)	Premium (High quality)	Macroflow	Bandwidth guarantees
	Assured (Base service)	Aggregate tragic from many sources/destinations	Lower bandwidth guarantees
Dynamic Packet State (DPS)		Flow	Bandwidth and delay guarantees
Adaptive services (A-Serv)	High quality	Flow	Bandwidth and delay guarantees
	Base service	Aggregate tragic from many sources/destinations	Lower bandwidth guarantees

Best-effort service is defined as a connectivity service which allows any two hosts in the Internet to communicate by exchanging packets, but it does not make any promise of whether a packet is actually delivered to the destination, or whether the packets are delivered in order or not. It requires little support from routers. Among the most significant properties of this architecture are:

- high level of scalability
- high level of performance
- robustness – potential communication in case of router and link failures, and/or network reconfiguration

Flow-protection model provides mechanisms to control congestion. The current Internet relies on end-to-end congestion control mechanisms in which senders reduce their transmission rates whenever they detect congestion in the network. As main drawbacks can be mentioned:

- implementation of equivalent congestion control algorithms applied at the end hosts
- restricted choice of architecture solutions – use of TCP-friendly protocols, compatible with the congestion algorithm of the TCP protocol.

As new applications such as IP telephony, video-conferencing, and audio and video streaming are deployed in the Internet, services more sophisticated than best effort are needed. The TCP protocol influence in designing QoS architectures decreases. As most perspective are accepted the connectionless oriented protocols, since they allow achievement of optimized qualitative and quantitative network parameters.

IntServ model [2] submits an example of a session-based architecture. It offers two services according to the customer needs:

- guaranteed service for applications requiring a fixed delay bound;
- controlled-load service – for applications requiring reliable and enhanced best-effort service;

To alleviate the scalability problems of the IntServ model is proposed the architecture model DiffServ [8]. Furthermore, this architecture also has two possible implementations:

- premium (high quality service) – bandwidth guarantees and use of centralized database (named Bandwidth Broker) in the process of resource scheduling between two end hosts (routers)
- assured (base service) - a large granularity service, which is associated with the aggregate traffic of a customer from/to multiple hosts.

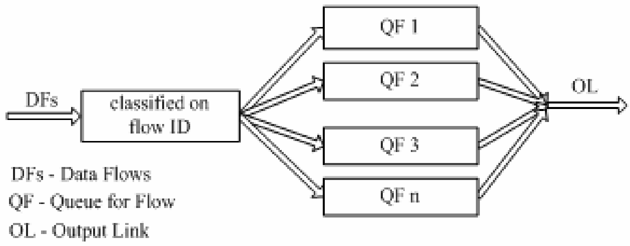
As a common characteristic between DiffServ and the recently proposed service models DPS [4] and A-Serv can be defined the higher percentage of scalability, because of the realization of stateless core architecture. Also, backbone routers have a major role in the process of packets treatment. DPS model has the packets to carry flow state information instead of having the routers to maintain it. Both architectures, DPS and A-Serv, allow per-flow treatment, but the second model behaviour depends on the load burden of the router. Each data flow can be treated either as per flow or aggregate traffic in the core routers according to the core routers' load burden. It allows being defined two services, based on the way of processing of the packets – a base service with use of aggregation of traffic and a high quality service, with an opportunity for separate flow treatment.

Next section focus on the movement in the development and implementation of the basic mechanisms and functions providing QoS guarantees in some of the most significant leading QoS architecture models.

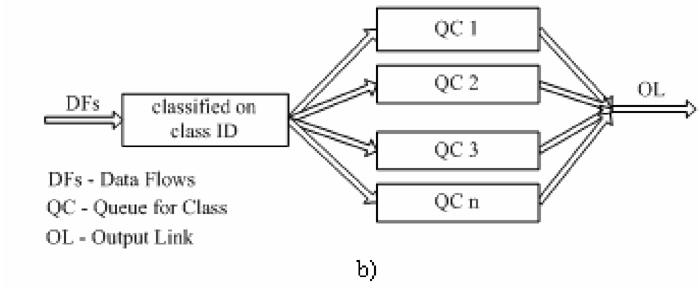
Points of intersection between the QoS architectures

According to one of the common definitions of QoS architectures, they represent mechanisms and functions to ensure that the service guarantees are indeed enforced. As a result, the recently proposed QoS architectures submitted in the previous section will allow us to trace the advance of the packet processing logic, respectively scalability level and network parameters, qualitative and quantitative and to outline the main points of intersection.

