Encyclopedia of Internet Technologies and Applications

Mario Freire
University of Beira Interior, Portugal

Manuela Pereira
University of Beira Interior, Portugal

Information Science
REFERENCE

INFORMATION SCIENCE REFERENCE
Hershey · New York

Acquisitions Editor:	Kristin Klinger
Development Editor:	Kristin Roth
Senior Managing Editor:	Jennifer Neidig
Managing Editor:	Sara Reed
Copy Editor:	Larissa Vinci and Mike Goldberg
Typesetter:	Amanda Appicello and Jeffrey Ash
Cover Design:	Lisa Tosheff
Printed at:	Yurchak Printing Inc.

Published in the United States of America by
Information Science Reference (an imprint of IGI Global)
701 E. Chocolate Avenue, Suite 200
Hershey PA 17033
Tel: 717-533-8845
Fax: 717-533-8661
E-mail: cust@igi-global.com
Web site: http://www.igi-global.com/reference

and in the United Kingdom by
Information Science Reference (an imprint of IGI Global)
3 Henrietta Street
Covent Garden
London WC2E 8LU
Tel: 44 20 7240 0856
Fax: 44 20 7379 0609
Web site: http://www.eurospanonline.com

Library of Congress Cataloging-in-Publication Data

Encyclopedia of Internet technologies and applications / Mario Freire and Manuela Pereira, editors.
p. cm.
Summary: "This book is the single source for information on the world's greatest network, and provides a wealth of information for the average Internet consumer, as well as for experts in the field of networking and Internet technologies. It provides the most thorough examination of Internet technologies and applications for researchers in a variety of related fields"--Provided by publisher.
Includes bibliographical references and index.
ISBN 978-1-59140-993-9 (hardcover) -- ISBN 978-1-59140-994-6 (ebook)
1. Internet--Encyclopedias. I. Freire, Mário Marques, 1969- II. Pereira, Manuela.
TK5105.875.I57E476 2007
004.67'803--dc22
2007024949

British Cataloguing in Publication Data
A Cataloguing in Publication record for this book is available from the British Library.

All work contributed to this encyclopedia set is original material. The views expressed in this encyclopedia set are those of the authors, but not necessarily of the publisher.

Quality of Service Architectures

Christos Bouras
Research Academic Computer Technology Institute and University of Patras, Greece

Apostolos Gkamas
Research Academic Computer Technology Institute and University of Patras, Greece

Dimitris Primpas
Research Academic Computer Technology Institute and University of Patras, Greece

Kostas Stamos
Research Academic Computer Technology Institute and University of Patras, Greece

INTRODUCTION

IP networks are built around the idea of best effort networking, which makes no guarantees regarding the delivery, speed, and accuracy of the transmitted data. While this model is suitable for a large number of applications, and works well for almost all applications when the network load is low (and therefore there is no congestion), there are two main factors that lead to the need for an additional capability of quality of service guarantees. One is the fact that an increasing number of Internet applications are related to real-time and other multimedia data, which have greater service requirements in order to be satisfying to the user. The other is that Internet usage is steadily increasing, and although the network infrastructure is also updated often, it is not always certain that network resource offerings will be ahead of usage demand. In order to deal with this situation, IETF has developed two architectures in order to enable QoS-based handling of data flows in IP networks. This article describes and compares these two architectures.

BACKGROUND

The two main architectures that have been proposed for **quality of service** are IntServ and DiffServ. They follow different philosophies as they approach the topic of Quality of Service from different point of views.

The IntServ architecture tries to provide absolute guarantees via resource reservations across the paths that the traffic class follows. The main protocol that works with this architecture is the Reservation Protocol (**RSVP**). However, its operation is quite complicated and it also contributes significant network overhead. On the other hand, DiffServ architecture is more flexible and efficient as it tries to provide Quality of Service via a different approach. It classifies all the network traffic into classes and tries to treat each class differently, according to the level of QoS guarantees that each class needs. In the DiffServ architecture, two different types (per hop behaviours in Nichols, 2001) have been proposed, the expedited forwarding (Jacobson et al., 1999) and the assured forwarding (Heinanen et al., 1999); their difference is on the packet forwarding behaviour. **expedited forwarding** (EF) aims at providing QoS for the class by minimizing the jitter and is generally focused on providing stricter guarantees. This type tries to simulate the virtual leased lines and its policy profile should be very tight. **Assured forwarding** (AF) inserts at most four classes with at most three levels of dropping packets. Every time the traffic of each class exceeds the policy criteria, then it is marked as a lower level QoS class.

