CHAPTER 19

Internetwork Operation

She occupied herself with studying a map on the opposite wall because she knew she would have to change trains at some point. Tottenham Court Road must be that point, an interchange from the black line to the red. This train would take her there, was bearing her there rapidly now, and at the station she would follow the signs, for signs there must be, to the Central Line going westward.

—*King Solomon's Carpet*, Barbara Vine (Ruth Rendell)

KEY POINTS

- The act of sending a packet from a source to multiple destinations is referred to as multicasting. Multicasting raises design issues in the areas of addressing and routing.
- Routing protocols in an internet function in a similar fashion to those used in packet-switching networks. An internet routing protocol is used to exchange information about reachability and traffic delays, allowing each router to construct a next-hop routing table for paths through the internet. Typically, relatively simple routing protocols are used between autonomous systems within a larger internet and more complex routing protocols are used within each autonomous system.
- The integrated services architecture is a response to the growing variety and volume of traffic experienced in the Internet and intranets. It provides a framework for the development of protocols such as RSVP to handle multimedia/multicast traffic and provides guidance to router vendors on the development of efficient techniques for handling a varied load.
- The differentiated services architecture is designed to provide a simple, easy-to-implement, low-overhead tool to support a range of network services that are differentiated on the basis of performance. Differentiated services are provided on the basis of a 6-bit label in the IP header, which classifies traffic in terms of the type of service to be given by routers for that traffic.

As the Internet and private internets grow in scale, a host of new demands march steadily into view. Low-volume TELNET conversations are leapfrogged by high-volume client/server applications. To this has been added more recently the tremendous volume of Web traffic, which is increasingly graphics intensive. Now real-time voice and video applications add to the burden.

To cope with these demands, it is not enough to increase internet capacity. Sensible and effective methods for managing the traffic and controlling congestion are needed. Historically, IP-based internets have been able to provide a simple best-effort delivery service to all applications using an internet. But the needs of users have changed. A company may have spent millions of dollars installing an IP-based internet designed to transport data among LANs but now finds that new real-time, multimedia, and multicasting applications are not well supported by such a configuration. The only networking scheme designed from day one to support both traditional TCP and UDP traffic and real-time traffic is ATM. However, reliance on ATM means either constructing a second networking infrastructure for real-time traffic or replacing the existing IP-based configuration with ATM, both of which are costly alternatives.

Thus, there is a strong need to be able to support a variety of traffic with a variety of quality-of-service (QoS) requirements, within the TCP/IP architecture. This chapter looks at the internetwork functions and services designed to meet this need.

We begin this chapter with a discussion of multicasting. Next we explore the issue of internetwork routing algorithms. Next, we look at the Integrated Services Architecture (ISA), which provides a framework for current and future internet services. Then we examine differentiated services. Finally, we introduce the topics of service level agreements and IP performance metrics.

Refer to Figure 2.5 to see the position within the TCP/IP suite of the protocols discussed in this chapter.

19.1 MULTICASTING

Typically, an IP address refers to an individual host on a particular network. IP also accommodates addresses that refer to a group of hosts on one or more networks. Such addresses are referred to as **multicast addresses**, and the act of sending a packet from a source to the members of a multicast group is referred to as **multicasting**.

Multicasting has a number of practical applications. For example,

- **Multimedia:** A number of users “tune in” to a video or audio transmission from a multimedia source station.
- **Teleconferencing:** A group of workstations form a multicast group such that a transmission from any member is received by all other group members.
- **Database:** All copies of a replicated file or database are updated at the same time.
- **Distributed computation:** Intermediate results are sent to all participants.
- **Real-time workgroup:** Files, graphics, and messages are exchanged among active group members in real time.

Multicasting done within the scope of a single LAN segment is straightforward. IEEE 802 and other LAN protocols include provision for MAC-level multicast addresses. A packet with a multicast address is transmitted on a LAN segment. Those stations that are members of the corresponding multicast group recognize the multicast address and accept the packet. In this case, only a single copy of the packet is ever transmitted. This technique works because of the broadcast nature of a LAN: A transmission from any one station is received by all other stations on the LAN.

In an internet environment, multicasting is a far more difficult undertaking. To see this, consider the configuration of Figure 19.1; a number of LANs are interconnected by routers. Routers connect to each other either over high-speed links or across a wide area network (network N4). A cost is associated with each link or network in