The ability to process packets on a per flow basis is important because of the opportunities for simultaneous support of applications with different performance requirements, and high available level of resources utilization [4]. With such an approach the flows critical to a definite metric, can receive per flow treatment, not management as an aggregate traffic. Fig. 1 gives an overview of data flow treatment at a set of QoS models.



a)



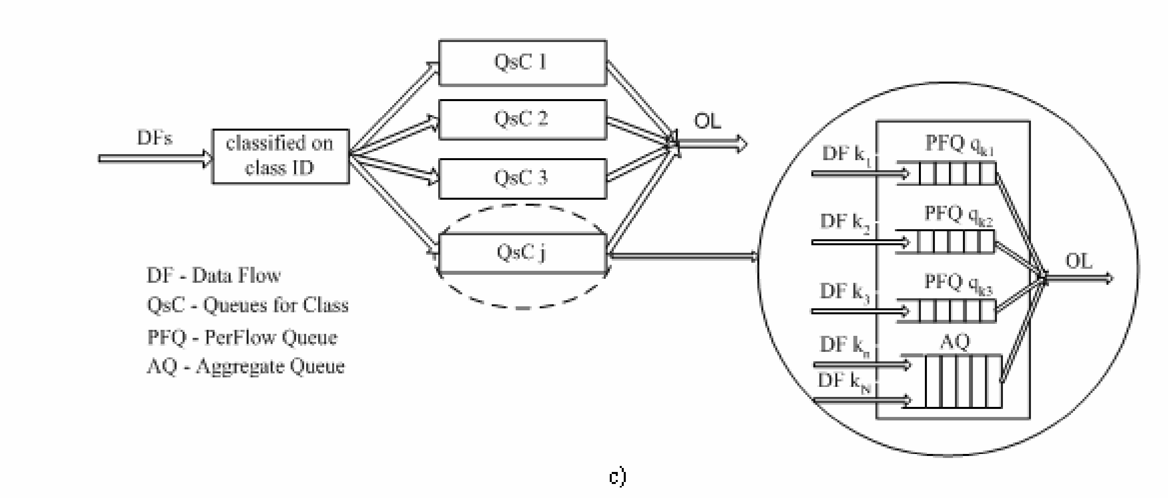


Fig. 1. a) Data Flow treatment in Integrated services
b) Data Flow treatment in Differentiated services
c) Data Flow treatment in Adaptive services

Although datagram protocols process the traffic as quickly as possible, they give no guarantee to the quality of service. Every data flow consists of packets, having one and the same 5-tuple flow identifier. In most of the cases it consists of 5 parts – source IP address, destination IP address, transport protocol, source port, and destination port.

Compared with IntServ (Fig. 1a), A-Serv architecture doesn't have the problem of scalability at core routers. When the number of data flow exceeds the processing ability or storage space of the load adaptive router, the rest data flows will be treated as aggregate traffic. Thus, the burden of state information storage and scheduling complexity in routers won't increase.

One of the features of the A-Serv includes maintenance of as much data flow state information as possible at core routers, which is different from core stateless DPS architecture. In A-Serv (Fig. 1c), each data flow can be treated either as per flow or aggregate traffic in the core routers according to the core routers' load burden. A-Serv doesn't have scalability problem in IntServ, and it can provide better service guarantee to individual data flow than DiffServ. In addition, A-Serv can be deployed gradually on existing QoS architectures. Simulation results from [11] show that A-Serv can provide differentiated service to the data flows in the same DiffServ class and solve the scalability problem in the core routers. First, the load adaptive routers with light load burden allow a data flow to be treated as per flow, whereas the load adaptive routers with heavy load burden treat separated data flows as aggregate traffic in most of the cases. Also, in the second case the type of traffic has no requirements for storage of flow state information or more complex traffic management scheme.

Another important feature of the QoS mechanisms development is the fact that DiffServ architecture treats all data flows as aggregate traffic, as shown on Fig. 1b, whereas the load adaptive routers, used in A-Serv, are always trying to fully utilize their processing ability to guarantee the QoS to as many data flows as possible by treating them as per flows. Furthermore, when the malicious data flow exists, it will affect the service received by all the data flows in the same class in DiffServ. In A-Serv, the per-flow treated data flows can be protected from being affected by the malicious data flow.

Among the main advantages, related to the most recent service model A-Serv is the possibility for its gradual deployment in existing IntServ or DiffServ network. First, the bottleneck core routers in IntServ domain can be replaced with load adaptive routers in order to deal with IntServ's scalability problem. Thus, the data flows can receive per-flow treatments in all the IntServ core routers and adaptive services in the load adaptive routers. Next, core routers can be replaced in DiffServ domain and the unchanged DiffServ core routers can be assumed as load adaptive routers with limited processing ability (one queue per each DiffServ class). The most important prerequisites, which impose the deployment of complex adaptive architecture mechanisms, as far as the main specific functional dependencies of the existing service models are summarized below.

Routers and packet processing logic dependencies in QoS architectures

QoS architectures represent solutions with common aims, but different implementations. In this section is discussed the functional description of some models of architectures providing QoS guarantees, their positive and negative sides and also the implementation restrictions. Moreover, the innovative flexible architecture model A-Serv is built on the basis of them.