MAIN QoS ARCHITECTURES

Integrated Services (IntServ)

Integrated Services (IntServ) makes use of the RSVP protocol in order to make reservations for resources across the network. It has been initially developed by IETF in order to extend the traditional “best effort” model that has been used on the Internet. Its basic idea is that it should not be necessary to modify the underlying architecture of the Internet, but simply to add some extensions that can offer additional services beyond the basic "best-effort" service.

Quality of service (QoS) in the IntServ framework refers to the nature of the service being offered by the network, characterized by parameters such as the available bandwidth, packet delay, and packet loss. A network node in the IntServ architecture has the capability to handle packets and subject them to appropriate control. An IntServ-capable node can offer one or more of the IntServ services, while an IntServ-aware node supports the interfaces needed by the IntServ model but cannot offer the required service itself. An IntServ-aware node can simply understand the parameters of the required service and answer negatively.

Resource management is an important aspect of the IntServ architecture, and therefore traffic is subjected to admission control mechanisms. Furthermore, IntServ is responsible for reserving the resources. For that purpose, the RSVP protocol (Resource Reservation Protocol) is used, which aims at specifying the necessary resources for achieving the required quality of service. RSVP (Braden et al., 1997) reserves resources across the whole path used by the packets in a sequential manner. The first router in the path signals to the next router in the path that a resource reservation is required. This process is repeated until the receiving node is reached, and then the same procedure begins in the opposite. The IntServ services that have been currently defined are the "Guaranteed" service, which is the closest service to the dedicated virtual circuits, and the Controlled Load service, which is equivalent to the best-effort service under no congestion.

Differentiated Services (DiffServ)

Differentiated Services (DiffServ) (Blake, 2001) classifies and prioritizes packets depending on the class they belong to. Classes with larger requirements are treated preferentially by the network that supports DiffServ.

DiffServ is the second important effort for providing Quality of Service and was developed in order to overcome some of the disadvantages of IntServ. In particular, IntServ proved to be non-scalable in large networks where a lot of resource reservations are required. DiffServ operation is based on the usage of a field in the IP header called DS, which is contained in the **Type Of Service** (TOS) field in the IPv4 header, and the Traffic Class field in the IPv6 header (Nichols, 2001). Clients that want to make use of the DiffServ architecture mark the DS field with a specific value. This value specifies the Per-Hop Behavior (PHB) for the client's packets. The possible DS values have to be agreed between the provider and the client in the form of a **service level agreement** (SLA) and they determine the quality of service parameters such as bandwidth, transmission, and rejection priority and queue priority.

DiffServ is a unidirectional and therefore non-symmetrical model. It can also be only used for unicast transmission.

Currently, the following two types of DiffServ services have been proposed:

- **Expedited forwarding (EF):** This service aim at minimizing packet delay and jitter, while providing highest quality of service. Packets that exceed the mutually agreed packet profile of the user are generally rejected. Services of this type emulate the operation of a virtual leased line (Jacobson et al., 1999).
- **Assured forwarding (AF):** This type provides at most four classes of service and at most three levels of rejection per class. AF traffic that exceeds the agreed profile is degraded but not necessarily rejected (Heinanen et al., 1999).

DiffServ operation is based on a number of mechanisms that operate on the traffic flows. These mechanisms are packet classification, marking, metering, and shaping, which are typically applied with this order, although traffic metering can precede marking. The mechanisms only need to be applied at the edge routers of a domain, while no application of the above mechanisms is needed for the core routers of the network. This feature of DiffServ overcomes the scalability problem of IntServ, since the core routers that handle a large number of flows do not have to apply the above mechanisms on these flows.

Packet Classification

Packet classification is the first step in the provisioning of quality of service. Classification of packets entering a network that supports QoS can be done either at a level of flows, or at a level of aggregate flows. This process mainly takes place by checking the header of each packet and using information from some field in order to make the classification. The relevant field is Type of Service (TOS) in IPv4, and the Traffic Class field in IPv6. The classification mechanism has to be very fast in order to be able to follow the rate of incoming packets, and very accurate.

Theoretically flows can be characterized by the following five elements:

- Sender IP address
- Sender port
- Destination IP address
- Destination port
- Protocol used

Classification per flow using these characteristics is called multifield classification and is quite difficult because checking so many fields requires a lot of processing power. Multifield classification is only used when classification based on individual flows is absolutely necessary (which is not a rare occurrence in the input points of DiffServ domains).