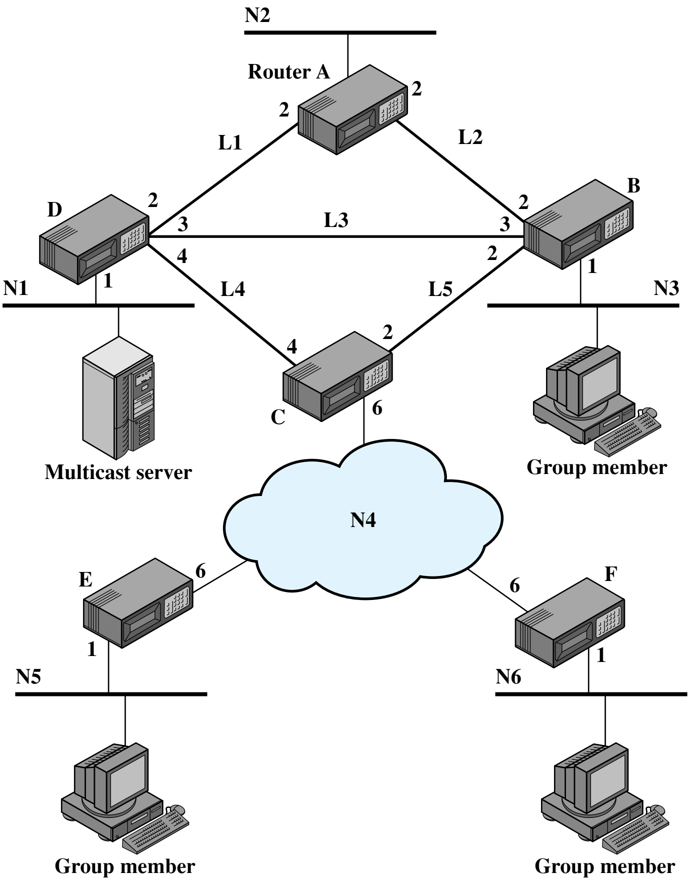


Figure 19.1 Example Configuration

each direction, indicated by the value shown leaving the router for that link or network. Suppose that the multicast server on network N1 is transmitting packets to a multicast address that represents the workstations indicated on networks N3, N5, N6. Suppose that the server does not know the location of the members of the multicast group. Then one way to assure that the packet is received by all members of the group is to **broadcast** a copy of each packet to each network in the configuration, over the least-cost route for each network. For example, one packet would be addressed to N3 and would traverse N1, link L3, and N3. Router B is responsible for translating the IP-level multicast address to a MAC-level multicast address before transmitting the MAC frame onto N3. Table 19.1 summarizes the number of packets generated on the various links and networks in order to transmit one packet to a multicast group by this method. In this table, the source is the multicast server on network N1 in Figure 19.1; the multicast address includes the group members on N3, N5, and N6. Each column in the table refers to the path taken from the source host to a destination router attached to a particular destination network. Each row of the table refers to a network or link in the configuration of Figure 19.1. Each entry in the table gives the number of packets that

Table 19.1 Traffic Generated by Various Multicasting Strategies

	(a) Broadcast					(b) Multiple Unicast				(c) Multicast
	S → N2	S → N3	S → N5	S → N6	Total	S → N3	S → N5	S → N6	Total	
N1	1	1	1	1	4	1	1	1	3	1
N2										
N3		1			1	1			1	1
N4			1	1	2		1	1	2	2
N5			1		1		1		1	1
N6				1	1			1	1	1
L1	1				1					
L2										
L3		1			1	1			1	1
L4			1	1	2		1	1	2	1
L5										
Total	2	3	4	4	13	3	4	4	11	8

traverse a given network or link for a given path. A total of 13 copies of the packet are required for the broadcast technique.

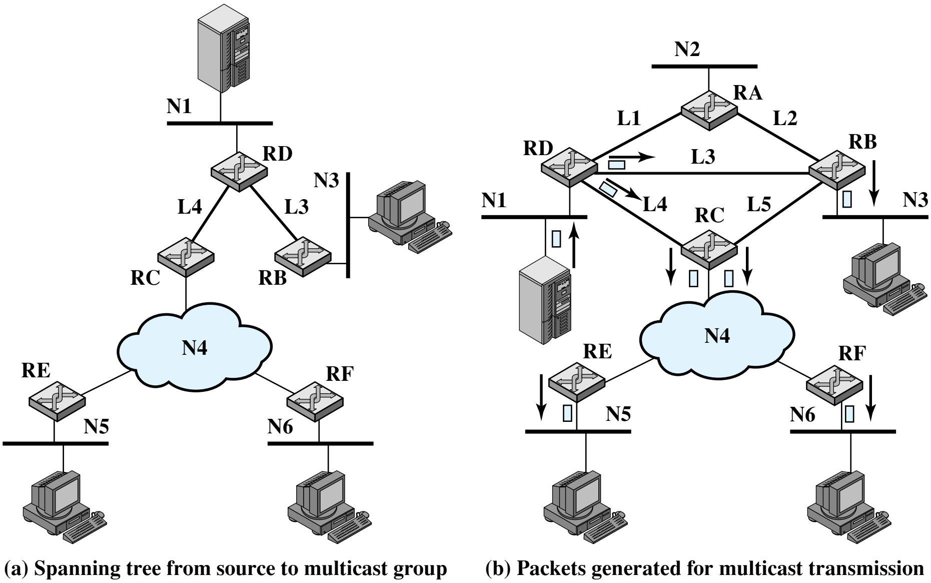
Now suppose the source system knows the location of each member of the multicast group. That is, the source has a table that maps a multicast address into a list of networks that contain members of that multicast group. In that case, the source need only send packets to those networks that contain members of the group. We could refer to this as the **multiple unicast** strategy. Table 19.1 shows that in this case, 11 packets are required.

Both the broadcast and multiple unicast strategies are inefficient because they generate unnecessary copies of the source packet. In a true **multicast** strategy, the following method is used:

1. The least-cost path from the source to each network that includes members of the multicast group is determined. This results in a spanning tree[1] of the configuration. Note that this is not a full spanning tree of the configuration. Rather, it is a spanning tree that includes only those networks containing group members.
2. The source transmits a single packet along the spanning tree.
3. The packet is replicated by routers only at branch points of the spanning tree.

Figure 19.2a shows the spanning tree for transmissions from the source to the multicast group, and Figure 19.2b shows this method in action. The source transmits a single packet over N1 to router D. D makes two copies of the packet, to transmit over

[1] The concept of spanning tree was introduced in our discussion of bridges in Chapter 15. A spanning tree of a graph consists of all the nodes of the graph plus a subset of the links (edges) of the graph that provides connectivity (a path exists between any two nodes) with no closed loops (there is only one path between any two nodes).



(a) Spanning tree from source to multicast group (b) Packets generated for multicast transmission

Figure 19.2 Multicast Transmission Example

links L3 and L4. B receives the packet from L3 and transmits it on N3, where it is read by members of the multicast group on the network. Meanwhile, C receives the packet sent on L4. It must now deliver that packet to both E and F. If network N4 were a broadcast network (e.g., an IEEE 802 LAN), then C would only need to transmit one instance of the packet for both routers to read. If N4 is a packet-switching WAN, then C must make two copies of the packet and address one to E and one to F. Each of these routers, in turn, retransmits the received packet on N5 and N6, respectively. As Table 19.1 shows, the multicast technique requires only eight copies of the packet.

Requirements for Multicasting

In ordinary unicast transmission over an internet, in which each datagram has a unique destination network, the task of each router is to forward the datagram along the shortest path from that router to the destination network. With multicast transmission, the router may be required to forward two or more copies of an incoming datagram. In our example, routers D and C both must forward two copies of a single incoming datagram.

Thus, we might expect that the overall functionality of multicast routing is more complex than unicast routing. The following is a list of required functions:

1. A convention is needed for identifying a multicast address. In IPv4, Class D addresses are reserved for this purpose. These are 32-bit addresses with 1110 as their high-order 4 bits, followed by a 28-bit group identifier. In IPv6, a 128-bit multicast address consists of an 8-bit prefix of all ones, a 4-bit flags field, a 4-bit scope field, and a 112-bit group identifier. The flags field, currently, only indicates whether this address is permanently assigned or not. The scope field indicates the scope of applicability of the address, ranging from a single network to global.

