Quality of Service in the Internet

Prashant Bharadwaj

Internet is currently the most popular technology for global communication and information retrieval. Hence, networking technologists involved in Internet research have the responsibility of developing technologies that support multimedia networking applications such as audio and video streaming, video emails, real-time interactive audio and video conferencing and virtual reality. This article discusses the underlying mechanisms that have been developed which augment packet switching in supporting the smooth functioning of these real-time applications.

Prashant Bharadwaj is a Strategic Marketing Manager, Telecom Solutions, Wipro Technologies. During his career in Wipro, he has been involved in developing routing protocol stacks and software solutions for Network Equipment companies like Cisco Systems & Ericsson. His current assignment involves market strategy and positioning of Wipro's telecom services and solutions in the world market.

Introduction to Best-Effort Service

Someone once said that each of us needs to spend time thinking and preparing for the future, because that is where we will be spending most of our time. The explosive growth of the number of computing devices that are connected to the Internet each year has made network researchers re-look at the underlying technologies that form the nuts and bolts of the Internet. Originally the major traffic on the Internet was text-based email and character files. These were transmitted with the help of packet-switched routers and the service offered under the realm of packet switching was the best-effort service. The best effort service meant that Internet protocol would make its 'best effort' to deliver packets between communicating devices, but with no guarantees. It also meant that, packets of all applications would get the same treatment (egalitarian approach!). This service was good enough for transmitting normal data traffic.

Limitations of Best-Effort Service

Keywords

Best-Effort, Quality of Service, Bandwidth, ATM, VOIP, Integrated Services, Resource Reservation protocol, RSVP.

Today, networks will need to transmit more than just text. Video, audio and other continuous media data coupled with graphics have become part of standard desktop computer

The primary requirement of multimedia traffic is that the packets reach their destination within a fixed time frame.

applications. Hence data networks need to be augmented to support real time multimedia traffic. The primary requirement of multimedia traffic is that the packets reach their destination within a fixed time frame. For this integration of data service with voice service to be complete, it is essential to overcome the limitations of best effort service. Packet loss, end-to-end delay and packet jitter constitutes the major limitations of best-effort service. Let us examine these limitations in more detail.

Packet Loss: Consider IP packets of an application moving from source to destination. If the network is heavily loaded with traffic, some of packets may be dropped at the routers due to insufficient storage space in its buffers. Hence these IP packets will fail to arrive at the destination. Readers familiar with the TCP/IP model would argue on this limitation. Their argument would be that if a TCP connection can be established between the source and destination, sender can retransmit those IP packets that have been dropped in the network. However, retransmission mechanisms are not acceptable in case of real time applications. Remember that real-time process is a process, which delivers the results of the processing in a given time-span!

End-to-End Delay: A packet while moving from source to destination may experience many kinds of delay within the network such as: propagation delay in the links, queuing and processing delay in the routers and end-system processing delays. Consider the example of Internet telephony, where voice is sent over IP Networks using Voice over IP (VOIP) protocol. Here delays more than a certain threshold will disturb the smooth flow of an interactive conversation!

Packet loss, end-to-end delay and packet jitter constitutes the major limitations of best-effort service.

Packet Jitter: The transmission of a packet from a source to a destination takes a finite amount of time resulting in some end-to-end delay. A crucial aspect in this issue is the variation of delays within the network. Often due to heavy network load, this variation of delay may manifest itself on a packet-to-packet basis. For example, consider two packets belonging to an Internet telephony application, sent from a source with a fixed spacing of x msec. Due to interfering traffic, two abnormal situations can occur. In the first case, these packets may reach the destination with the spacing between the packets vastly exceeding x msec. In the second case, these packets may reach the destination with the spacing between the packets much less than x msec. In both these situations, the resulting audio quality deteriorates at the receiver end because voice applications are very sensitive to variation in delay.

QoS is a generic term used to describe the over-all experience an application will receive over a network.