On the other hand, classification based on aggregates of flows is called behavior aggregate classification and only a combination of the above characteristics is needed. This classification method is easier and can be performed much faster.

Practically, classification will be done in a limited number of classes, so a single field in the packet header is enough. This is the simplest and most efficient method, and achieves classification at the level of aggregate flows.

Traffic Conditioning

Traffic conditioning includes the marking, metering, and shaping or traffic rejection mechanisms. Usually these mechanisms are applied to the sender's packets as soon as the traffic enters a domain. Nevertheless, the metering mechanism can be applied to the destination, under certain conditions. For this to be possible, all the routers of the network have to support the ECN (Explicit Congestion Notification), which is a congestion control functionality. ECN is a bit at the packets header which is set to 1 when congestion is detected at the network. This enables the rest of the nodes on the path to be notified of congestion.

The traffic control mechanisms presented below are based on the assumption that marking and metering of packets takes place at the entry points of the network.

Policing

Policing also takes place at the traffic entry points of a DiffServ domain. The policing mechanism controls traffic based on a specified profile that has been agreed upon and then makes certain decisions for handling traffic that exceeds the specified profile. These decisions can be marking the packets at a lower class of service, servicing the packets without guaranteed quality, or even dropping the packets. Which of these policies will be followed has already been agreed upon between the client and the administrator of the network and has been formulated in the form of a service level agreement (SLA). The policing criteria used can depend on the time or day, the source, the destination, or generally any other characteristic of the traffic.

The shaping mechanism aims at shaping the traffic in such a way that bursts (sudden transmission of many packets) are smoothed out and can be configured so that packets out of the specified profile (that would normally be dropped) are temporarily stored and forwarded to the network as soon as the burstiness of their transmission has been eliminated. Therefore, policing and shaping mechanisms can be used simultaneously so that part of the packets that are considered out of profile to be shaped and transmitted.

Queue Management

Queue management is important for the network administrator in order to be able to provide quality of service to the flows as has been agreed. Furthermore, queue management is a basic condition for the time scheduling mechanism that will be presented in the next section. In order for the network to satisfy all quality of service guarantees, it has to be able to handle packets of each class of service at a separate queue, so that the suitable time scheduling mechanism can be applied. Otherwise, the time scheduling mechanism is not able to differentiate between the classes and cannot offer the proper guarantees to the corresponding traffic flows. More specifically, if, for example, no differentiation of classes into separate queues takes place, different flows with different requirements will be accumulated at the same queue, and then packets will either be dropped or delivered with large delay. As a result, the network will not be able to provide the required guarantees and the throughput experienced by the client applications will be significantly downgraded.

The main functions of queue management aim at the proper queue operation and the usage of mechanisms for their control. They are the following:

- Add a packet at the proper queue according to the packet classification by the classification mechanism.
- Reject a packet if the queue that the packet should be added is full.
- Withdraw a packet from the top of the queue when the time scheduler requests so, in order for the packet to be transmitted to the next network node.
- Check the state of the queue. This includes checking the average size of the queue and taking actions in order to keep its size small. Possible actions are the rejection of a packet if the queue is starting to fill and the marking of a packet with the ECN bit when the queue size is large.

As the above functions suggest, queue management does not only deal with the reception and transmission of a packet, but it is also concerned with the efficient operation of the queue through the preservation of small average queue size. By keeping the average queue size small, the queues can easily absorb traffic bursts. If the average queue size gets large, then many packets have to be dropped during traffic bursts. Furthermore, another desired result of the small average queue size is that the average service delay will be small.

Queue management becomes even more critical under network congestion, when queues have to operate quickly and correctly. The main problem in this case is to identify the most appropriate strategies for action. A critical decision is whether packets will be dropped as soon as they reach the queue, or whether it is allowed to drop packets that are already inside the queue, in order to service other, higher priority packets. Another critical aspect is the criteria and information that will determine which packets should be dropped.

The general purpose of the queue management is to handle queues fairly for all classes of service while adhering to the agreements that have been made with the network clients. In order to avoid network congestion (which leads to increased average queue size), TCP at the transport layer offers several mechanisms. In addition, network administrators have a number of further options, which will be described below. Their purpose is to ease the congestion problem that stems from the transmission of packets with a higher rate than the network can handle, and not congestion that stems from temporary bursts. These mechanisms that can be used by the network administrator are:

- **Dropping packets:** This method achieves two objectives, by directly reducing the network load and also informing the TCP protocol of the congestion condition. This is due to the fact that TCP congestion control mechanism relies on the assumption that each packet loss is due to congestion and therefore the transmission rate is automatically reduced.
- **Packet marking:** This method is less intrusive than the previous one, since it does not directly drop packets, but is also less direct, since the network is not automatically relieved.