2. Each node (router or source node participating in the routing algorithm) must translate between an IP multicast address and a list of networks that contain members of this group. This information allows the node to construct a shortest-path spanning tree to all of the networks containing group members.
3. A router must translate between an IP multicast address and a network multicast address in order to deliver a multicast IP datagram on the destination network. For example, in IEEE 802 networks, a MAC-level address is 48 bits long; if the highest-order bit is 1, then it is a multicast address. Thus, for multicast delivery, a router attached to an IEEE 802 network must translate a 32-bit IPv4 or a 128-bit IPv6 multicast address into a 48-bit IEEE 802 MAC-level multicast address.
4. Although some multicast addresses may be assigned permanently, the more usual case is that multicast addresses are generated dynamically and that individual hosts may join and leave multicast groups dynamically. Thus, a mechanism is needed by which an individual host informs routers attached to the same network as itself of its inclusion in and exclusion from a multicast group. IGMP, described subsequently, provides this mechanism.
5. Routers must exchange two sorts of information. First, routers need to know which networks include members of a given multicast group. Second, routers need sufficient information to calculate the shortest path to each network containing group members. These requirements imply the need for a multicast routing protocol. A discussion of such protocols is beyond the scope of this book.
6. A routing algorithm is needed to calculate shortest paths to all group members.
7. Each router must determine multicast routing paths on the basis of both source and destination addresses.

The last point is a subtle consequence of the use of multicast addresses. To illustrate the point, consider again Figure 19.1. If the multicast server transmits a unicast packet addressed to a host on network N5, the packet is forwarded by router D to C, which then forwards the packet to E. Similarly, a packet addressed to a host on network N3 is forwarded by D to B. But now suppose that the server transmits a packet with a multicast address that includes hosts on N3, N5, and N6. As we have discussed, D makes two copies of the packet and sends one to B and one to C. What will C do when it receives a packet with such a multicast address? C knows that this packet is intended for networks N3, N5, and N6. A simple-minded approach would be for C to calculate the shortest path to each of these three networks. This produces the shortest-path spanning tree shown in Figure 19.3. As a result, C sends two copies of the packet out over N4, one intended for N5 and one intended for N6. But it also sends a copy of the packet to B for delivery on N3. Thus B will receive two copies of the packet, one from D and one from C. This is clearly not what was intended by the host on N1 when it launched the packet.

To avoid unnecessary duplication of packets, each router must route packets on the basis of both source and multicast destination. When C receives a packet intended for the multicast group from a source on N1, it must calculate the spanning

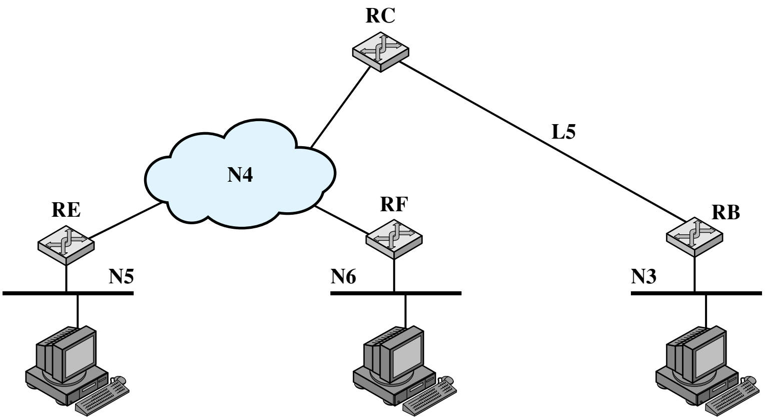


Figure 19.3 Spanning Tree from Router C to Multicast Group

tree with N1 as the root (shown in Figure 19.2a) and route on the basis of that spanning tree.

Internet Group Management Protocol (IGMP)

IGMP, defined in RFC 3376, is used by hosts and routers to exchange multicast group membership information over a LAN. IGMP takes advantage of the broadcast nature of a LAN to provide an efficient technique for the exchange of information among multiple hosts and routers. In general, IGMP supports two principal operations:

1. Hosts send messages to routers to subscribe to and unsubscribe from a multicast group defined by a given multicast address.
2. Routers periodically check which multicast groups are of interest to which hosts.

IGMP is currently at version 3. In IGMPv1, hosts could join a multicast group and routers used a timer to unsubscribe group members. IGMPv2 enabled a host to request to be unsubscribed from a group. The first two versions used essentially the following operational model:

- Receivers have to subscribe to multicast groups.
- Sources do not have to subscribe to multicast groups.
- Any host can send traffic to any multicast group.

This paradigm is very general, but it also has some weaknesses:

1. Spamming of multicast groups is easy. Even if there are application level filters to drop unwanted packets, still these packets consume valuable resources in the network and in the receiver that has to process them.
2. Establishment of the multicast distribution trees is problematic. This is mainly because the location of sources is not known.

3. Finding globally unique multicast addresses is difficult. It is always possible that another multicast group uses the same multicast address.