Quality of Service

To overcome the above limitations, researchers started looking at providing special service for supporting delay sensitive multimedia applications over an IP-based infrastructure. In networking parlance, service is measured by how well the packets of an application are handled in the network. The way real-time packets are handled can be measured using different quantities called the characteristics of packet transmission. The aggregate value of this measure is termed as Quality of Service (QoS). Hence, QoS is a generic term used to describe the over-all experience an application will receive over a network. QoS can also be looked from the perspective of Internet service providers (ISPs). This is explained with the help of an analogy. A letter can be delivered using either ordinary mail or courier service. When compared to the ordinary mail, the courier service provides assured delivery service to the customer provided he makes the requisite payment. Similarly, an Internet service provider will want to provide a consistent and an assured level of service at any point of time in the network only to premium customers.

An Internet service provider will want to provide a consistent and an assured level of service at any point of time in the network only to premium customers.

Pillars of QoS

Quality of Service can be better explained in terms of the new architectural components that can be added to the current IP infrastructure. These architectural components are aptly called 'The Four Pillars of QoS' [1]. These four Pillars of QoS have been designed so as to enable perfectly synchronized transfer of

Marking of the packets according to a class enables a router to distinguish between the classes and provide priority service to delay sensitive applications.

packets of different applications. They are:

Packet Classification: The varied nature of applications running on the Internet necessitates a different type of service for each category of applications. Hence, the traffic can be classified into different classes depending on the type of service required by each application. Marking of the packets according to a class enables a router to distinguish between the classes and provide priority service to delay sensitive applications. It also ensures that non-critical traffic (e.g. downloading of music files!) do not hog the network bandwidth.

Isolation, Scheduling and Policing: When traffic of many applications is passing through a shared link, it results in the traffic of one application infringing upon the bandwidth reserved for another application. So there should be a mechanism to isolate the different traffic flows. This will enable proper scheduling and monitoring of packets of different applications.

High Resource Utilization: Isolation is necessary. However it should not hamper the efficiency of reuse of network resources (link bandwidth, router buffer space and computational capacity of router forwarding engine). If resources allocated to a traffic flow are not being used, then another flow must be able to use the resources.

A traffic flow must declare its QoS requirements prior to entering the network. A network can then decide on the admission of this traffic flow based on the available bandwidth at that time.

Call Admission: A link can get overloaded if the combined bandwidth requirement of the traffic flows exceeds the capacity of the link. For example consider two similar applications needing 1Mbps each and there exists a 1.5Mbps link between intermediate routers. As the link cannot simultaneously handle the requirement of these two applications, the bandwidth has to be equally divided into 0.75Mbps for each application. This results in an abysmally low quality of service for both applications. So a traffic flow must declare its QoS requirements prior to entering the network. A network can then decide on the admission of this traffic flow based on the available bandwidth at that time.

Box 1. Asynchronous Transfer Mode (ATM) Networks

Initially there were broadly two types of Networks – telephone networks and data networks. The primitive data networks were primarily used to transfer text file, facilitate remote logins and provide email support. There was a need for a networking technology that could transfer real-time audio and video along with the traditional data traffic. This gave rise to the first ATM networks. ATM was developed to have a very efficient QoS implementation. It supports traffic with different bit-rates. ATM networks transmit only small fixed size packets known as cells. The small fixed length cells are well suited to transfer video and voice traffic as such traffic is intolerant of delays that could result from having to wait for a large packet to be transmitted. In ATM, a virtual circuit (path) is setup between the source and destination before data is transmitted. These circuits are called virtual channels. A *virtual channel identifier* (VCI) uniquely identifies each such circuit. The VCI is used to map the cells from the incoming links to the outgoing links. This process is called *switching* and is faster than traditional routing used in data networks. So they have been deployed in the core networks (trunk lines). But ATM also has its own set of deficiencies. It is not suitable for deployment in desktops and small networks like Local Area Networks (LAN). This is due to the fact that the overhead of setting up circuits offsets the speed advantage due to switching.