Time Scheduler

Time scheduling is the way that the network handles the queues, meaning which queue will send data and for how long. The time scheduling mechanism has all the queues of a router available and decides in what order they are going to transmit packets and for how long.

The role of the time scheduler is critical for a network that wishes to offer quality of service guarantees. The reason is that the time scheduling mechanism determines the delay at each queue and the way that the line is shared between the queues. The time scheduling mechanism actually determines the type of quality of service that will be provided by the network. The parameters that can be affected by the time scheduling mechanism are:

- The throughput of each flow, since the time scheduler can control the intervals that this flow will transmit data.
- The delay of each flow, since the time scheduler controls the transmission rate of the flow and therefore the duration that the packets remain in the queue.
- The jitter.

These parameters determine the quality of service that the network can provide. Therefore, because of the importance of the time scheduler, and because of the fact that there are several time scheduling mechanisms, it is necessary that the proper time scheduling mechanism is chosen according to several criteria. The kind of quality guarantees offered and their level of success should match the nature of applications that will be supported.

Some of the most widely used mechanisms for time scheduling are:

- **First in first out (FIFO):** This is the oldest mechanism, and it assumes there is only one queue. Every packet exits the queue in the order it arrived. As a result, FIFO uses no priorities. It is a simple mechanism, and it is useful for high capacity lines where there is no congestion. On the other hand, it performs badly when there is congestion, or when bursty applications dominate the queue and other applications' packets are rejected.
- **Priority queuing (PQ):** Priority queuing allows different priorities and can handle multiple queues. One queue has strict priority and is always preferentially served. Packets are inserted in the proper queue depending on their classification. The priority queuing mechanism checks the queues sequentially by starting from the highest priority queue, until a non-empty queue is found. The first packet from that queue is then transmitted, and the procedure starts over. Depending on the incoming rate for high priority packets, other queues might be served very slowly or not at all. The latter might occur if high priority traffic arrives at a rate close to or higher than the link capacity, and can be remedied by applying policing or shaping mechanisms to high priority traffic. The main advantage of Priority Queuing is that it achieves very low delay for high priority packets.
- **Modified deficit round robin (M-DRR):** M-DRR is based on deficit round robin (DRR) and round robin (RR) mechanisms. Round robin handles all queues equally and checks them periodically. It transmits any packets that are waiting in a queue, and continues checking other queues. DRR functions similarly, but now queues try to maintain a steady transmission rate. This is achieved by defining for each queue a quantum Q and a deficit D. Q is the maximum number of bytes that can be transmitted each time. If less bytes are transmitted, the remaining number is stored at D, which increases the maximum number of bytes that can be transmitted next time. M-DRR also introduces a priority queue for achieving low delay. The rest of the queues are served according to the DRR mechanism, and the priority queue is served either alternately with the rest of the queues or in absolute priority. These two variations of M-DRR are correspondingly called Alternate Priority and Strict Priority. M-DRR is flexible and efficient, but only under conditions with not too much congestion. Therefore, it is usually used in conjunction with a mechanism that prevents congestion.
- **Fair queuing (FQ) and weighted fair queuing (WFQ):** Fair queuing is a variation of round robin, with the added goal of serving all queues for a long term equal bandwidth sharing. It schedules packet transmissions in the order that they would have arrived at the other end of the line if an ideal time scheduling mechanism had been used. Its disadvantages are that the computations to achieve such a goal are complex and have to be performed approximately, and that it cannot operate with the aggregate classes model. Weighted Fair Queuing on the other hand, assigns weights to each queue. In case some queues are empty, the excess bandwidth that would be used by these queues is shared among the rest of the queues according to their weights.

FUTURE TRENDS

Nowadays, DiffServ is the most widely used architecture (Grossman, 2002). Its main advantage is its better scalability, since the core network devices only deal with the bulk of flows and not individual flows and reservations. Therefore, core routers are not involved in the complexities of enforcing agreements or collecting payments.