IGMPv3 addresses these weaknesses by

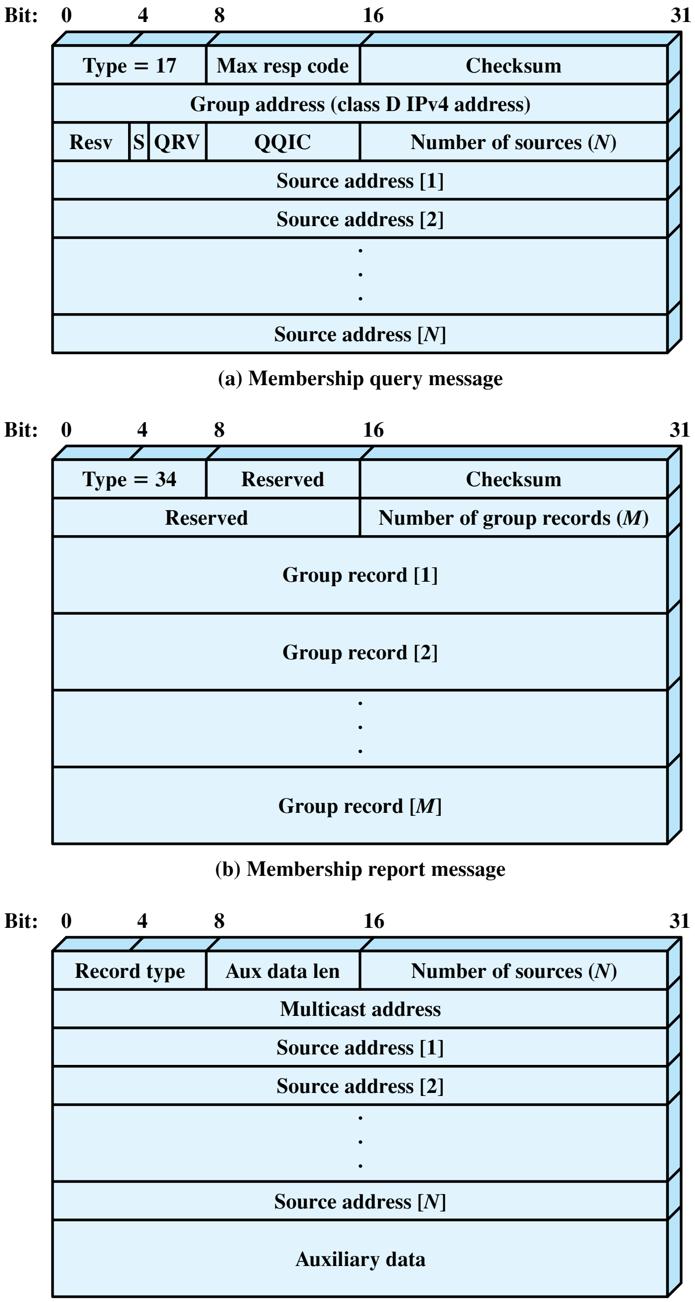
1. Allowing hosts to specify the list of hosts from which they want to receive traffic. Traffic from other hosts is blocked at routers.
2. Allowing hosts to block packets that come from sources that send unwanted traffic.

The remainder of this section discusses IGMPv3.

IGMP Message Format All IGMP messages are transmitted in IP datagrams. The current version defines two message types: Membership Query and Membership Report.

A **Membership Query** message is sent by a multicast router. There are three subtypes: a **general query**, used to learn which groups have members on an attached network; a **group-specific query**, used to learn if a particular group has any members on an attached network; and a **group-and-source-specific query**, used to learn if any attached device desires reception of packets sent to a specified multicast address, from any of a specified list of sources. Figure 19.4a shows the message format, which consists of the following fields:

- **Type:** Defines this message type.
- **Max Response Code:** Indicates the maximum allowed time before sending a responding report in units of 1/10 second.
- **Checksum:** An error-detecting code, calculated as the 16-bit ones complement addition of all the 16-bit words in the message. For purposes of computation, the Checksum field is itself initialized to a value of zero. This is the same checksum algorithm used in IPv4.
- **Group Address:** Zero for a general query message; a valid IP multicast group address when sending a group-specific query or group-and-source-specific query.
- **S Flag:** When set to one, indicates to any receiving multicast routers that they are to suppress the normal timer updates they perform upon hearing a query.
- **QRV (querier's robustness variable):** If nonzero, the QRV field contains the RV value used by the querier (i.e., the sender of the query). Routers adopt the RV value from the most recently received query as their own RV value, unless that most recently received RV was zero, in which case the receivers use the default value or a statically configured value. The RV dictates how many times a host will retransmit a report to assure that it is not missed by any attached multicast routers.
- **QQIC (querier's querier interval code):** Specifies the QI value used by the querier, which is a timer for sending multiple queries. Multicast routers that are not the current querier adopt the QI value from the most recently received query as their own QI value, unless that most recently received QI was zero, in which case the receiving routers use the default QI value.



(a) Membership query message

(b) Membership report message

(c) Group record

Figure 19.4 IGMPv3 Message Formats

- **Number of Sources:** Specifies how many source addresses are present in this query. This value is nonzero only for a group-and-source-specific query.
- **Source Addresses:** If the number of sources is N, then there are N 32-bit unicast addresses appended to the message.

A **Membership Report** message consists of the following fields:

- **Type:** Defines this message type.
- **Checksum:** An error-detecting code, calculated as the 16-bit ones complement addition of all the 16-bit words in the message.
- **Number of Group Records:** Specifies how many group records are present in this report.
- **Group Records:** If the number of group records is M, then there are M 32-bit unicast group records appended to the message.

A group record includes the following fields:

- **Record Type:** Defines this record type, as described subsequently.
- **Aux Data Length:** Length of the auxiliary data field, in 32-bit words.
- **Number of Sources:** Specifies how many source addresses are present in this record.
- **Multicast Address:** The IP multicast address to which this record pertains.
- **Source Addresses:** If the number of sources is N, then there are N 32-bit unicast addresses appended to the message.
- **Auxiliary Data:** Additional information pertaining to this record. Currently, no auxiliary data values are defined.

IGMP Operation The objective of each host in using IGMP is to make itself known as a member of a group with a given multicast address to other hosts on the LAN and to all routers on the LAN. IGMPv3 introduces the ability for hosts to signal group membership with filtering capabilities with respect to sources. A host can either signal that it wants to receive traffic from all sources sending to a group except for some specific sources (called EXCLUDE mode) or that it wants to receive traffic only from some specific sources sending to the group (called INCLUDE mode). To join a group, a host sends an IGMP membership report message, in which the group address field is the multicast address of the group. This message is sent in an IP datagram with the same multicast destination address. In other words, the Group Address field of the IGMP message and the Destination Address field of the encapsulating IP header are the same. All hosts that are currently members of this multicast group will receive the message and learn of the new group member. Each router attached to the LAN must listen to all IP multicast addresses in order to hear all reports.

To maintain a valid current list of active group addresses, a multicast router periodically issues an IGMP general query message, sent in an IP datagram with an *all-hosts* multicast address. Each host that still wishes to remain a member of one or more multicast groups must read datagrams with the all-hosts address. When such a host receives the query, it must respond with a report message for each group to which it claims membership.