The advantages that a customer gets from QoS deployment are:

- Differential treatment to applications with different communication requirements.
- Minimization of end-to-end transfer delay.
- Tangible reduction in loss of packets that occurs when routers/switches drop packets due to congestion.

Box 2. Commonly Used QoS Characteristics

Bandwidth: The bandwidth of a channel is the range of frequencies that is passed by the channel. But this is proportional to the channel capacity i.e. the number of bits that can be transmitted per second. So it has become synonymous with the speed at which data can be transmitted through a link measured as bits per second (bps).

Latency: It is defined as the time interval between when the client requests for the TCP connection to the time when the requested data is received in its entirety. For example, consider the World Wide Web. Here, the latency is the time interval between the request for the web page (when you type in the URL and press the 'Enter' key), and the time when the complete web page is received.

Jitter: The transmission of a packet from a source to a destination takes a finite amount of time. Sometimes due to network faults or congestion, this end-to-end delay varies from packet to packet. The variation in this delay is called jitter.

QoS is a relatively new phenomenon in IP networks and involves a broad range of technologies, architecture and protocols. The focus has changed to building QoS onto the existing IP network infrastructure rather than replacing it with ATM networks.

QoS is a relatively new phenomenon in IP networks and involves a broad range of technologies, architecture and protocols. ATM networks already had a QoS implementation. On first thought, it would seem that replacing the existing IP networks with ATM networks would solve the problem. However the complex implementation coupled with the deployment issues has made ATM on desktop a utopian dream! So the focus has changed to building QoS onto the existing IP network infrastructure rather than replacing it with ATM networks.

Initially, IP Networks had a very basic implementation of QoS. This basically involved using an 8-bit field called 'Type of service' in the IP header to indicate the service assigned to the IP packet. However it was not possible to distinguish between different service levels. Also, the use of this field was optional. To overcome this hurdle, major networking companies like Cisco systems started developing proprietary QoS solutions. Now to ensure smooth operation of the Internet, it was essential to ensure that all vendors used a standardized QoS model. Keeping this in mind, Internet Engineering Task Force (IETF) defined two models to provide QoS namely, *Integrated Services* (Int-Serv) [2, 3] & *Differentiated Services* (Diff-Serv) [4]. In this article we will explain the IntServ model. In another article we will cover the DiffServ model and the integration of these two models with a robust protocol namely MPLS, thereby providing the holistic view of QoS deployment in the Internet.

Integrated Services

Integrated services framework facilitates the transmission of both real-time and non-real time traffic over the same IP network with individualized QoS.

In the early 90s', IETF community began to work on a framework that would facilitate the transmission of both real-time and non-real time traffic over the same IP network with individualized quality-of-service. This framework was called Integrated Services (IntServ) framework [3].

Consider the following scenario; An IP telephony session arrives at the edge of a network needing to go from one end of the network to another within a given amount of time.

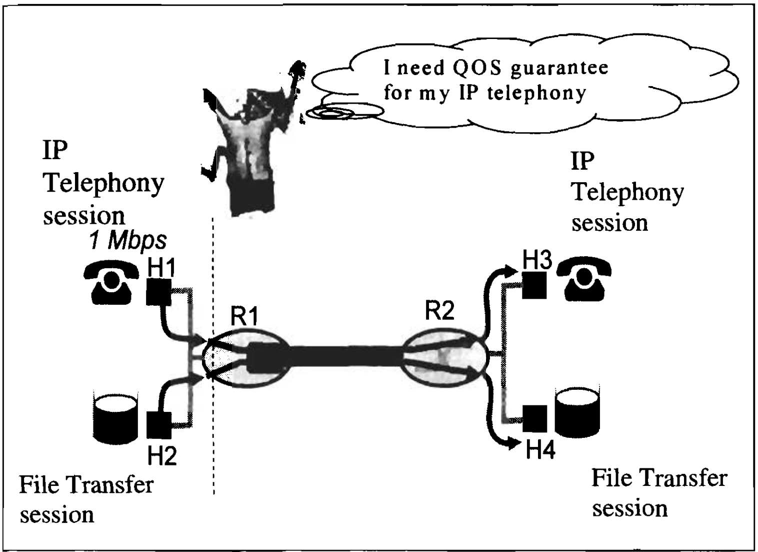

Figure 1. IntServ Framework.