IntServ architecture is characterized by resource reservation for each data flow through Resource Reservation Protocol (RSVP) signaling protocol [7]. All routers, including edge routers and core routers, keep the per flow state information and allocate resources (such as buffer space and link bandwidth) to each data flow. Packets are identified by the flow ID (5-tuple with 104 bits in IPv4 and 296 bits in IPv6) and guaranteed services can be provided to each individual data flow. The major scalability problem of IntServ is that the amount of data flow state information increases proportionally with the number of data flows.

In DiffServ, packets are marked in DS field (6 bits differentiated services code point defined in IPv4 and IPv6 packet header) with different DiffServ class IDs to create 2-8 DiffServ classes. Packets are classified and assigned DiffServ class ID at the ingress edge router of a DiffServ capable domain. Subsequent packet classification and forwarding in the core routers are based on the DiffServ class ID in the header of every packet. The data flows in the same DiffServ class are treated as aggregate traffic in the core routers. Data packets with different class IDs receive different services (e.g. Expedited Forwarding (EF) [10] and Assured Forwarding (AF) [6]). DiffServ is scalable in the core routers for the limited number of DiffServ classes, which bounds the amount of state information maintained by each core router. One problem of DiffServ is that service received by data flows in aggregate traffic can be affected by the other data flows in the same DiffServ class and thus individual data flow's QoS cannot be guaranteed even with EF treatment [1, 5]. Thus the scalability achieved by DiffServ is at the expense of reduced performance [9].

DPS architecture [4] has the packets to carry flow state information instead of having the routers to maintain it. The ingress edge router inserts the data flow state information into the header of each packet (17 bits defined in DPS architecture). The core routers process each packet according to the data flow state information and the routers' internal state information, which does not increase with the number of data flows. Before forwarding a packet, the core routers update the flow state information in the packet's header and the internal state information in routers. By this means, DPS provides scalable QoS services with improved performance than DiffServ [9]. DPS requires special scheduling scheme to be installed in each core router to perform scheduling operation and modification in each packet's header. Therefore, deploying DPS requires changing all the edge and core routers in one domain and the gradual deployment is not achievable.

In summary, backward compatibility is among the nowadays conditions for providing assured network service. They also include higher level of service guarantees corresponding to the application demands and a flexible packet-router interface scheme, involving adaptive implementation towards the network environment.

Conclusions

This paper deals with the functional dependencies between the router and packet characteristics as a common problem in the course of creation and implementation of QoS architectures. It summarizes some special functional features of leading QoS network architectures, concerning mainly the router functions and the packet processing logic. In addition, their positive and negative aspects are defined. As a result, it points out the main guidelines in the network service providing and shows that the most important goals in the contemporary packet switching environment do not remain fixed. Although network scalability and higher level of network parameters are preserved, a new approach for designing of a packet-router interaction scheme is shaped. The future development directions include building of a flexible traffic management scheme, which could offer an adaptive to the load burden of the routers service as well as an opportunity for its gradual deployment in the existing computer networks.

References

[1] A. Charny and J.Y. Le Boudec. Delay bounds in a network with aggregate scheduling. Proc. Of First International Workshop of QofIS'2000, Berlin, Germany, September, 2000.

[2] B. Braden. Integrated services in Internet architecture - an overview. Internet RFC-1633, June 1994.

[3] I. Stoica. Stateless Core: A scalable approach for quality of service in the Internet. Thesis, Carnegie Mellon University, 2000.

[4] I. Stoica and H. Zhang. Providing guaranteed services without per flow management. Proc. Of ACM SIGCOMM, pp. 81-94, September 1999.

[5] J.C.R. Bennett, K. Benson, A. Charny, W.F. Courtney, and J.Y. Le Boudec. Delay jitter bounds and packet scale rate guarantee for expedited forwarding. IEEE/ACM Transactions on Networking, vol. 10, Issue 4, pp. 529-540, August 2002.

[6] J. Heinanen, F. Baker, W. Weiss, and J. Wroclawski. Assured forwarding PHB group. RFC 2597, Internet Engineering task Force, June 1999.

[7] L. Zhang, S. Deering, D. Estrin, S. Shenker, and D. Zappala. RSVP: A new resource reservation protocol. IEEE Communications Magazine, May 2002.

[8] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss. An architecture for differential services. IETF, RFC 2475, December 1998.

[9] V. Firoiu, J. -Y. Le Boudec, D. Towsley, Zhi-Li Zhang. Theories and models for Internet quality of service. Proc. Of the IEEE, vol. 90, Issue 9, pp. 1565-1591, September 2002.

[10] V. Jacobson, K. Nichols, and K. Poduri. An expedited forwarding PHB. RFC 2598, Internet Engineering Task Force, June 1999.

[11] Y. Wang, A. Krishnamurthy, L. Qia, P. Dauchy, and A. Conte. A-Serv: A Novel Architecture Providing Scalable Quality of Service. GLOBECOM 2004 - IEEE Global Telecommunications Conference, no. 1, 2004.

ABOUT THE AUTHOR

Vanya Topalova, PhD Student, Department of Computer Systems and Technologies, Technical University of Gabrovo, Phone: +359 889 118 309, E-mail: vtopalova@tugab.bg.