Note that the multicast router does not need to know the identity of every host in a group. Rather, it needs to know that there is at least one group member still active. Therefore, each host in a group that receives a query sets a timer with a random delay. Any host that hears another host claim membership in the group will cancel its own report. If no other report is heard and the timer expires, a host sends a report. With this scheme, only one member of each group should provide a report to the multicast router.

When a host leaves a group, it sends a leave group message to the all-routers static multicast address. This is accomplished by sending a membership report message with the INCLUDE option and a null list of source addresses; that is, no sources are to be included, effectively leaving the group. When a router receives such a message for a group that has group members on the reception interface, it needs to determine if there are any remaining group members. For this purpose, the router uses the group-specific query message.

Group Membership with IPv6 IGMP was defined for operation with IPv4 and makes use of 32-bit addresses. IPv6 internets need this same functionality. Rather than to define a separate version of IGMP for IPv6, its functions have been incorporated into the new version of the Internet Control Message Protocol (ICMPv6). ICMPv6 includes all of the functionality of ICMPv4 and IGMP. For multicast support, ICMPv6 includes both a group-membership query and a group-membership report message, which are used in the same fashion as in IGMP.

19.2 ROUTING PROTOCOLS

The routers in an internet are responsible for receiving and forwarding packets through the interconnected set of networks. Each router makes routing decision based on knowledge of the topology and traffic/delay conditions of the internet. In a simple internet, a fixed routing scheme is possible. In more complex internets, a degree of dynamic cooperation is needed among the routers. In particular, the router must avoid portions of the network that have failed and should avoid portions of the network that are congested. To make such dynamic routing decisions, routers exchange routing information using a special routing protocol for that purpose. Information is needed about the status of the internet, in terms of which networks can be reached by which routes, and the delay characteristics of various routes.

In considering the routing function, it is important to distinguish two concepts:

- **Routing information:** Information about the topology and delays of the internet
- **Routing algorithm:** The algorithm used to make a routing decision for a particular datagram, based on current routing information

Autonomous Systems

To proceed with our discussion of routing protocols, we need to introduce the concept of an **autonomous system**. An autonomous system (AS) exhibits the following characteristics:

1. An AS is a set of routers and networks managed by a single organization.
2. An AS consists of a group of routers exchanging information via a common routing protocol.
3. Except in times of failure, an AS is connected (in a graph-theoretic sense); that is, there is a path between any pair of nodes.

A shared routing protocol, which we shall refer to as an **interior router protocol** (IRP), passes routing information between routers within an AS. The protocol used within the AS does not need to be implemented outside of the system. This flexibility allows IRPs to be custom tailored to specific applications and requirements.

It may happen, however, that an internet will be constructed of more than one AS. For example, all of the LANs at a site, such as an office complex or campus, could be linked by routers to form an AS. This system might be linked through a wide area network to other ASs. The situation is illustrated in Figure 19.5. In this case, the routing algorithms and information in routing tables used by routers in different ASs may differ. Nevertheless, the routers in one AS need at least a minimal level of information concerning networks outside the system that can be reached.

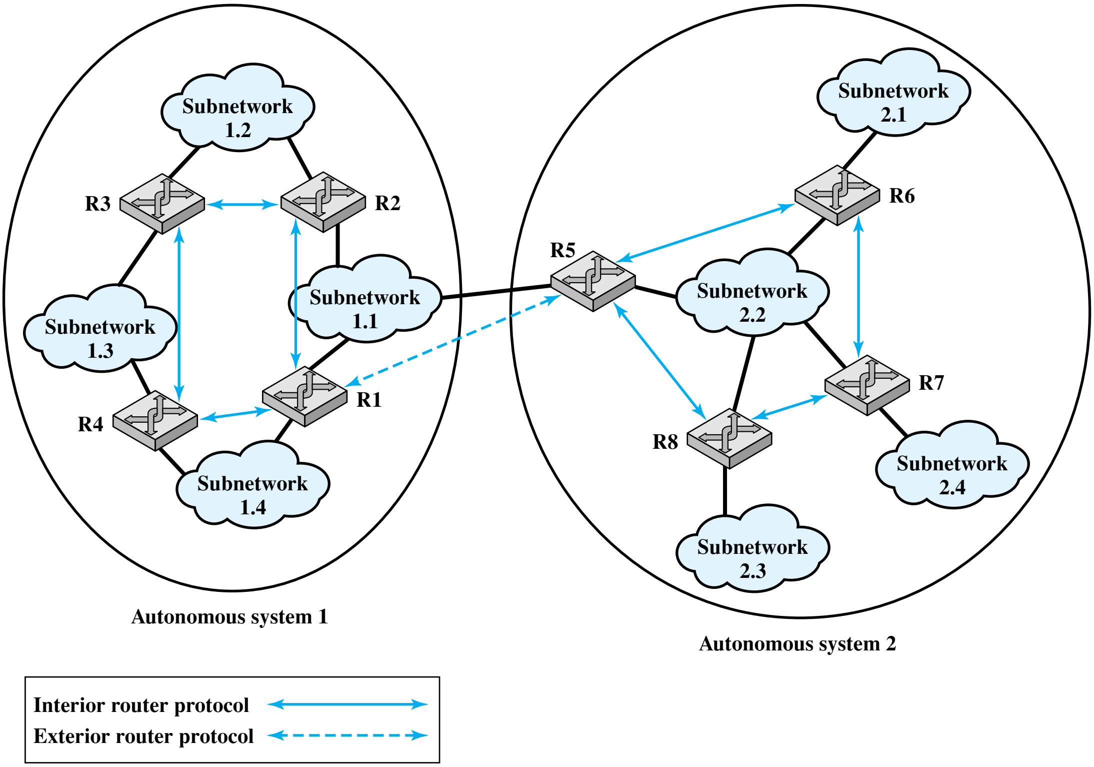


Figure 19.5 Application of Exterior and Interior Routing Protocols

We refer to the protocol used to pass routing information between routers in different ASs as an **exterior router protocol** (ERP).[2]

We can expect that an ERP will need to pass less information than an IRP, for the following reason. If a datagram is to be transferred from a host in one AS to a host in another AS, a router in the first system need only determine the target AS and devise a route to get into that target system. Once the datagram enters the target AS, the routers within that system can cooperate to deliver the datagram; the ERP is not concerned with, and does not know about, the details of the route followed within the target AS.