Now if the IntServ model is implemented in *Figure* 1, the network will have to address the following question. 'Can I guarantee that this session will go from one end of the network to another with no more than this loss and no more than this delay without violating the QoS guarantees already given to the existing sessions'?

Different applications need different levels of delivery service. The IntServ model provides multiple controlled levels of delivery service. IntServ supports three types of traffic – *best-effort*, *rate-sensitive* and *delay-sensitive*. Delay-sensitive traffic has hard real-time requirements and requires a guaranteed service. Common applications mapping to delay sensitive traffic are audio 'playback' and streaming video applications wherein the applications are intolerant of any packet arriving after their play-back time [5]. Rate-sensitive traffic is sensitive to variation in the amount of bandwidth available. Applications such as H.323 videoconferencing are rate-sensitive and are designed to run at a nearly constant rate. Once a rate-sensitive session has been established, IntServ will not grant a subsequent request that would cause the network to provide less than the required rate to existing rate-guaranteed sessions.

IntServ supports three types of traffic – *best-effort*, *rate-sensitive* and *delay-sensitive*.

IntServ uses two control mechanisms namely, *Guaranteed Service* and *Controlled-Load Service*.

To control the quality of service delivered to packets, IntServ uses two control mechanisms namely, *Guaranteed Service* and *Controlled-Load Service*. In simple terms, Guaranteed Service envisages ensuring that the packets are forwarded to the next router at the earliest possible time with strict conformance to the promised traffic characteristics. Guaranteed service is useful only if it is provided by every network element along the route. On the other hand, Controlled-Load Service is used for applications that can adapt and tolerate some delay occurring in the network (rate-adaptive multimedia applications). It assures that a 'very high percentage' of packets will pass through the network without being dropped and will experience a very negligible queuing delay [6]. However the service does not quantify what constitutes a 'very high percentage' of packets. The performance is satisfactory when the network is lightly loaded, but degrades when the network is overloaded.

Resource ReSerVation Protocol

One of the ways QoS is realized in the integrated services Internet is by using a completely separate resource reservation setup protocol termed RSVP. *Resource ReSerVation Protocol* (RSVP) is a signaling protocol used to communicate the service requirements of applications to the routers i.e., RSVP is a bandwidth reservation protocol. It enables an application to request a specific QoS for the transmission of data between a source-destination pair. In other words, it allows an application to dynamically reserve network bandwidth. RSVP's basic operation consists of providing reservations for network resources in multicast scenarios (unicast scenario is a degenerative case of multicast) before the actual transmission of data. RSVP works in conjunction with routing protocols. The routing protocols compute the shortest paths between hosts based on link cost and other administrative constraints. This information is used by RSVP for setting up the paths. RSVP's operation is independent of the underlying routing protocol. It is necessary to clarify that RSVP does not mandate the exact provisioning of the bandwidth to the data flows. RSVP only allows applications to reserve

RSVP's basic operation consists of providing reservations for network resources in multicast scenarios before the actual transmission of data.

resources, once the reservations are in place, the routers packet schedulers must actually provide the reserved bandwidth to the data flows.

Data Flow and Session

A data flow is a sequence of messages meaningful to an application with the same source, destination and transport layer protocol that require a specific QoS, e.g., a video stream between a given host pair. A *session* is a set of data flows having the same unicast or multicast destination address. A session is simplex in nature. Bi-directional exchange of data requires two separate RSVP simplex sessions. RSVP treats each session independently. A flow can originate only from a single sender, whereas a session can have any number of senders and receivers. A typical example of a session is video conferencing where at the receiving host; we can see audio and video of different senders who are located at different locations.