In the framework of the DiffServ architecture, IETF has defined the entity of bandwidth broker (Nichols, et al., 1999). The bandwidth broker is an agent that has some knowledge of an organization's priorities and policies and allocates bandwidth with respect to those policies. In order to achieve an end-to-end allocation of resources across separate domains, the bandwidth broker managing a domain will have to communicate with its adjacent peers, which allows end-to-end services to be constructed out of purely bilateral agreements. bandwidth brokers can be configured with organizational policies, keep track of the current allocation of marked traffic, and interpret new requests to mark traffic in light of the policies and current allocation. Bandwidth brokers only need to establish relationships of limited trust with their peers in adjacent domains, unlike schemes that require the setting of flow specifications in routers throughout an end-to-end path.

The main disadvantage of DiffServ and the architectures for providing quality of service at the network and transport layer is that it is often regarded as a technical solution for a problem (scarcity of network resources) that can be more simply solved by increasing the network's capacity.

Furthermore, the field of policing and shaping mechanisms is still open and various mechanisms can be presented that combine both.

CONCLUSION

While IntServ was the first main architecture for Quality of Service proposed by IETF, its drawbacks led to the development of DiffServ, which has largely substituted IntServ as the main standardized architecture for guaranteed services in IP environments. DiffServ is widely supported in networking equipment and software, and has been implemented, tested, and used in real world environments around the world. It is an important part of the network architecture that has to deal with the real-time and high bandwidth requirements of popular Internet applications.

REFERENCES

Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., & Weiss, W. (1998). *An architecture for differentiated service.* RFC 2475.

Black, D., Brim, S., Carpenter, B., & Le Faucheur, F. (June 2001). *Per hop behavior identification codes.* RFC 3140.

Braden, R., Zhang, L., Berson, S., Herzog, S., & Jamin, S. (1997). *Resource ReSerVation protocol (RSVP).* RFC 2205.

Crawley, E., Nair, R., Rajagopalan, B., & Sandick, H. (August 1998). *A framework for QoS-based routing in the Internet*. RFC 2386.

Grossman, D. (April 2002). *New terminology and clarifications for Diffserv.* RFC 3260.

Heinanen, J., Baker, F., Weiss, W., & Wroclawski, J. (June 1999). *Assured forwarding PHB group.* RFC 2597.

IETF DiffServ working group. Retrieved 27 July, 2006 from http://www.ietf.org/html.charters/OLD/diffserv-charter.html

IETF IntServ working group. Retrieved 27 July, 2006 from http://www3.ietf.org/proceedings/96mar/area.and.wg.reports/tsv/intserv/intserv.html

Internet2 QoS Working Group. Retrieved 27 July, 2006 from http://qos.internet2.edu/wg/

Jacobson, V., Nichols, K., & Poduri, K. (1999). *An expedited forwarding PHB.* RFC 2598.

Nichols, K., Blake S., Baker F., & Black D. (1998). Definition of the differentiated Services. Field (DS Field) in the IPv4 and IPv6 Headers. RFC 2474.

Nichols, K., & Carpenter, B. (April 2001). *Definition of differentiated services per domain behaviors and rules for their specification.* RFC 3086.

Nichols, K., Jackobson, V., & Zhang, L. (July 1999). *A two-bit differentiated services architecture for the Internet.* RFC 2638.

Shenker, S., Partridge, C., & Guerin, R. (1997). *Specification of guaranteed quality of service.* RFC 2212.

Snir Y., Ramberg ,Y., Strassner, J., Cohen, R., & Moore, B. (November 2003). *Policy quality of service (QoS) information model*. RFC 3644. The Internet Engineering Task Force.

Wroclawski, J. (1997). *Specification of the controlled-load network Eeement service.* RFC 2211.

KEY TERMS

Differentiated Services (DiffServ): An architecture that has been defined by IETF in order to provide quality of service in IP networks, which works based on aggregates of flows, by classifying traffic into different types of service, allowing the core routers of the network to deal with only a limited number of aggregated flows.

First-In, First-Out (FIFO): Queue organization method, where each element exits the queue in the order it originally arrived.

Integrated Services (IntServ): An architecture that has been defined by IETF in order to provide Quality of Service in IP networks, which is based on flow-based allocation of resources using RSVP.

Internet Engineering Task Force (IETF): The organization comprised of a large open international community of network designers, operators, vendors, and researchers concerned with the evolution of the Internet architecture and the smooth operation of the Internet.

Per-Hop Behaviour (PHB): The aggregated way packets are forwarded at a differentiated services-compliant node.

Quality of Service (QoS): The ability to provide specific guarantees to traffic flows regarding the network characteristics such as packet loss, delay, and jitter experienced by the flows.

RSVP: Resource Reservation Protocol.