In the remainder of this section, we look at what are perhaps the most important examples of these two types of routing protocols: BGP and OSPF. But first, it is useful to look at a different way of characterizing routing protocols.

Approaches to Routing

Internet routing protocols employ one of three approaches to gathering and using routing information: distance-vector routing, link-state routing, and path-vector routing.

Distance-vector routing requires that each node (router or host that implements the routing protocol) exchange information with its neighboring nodes. Two nodes are said to be neighbors if they are both directly connected to the same network. This approach is that used in the first generation routing algorithm for ARPANET, as described in Section 12.2. For this purpose, each node maintains a vector of link costs for each directly attached network and distance and next-hop vectors for each destination. The relatively simple Routing Information Protocol (RIP) uses this approach.

Distance-vector routing requires the transmission of a considerable amount of information by each router. Each router must send a distance vector to all of its neighbors, and that vector contains the estimated path cost to all networks in the configuration. Furthermore, when there is a significant change in a link cost or when a link is unavailable, it may take a considerable amount of time for this information to propagate through the internet.

Link-state routing is designed to overcome the drawbacks of distance-vector routing. When a router is initialized, it determines the link cost on each of its network interfaces. The router then advertises this set of link costs to all other routers in the internet topology, not just neighboring routers. From then on, the router monitors its link costs. Whenever there is a significant change (a link cost increases or decreases substantially, a new link is created, an existing link becomes unavailable), the router again advertises its set of link costs to all other routers in the configuration.

Because each router receives the link costs of all routers in the configuration, each router can construct the topology of the entire configuration and then calculate the shortest path to each destination network. Having done this, the router can construct its routing table, listing the first hop to each destination. Because the

[2] In the literature, the terms *interior gateway protocol* (IGP) and *exterior gateway protocol* (EGP) are often used for what are referred to here as IRP and ERP. However, because the terms *IGP* and *EGP* also refer to specific protocols, we avoid their use to define the general concepts.

router has a representation of the entire network, it does not use a distributed version of a routing algorithm, as is done in distance-vector routing. Rather, the router can use any routing algorithm to determine the shortest paths. In practice, Dijkstra's algorithm is used. The Open Shortest Path First (OSPF) protocol is an example of a routing protocol that uses link-state routing. The second-generation routing algorithm for ARPANET also uses this approach.

Both link-state and distance-vector approaches have been used for interior router protocols. Neither approach is effective for an exterior router protocol.

In a distance-vector routing protocol, each router advertises to its neighbors a vector listing each network it can reach, together with a distance metric associated with the path to that network. Each router builds up a routing database on the basis of these neighbor updates but does not know the identity of intermediate routers and networks on any particular path. There are two problems with this approach for an exterior router protocol:

1. This distance-vector protocol assumes that all routers share a common distance metric with which to judge router preferences. This may not be the case among different ASs. If different routers attach different meanings to a given metric, it may not be possible to create stable, loop-free routes.
2. A given AS may have different priorities from other ASs and may have restrictions that prohibit the use of certain other AS. A distance-vector algorithm gives no information about the ASs that will be visited along a route.

In a link-state routing protocol, each router advertises its link metrics to all other routers. Each router builds up a picture of the complete topology of the configuration and then performs a routing calculation. This approach also has problems if used in an exterior router protocol:

1. Different ASs may use different metrics and have different restrictions. Although the link-state protocol does allow a router to build up a picture of the entire topology, the metrics used may vary from one AS to another, making it impossible to perform a consistent routing algorithm.
2. The flooding of link state information to all routers implementing an exterior router protocol across multiple ASs may be unmanageable.

An alternative, known as **path-vector routing**, is to dispense with routing metrics and simply provide information about which networks can be reached by a given router and the ASs that must be crossed to get there. The approach differs from a distance-vector algorithm in two respects: First, the path-vector approach does not include a distance or cost estimate. Second, each block of routing information lists all of the ASs visited in order to reach the destination network by this route.

Because a path vector lists the ASs that a datagram must traverse if it follows this route, the path information enables a router to perform policy routing. That is, a router may decide to avoid a particular path in order to avoid transiting a particular AS. For example, information that is confidential may be limited to certain kinds of ASs. Or a router may have information about the performance or quality of the portion of the internet that is included in an AS that leads the router to avoid that AS. Examples of performance or quality metrics include link speed, capacity, tendency

Table 19.2 BGP-4 Messages

Open	Used to open a neighbor relationship with another router.
Update	Used to (1) transmit information about a single route and/or (2) list multiple routes to be withdrawn.
Keepalive	Used to (1) acknowledge an Open message and (2) periodically confirm the neighbor relationship.
Notification	Send when an error condition is detected.

to become congested, and overall quality of operation. Another criterion that could be used is minimizing the number of transit ASs.

Border Gateway Protocol

The Border Gateway Protocol (BGP) was developed for use in conjunction with internets that employ the TCP/IP suite, although the concepts are applicable to any internet. BGP has become the preferred exterior router protocol for the Internet.

Functions BGP was designed to allow routers, called gateways in the standard, in different autonomous systems (ASs) to cooperate in the exchange of routing information. The protocol operates in terms of messages, which are sent over TCP connections. The repertoire of messages is summarized in Table 19.2. The current version of BGP is known as BGP-4 (RFC 1771).

Three functional procedures are involved in BGP:

- Neighbor acquisition
- Neighbor reachability
- Network reachability

Two routers are considered to be neighbors if they are attached to the same network. If the two routers are in different autonomous systems, they may wish to exchange routing information. For this purpose, it is necessary first to perform **neighbor acquisition**. In essence, neighbor acquisition occurs when two neighboring routers in different autonomous systems agree to exchange routing information regularly. A formal acquisition procedure is needed because one of the routers may not wish to participate. For example, the router may be overburdened and does not want to be responsible for traffic coming in from outside the system. In the neighbor acquisition process, one router sends a request message to the other, which may either accept or refuse the offer. The protocol does not address the issue of how one router knows the address or even the existence of another router, nor how it decides that it needs to exchange routing information with that particular router. These issues must be dealt with at configuration time or by active intervention of a network manager.

To perform neighbor acquisition, two routers send Open messages to each other after a TCP connection is established. If each router accepts the request, it returns a Keepalive message in response.