Box 3. Multicast Scenario

A network delivery scenario wherein the routing algorithm, switching function, and call setup mechanisms are designed to allow a packet that is sent just once by a sender to be delivered to multiple destinations.

Box 4. Classification of Traffic Based on the Nature of Application.

Traffic Class	Typical Application
Real Time – Packets must reach the destination within a fixed time frame	Voice over IP (VOIP) better known as Internet Telephony
Interactive – The response time must be within a time frame acceptable to the user	Telnet (provides a virtual terminal for running tasks on another machine)
Video – There should be no jitter i.e. the time lag between two consecutive frames must be constant	Movies or video clips viewed on the web (NetMeeting or Streaming Video)
Transactional - Here the time taken is not as important as the correctness of data	E-commerce i.e., buying and selling on the Internet (the database must be updated without errors)
Best-Effort – The normal internet service wherein no guarantees for delivery of data is provided	Web browsing and file transfer applications
Control – Routing tables and other network information, have to be sent using the same networks. They have to be given top priority	Network Management Traffic and Routing Protocol Traffic

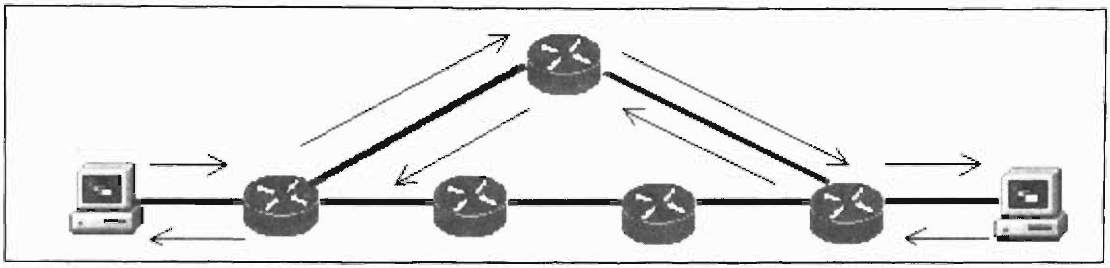

Figure 2. RSVP Path.

RSVP Operation

A path between the source and destination is setup with the help of a dynamic unicast/multicast IP routing protocol before the transmission of the actual data. All the network resources needed by the data flow are also reserved during the *path* setup. The service parameters that the application desires are 'advertised' in a Path message sent by the source. In RSVP parlance, the service parameters are expressed in the path message in the form of *SENDER_TSPEC* and *ADSPEC*. *SENDER_TSPEC* describes the characteristic of traffic flow (such as Token bucket rate, peak rate etc., [3]) and *ADSPEC* carries information about QoS control capability (Minimum path latency, Composed MTU etc., [3]) requirements of the sender. Each node in the path updates the ADSPEC to depict the QoS capability of the path till that node.

A path between the source and destination is setup with the help of a dynamic unicast/multicast IP routing protocol before the transmission of the actual data. All the network resources needed by the data flow are also reserved during the *path* setup.

Each router that receives the path message checks whether it can allocate the requested resources. If the resources can be allocated without infringing on the resources committed to the existing sessions, the *Path* message is forwarded to the next hop as determined by the routing protocol. Otherwise, a *PathErr* message is sent to the initiating host. When the sender receives a *PathErr* message, it responds with a *PathTear* message. The *PathTear* message is sent in the downstream direction. All participating routers on receiving this message immediately remove the corresponding Path State information and free the allocated resources.

When the receiving host gets the *Path* message, it responds with a *Reservation-Request* (Resv) message. In RSVP parlance, *Resv* message defines the desired QoS in the form of *flow descriptor*. The *flow descriptor* is a combination of the classification parameters that define the source-side of the flow (source IP address, source port number of the sending application [2]) and the scheduling parameters that define the QoS of the flow (Token Bucket rate and size, peak data rate, etc. [3]). In RSVP parlance, classification parameters are called *FilterSpec*, while scheduling parameters are called *FlowSpec*. When each of the upstream router(s) along the path receives the *Resv* message, resources are marked for allocation and information about the data flow is stored as a Path State. Once the sender receives the *Resv* message it assumes that the path has been setup and it starts sending data packets.

The *flow descriptor* is a combination of the classification parameters that define the source-side of the flow (source IP address, source port number of the sending application [2]) and the scheduling parameters that define the QoS of the flow (Token Bucket rate and size, peak data rate, etc. [3]).

In RSVP, the Path State is a soft state i.e. it can be changed without consulting the end points thereby enabling re-routing in case of failures. Since the resources are reserved only on the receipt of a *Resv* message from the receiving host, RSVP is called a *Receiver Initiated Protocol*. During the path-setup if any error occurs in the *Resv* message, then a *ResvErr* message is sent to the receiving host. On receiving a *ResvErr* message, the receiver host responds with a *ResvTear* message to its upstream router(s) directing each of the routers to remove the corresponding reservation states.

The *Path* and *Resv* messages are periodically sent to refresh the Path State information. In case the refresh messages are not received by the participating routers within a fixed time interval (called the *cleanup timeout period*), the Path State is deleted and the path is torn down.

Reservation Styles

Reservation style refers to a set of control options that specify how the reservation of resources is made for data flow of the different senders in the same session. RSVP supports two major classes of reservation – *distinct* reservations and *shared* reservations. In a distinct reservation, separate reservation is made for

Reservation style refers to a set of control options that specify how the reservation of resources is made for data flow of the different senders in the same session.

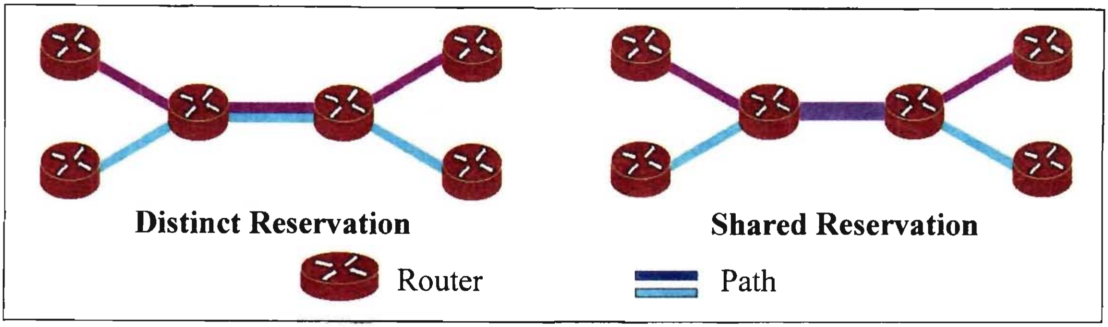

Figure 3. Reservation Styles.

each sender data flow. In a shared reservation, a single reservation is made, into which data flows from different senders share the bandwidth. A good analogy to the shared reservation concept would be channels on a television. The channels are sent by multiple stations and share the bandwidth allocated to the TV media while not interfering with each other.

In RSVP parlance, distinct reservation is called *Fixed-Filter* (FF) style. In this style, a distinct reservation request is made for data packets from a sender. Applications requiring guaranteed service normally use this reservation style (e.g. Video Conferencing). The shared reservation is of two types – *Wildcard-Filter* (WF) style and *Shared-Explicit* (SE) style. The wild-card filter style creates a single reservation shared by all the senders. Here, the reserved size is the largest if the resource requests from all receivers. Consider the following example; an audio conference call is scheduled between individuals P, Q and R working in WIPRO's Sarjapur, Koramangala and Madivala locations respectively. Each need reservations of 'x', 'y' and 'z' Kbytes of bandwidth respectively where x > y and z, if we use Wildcard filter, a single 'x' Kbytes reservation is made and is shared by all the senders. Remember that in an audio conference call, only one individual can talk at a time and hence the shared reservation is used by the person currently speaking.