Once a neighbor relationship is established, the **neighbor reachability** procedure is used to maintain the relationship. Each partner needs to be assured that the other partner still exists and is still engaged in the neighbor relationship. For this purpose, the two routers periodically issue Keepalive messages to each other.

The final procedure specified by BGP is **network reachability**. Each router maintains a database of the networks that it can reach and the preferred route for reaching each network. When a change is made to this database, the router issues an Update message that is broadcast to all other routers implementing BGP. Because the Update message is broadcast, all BGP routers can build up and maintain their routing information.

BGP Messages Figure 19.6 illustrates the formats of all of the BGP messages. Each message begins with a 19-octet header containing three fields, as indicated by the shaded portion of each message in the figure:

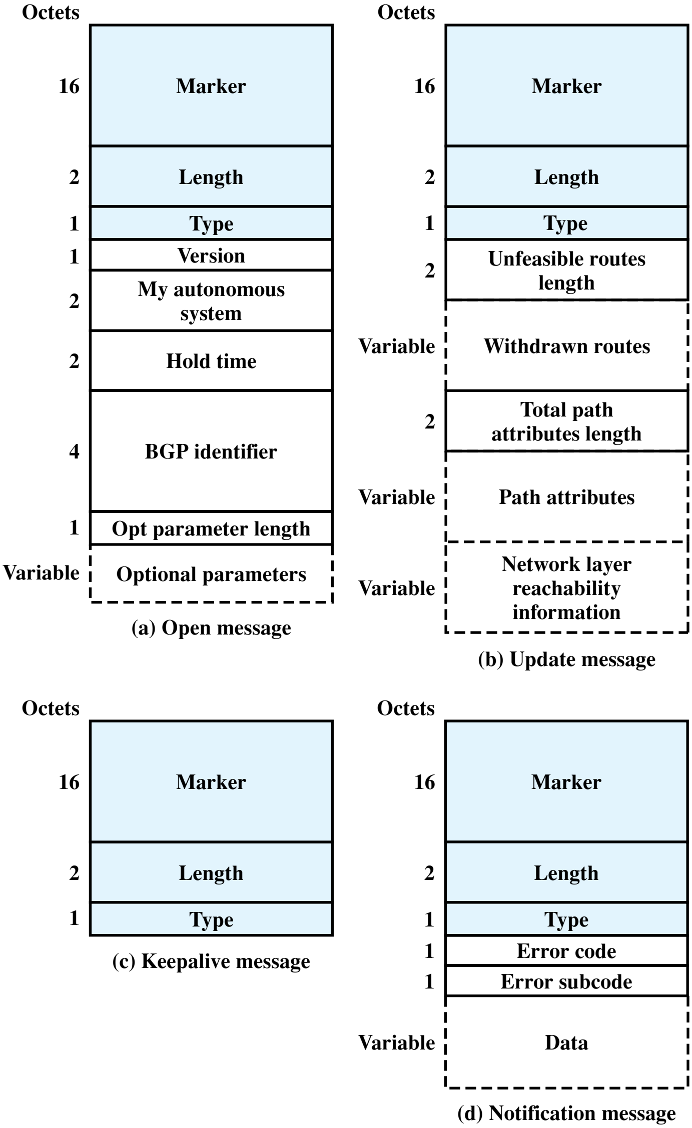


Figure 19.6 BGP Message Formats

- **Marker:** Reserved for authentication. The sender may insert a value in this field that would be used as part of an authentication mechanism to enable the recipient to verify the identity of the sender.
- **Length:** Length of message in octets.
- **Type:** Type of message: Open, Update, Notification, Keepalive.

To acquire a neighbor, a router first opens a TCP connection to the neighbor router of interest. It then sends an Open message. This message identifies the AS to which the sender belongs and provides the IP address of the router. It also includes a Hold Time parameter, which indicates the number of seconds that the sender proposes for the value of the Hold Timer. If the recipient is prepared to open a neighbor relationship, it calculates a value of Hold Timer that is the minimum of its Hold Time and the Hold Time in the Open message. This calculated value is the maximum number of seconds that may elapse between the receipt of successive Keepalive and/or Update messages by the sender.

The Keepalive message consists simply of the header. Each router issues these messages to each of its peers often enough to prevent the Hold Timer from expiring.

The Update message communicates two types of information:

- Information about a single route through the internet. This information is available to be added to the database of any recipient router.
- A list of routes previously advertised by this router that are being withdrawn.

An Update message may contain one or both types of information. Information about a single route through the network involves three fields: the Network Layer Reachability Information (NLRI) field, the Total Path Attributes Length field, and the Path Attributes field. The NLRI field consists of a list of identifiers of networks that can be reached by this route. Each network is identified by its IP address, which is actually a portion of a full IP address. Recall that an IP address is a 32-bit quantity of the form {network, host}. The left-hand or prefix portion of this quantity identifies a particular network.

The Path Attributes field contains a list of attributes that apply to this particular route. The following are the defined attributes:

- **Origin:** Indicates whether this information was generated by an interior router protocol (e.g., OSPF) or an exterior router protocol (in particular, BGP).
- **AS_Path:** A list of the ASs that are traversed for this route.
- **Next_Hop:** The IP address of the border router that should be used as the next hop to the destinations listed in the NLRI field.
- **Multi_Exit_Disc:** Used to communicate some information about routes internal to an AS. This is described later in this section.
- **Local_Pref:** Used by a router to inform other routers within the same AS of its degree of preference for a particular route. It has no significance to routers in other ASs.
- **Atomic_Aggregate, Aggregator:** These two fields implement the concept of route aggregation. In essence, an internet and its corresponding address space can be organized hierarchically (i.e., as a tree). In this case, network addresses

are structured in two or more parts. All of the networks of a given subtree share a common partial internet address. Using this common partial address, the amount of information that must be communicated in NLRI can be significantly reduced.

The AS_Path attribute actually serves two purposes. Because it lists the ASs that a datagram must traverse if it follows this route, the AS_Path information enables a router to implement routing policies. That is, a router may decide to avoid a particular path to avoid transiting a particular AS. For example, information that is confidential may be limited to certain kinds of ASs. Or a router may have information about the performance or quality of the portion of the internet that is included in an AS that leads the router to avoid that AS. Examples of performance or quality metrics include link speed, capacity, tendency to become congested, and overall quality of operation. Another criterion that could be used is minimizing the number of transit ASs.