In a distinct reservation, separate reservation is made for each sender data flow. In a shared reservation, a single reservation is made, into which data flows from different senders share the bandwidth.

When a receiver uses the shared-explicit style in its reservation message, it specifies a list of senders from which it wants to receive a data flow along with a single bandwidth reservation. This reservation is to be shared among all the senders in the list.

Limitations of Integrated Services

We have seen how IntServ strives to ensure that necessary network resources are provided along the path that the real-time packets will follow. However there are some deficiencies in the current IntServ model with respect to scalability and flexibility.

Lack of Scalability: If reservation protocols like RSVP are used, then the resource reservation information about each data flow should be maintained in all participation routers. In large networks, especially in Backbone routers, the number of source-destination pairs at any instant may be very large. So maintaining this information will pose a considerable overhead. If there are ½ million flows going through a router/sec, just imagine the overhead involved in maintaining the per-flow router state and performing the signalling activity! Hence there is a preference to deploy RSVP predominantly in the edges of the network where scalability is not a major concern.

In RSVP resource reservation information about each data flow should be maintained in all participation routers.

Inflexible Service Models: We have seen that the IntServ model can provide individualized QoS guarantees to different traffic flows. But there is a catch; when all the available RSVP resources are consumed, new requests are rejected until the existing RSVP-allocated resources are released. This is similar in functionality to how the telephone system works, where the networks response to a connection request is either commitment or denial. Such a service may not prove to be a viable method to operate in a data network environment where best effort services arguably should always be available. Remember that signalling in telephone networks is much simpler than signalling via RSVP protocol.

In RSVP when all the available RSVP resources are consumed, new requests are rejected until the existing RSVP-allocated resources are released.

Conclusion

In this article we have understood the limitations of best-effort service and the necessity of QoS. We have also looked at how the IntServ model can be used for making solid QoS guarantees using the signaling protocol (RSVP). In the absence of QoS, the World Wide Web (WWW) would have been 'World Wide Wait' for a user of Internet voice applications. Given the phenomenal rate of growth in usage of real-time multimedia applications over the Internet, attention must be paid to optimal network resource management. We will look at this in more detail in another article.

Suggested Reading

[1] James F Kurose and Keith W Ross, *Computer Networking*, Addison Wesley, 2000.

[2] RFC 2205, Resource ReSerVation Protocol (RSVP) – Functional Specification, *IETF*, September 1997. (http://www.ietf.org/rfc/rfc2205.txt)

[3] RFC 2210, The Use of RSVP with IETF Integrated Services, *IETF*, September 1997. (http://www.ietf.org/rfc/rfc2210.txt)

[4] RFC 2474, Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6Headers, *IETF*, December1998. (http://www.ietf.org/rfc/rfc2474.txt)

[5] Alberto Leon-Garcia & Indra Widjaja, *Communication Networks-Fundamental Concepts and Key Architectures*, McGraw-Hill, 2000.

[6] Chris Metz, IP QOS: Traveling in First Class on the Internet, *IEEE Internet Computing*, March-April 1999.

Address for Correspondence
Prashant Bharadwaj
273, 10th Cross, N.R.Colony
Bangalore 560019, India.
Email: prashant.bharadwaj@wipro.com

Statement about ownership and other particulars of *Resonance* – journal of science education

1. Place of Publication	:	Bangalore
2. Periodicity of Publication	:	Monthly
3. Printer's Name	:	N Mukunda Indian Academy of Sciences Bangalore 560 080
4. and 5. Publisher and Editor	:	N Mukunda
6. Nationality	:	Indian
7. Address	:	Indian Academy of Sciences P.B. No. 8005, Bangalore 560 080
8. Name and Address of the Owner	:	Indian Academy of Sciences Bangalore 560 080

I, N Mukunda, hereby declare that the particulars given above are true to the best of my knowledge.

Date: 1st March 2005

N Mukunda
Signature of Publisher