The reader may wonder about the purpose of the Next_Hop attribute. The requesting router will necessarily want to know which networks are reachable via the responding router, but why provide information about other routers? This is best explained with reference to Figure 19.5. In this example, router R1 in autonomous system 1 and router R5 in autonomous system 2 implement BGP and acquire a neighbor relationship. R1 issues Update messages to R5, indicating which networks it can reach and the distances (network hops) involved. R1 also provides the same information on behalf of R2. That is, R1 tells R5 what networks are reachable via R2. In this example, R2 does not implement BGP. Typically, most of the routers in an autonomous system will not implement BGP. Only a few routers will be assigned responsibility for communicating with routers in other autonomous systems. A final point: R1 is in possession of the necessary information about R2, because R1 and R2 share an interior router protocol (IRP).

The second type of update information is the withdrawal of one or more routes. In this case, the route is identified by the IP address of the destination network.

Finally, the Notification Message is sent when an error condition is detected. The following errors may be reported:

- **Message header error:** Includes authentication and syntax errors.
- **Open message error:** Includes syntax errors and options not recognized in an Open message. This message can also be used to indicate that a proposed Hold Time in an Open message is unacceptable.
- **Update message error:** Includes syntax and validity errors in an Update message.
- **Hold timer expired:** If the sending router has not received successive Keepalive and/or Update and/or Notification messages within the Hold Time period, then this error is communicated and the connection is closed.
- **Finite state machine error:** Includes any procedural error.
- **Cease:** Used by a router to close a connection with another router in the absence of any other error.

BGP Routing Information Exchange The essence of BGP is the exchange of routing information among participating routers in multiple ASs. This process can be quite complex. In what follows, we provide a simplified overview.

Let us consider router R1 in autonomous system 1 (AS1), in Figure 19.5. To begin, a router that implements BGP will also implement an internal routing protocol such as OSPF. Using OSPF, R1 can exchange routing information with other routers within AS1 and build up a picture of the topology of the networks and routers in AS1 and construct a routing table. Next, R1 can issue an Update message to R5 in AS2. The Update message could include the following:

- **AS_Path:** The identity of AS1
- **Next_Hop:** The IP address of R1
- **NLRI:** A list of all of the networks in AS1

This message informs R5 that all of the networks listed in NLRI are reachable via R1 and that the only autonomous system traversed is AS1.

Suppose now that R5 also has a neighbor relationship with another router in another autonomous system, say R9 in AS3. R5 will forward the information just received from R1 to R9 in a new Update message. This message includes the following:

- **AS_Path:** The list of identifiers {AS2, AS1}
- **Next_Hop:** The IP address of R5
- **NLRI:** A list of all of the networks in AS1

This message informs R9 that all of the networks listed in NLRI are reachable via R5 and that the autonomous systems traversed are AS2 and AS1. R9 must now decide if this is its preferred route to the networks listed. It may have knowledge of an alternate route to some or all of these networks that it prefers for reasons of performance or some other policy metric. If R9 decides that the route provided in R5's update message is preferable, then R9 incorporates that routing information into its routing database and forwards this new routing information to other neighbors. This new message will include an AS_Path field of {AS3, AS2, AS1}.

In this fashion, routing update information is propagated through the larger internet, consisting of a number of interconnected autonomous systems. The AS_Path field is used to assure that such messages do not circulate indefinitely: If an Update message is received by a router in an AS that is included in the AS_Path field, that router will not forward the update information to other routers.

Routers within the same AS, called internal neighbors, may exchange BGP information. In this case, the sending router does not add the identifier of the common AS to the AS_Path field. When a router has selected a preferred route to an external destination, it transmits this route to all of its internal neighbors. Each of these routers then decides if the new route is preferred, in which case the new route is added to its database and a new Update message goes out.

When there are multiple entry points into an AS that are available to a border router in another AS, the Multi_Exit_Disc attribute may be used to choose among them. This attribute contains a number that reflects some internal metric for reaching destinations within an AS. For example, suppose in Figure 19.5 that both R1 and R2 implement BGP and both have a neighbor relationship with R5. Each provides an Update message to R5 for network 1.3 that includes a routing metric used internal to AS1, such as a routing metric associated with the OSPF internal router protocol. R5 could then use these two metrics as the basis for choosing between the two routes.

Open Shortest Path First (OSPF) Protocol

The OSPF protocol (RFC 2328) is now widely used as the interior router protocol in TCP/IP networks. OSPF computes a route through the internet that incurs the least cost based on a user-configurable metric of cost. The user can configure the cost to express a function of delay, data rate, dollar cost, or other factors. OSPF is able to equalize loads over multiple equal-cost paths.

Each router maintains a database that reflects the known topology of the autonomous system of which it is a part. The topology is expressed as a directed graph. The graph consists of the following:

- Vertices, or nodes, of two types:
  1. router
  2. network, which is in turn of two types
     a. transit, if it can carry data that neither originate nor terminate on an end system attached to this network
     b. stub, if it is not a transit network
- Edges of two types:
  1. graph edges that connect two router vertices when the corresponding routers are connected to each other by a direct point-to-point link
  2. graph edges that connect a router vertex to a network vertex when the router is directly connected to the network

Figure 19.7, based on one in RFC 2328, shows an example of an autonomous system, and Figure 19.8 is the resulting directed graph. The mapping is straightforward:

- Two routers joined by a point-to-point link are represented in the graph as being directly connected by a pair of edges, one in each direction (e.g., routers 6 and 10).
- When multiple routers are attached to a network (such as a LAN or packet-switching network), the directed graph shows all routers bidirectionally connected to the network vertex (e.g., routers 1, 2, 3, and 4 all connect to network 3).
- If a single router is attached to a network, the network will appear in the graph as a stub connection (e.g., network 7).
- An end system, called a host, can be directly connected to a router, in which case it is depicted in the corresponding graph (e.g., host 1).
- If a router is connected to other autonomous systems, then the path cost to each network in the other system must be obtained by some exterior router protocol (ERP). Each such network is represented on the graph by a stub and an edge to the router with the known path cost (e.g., networks 12 through 15).

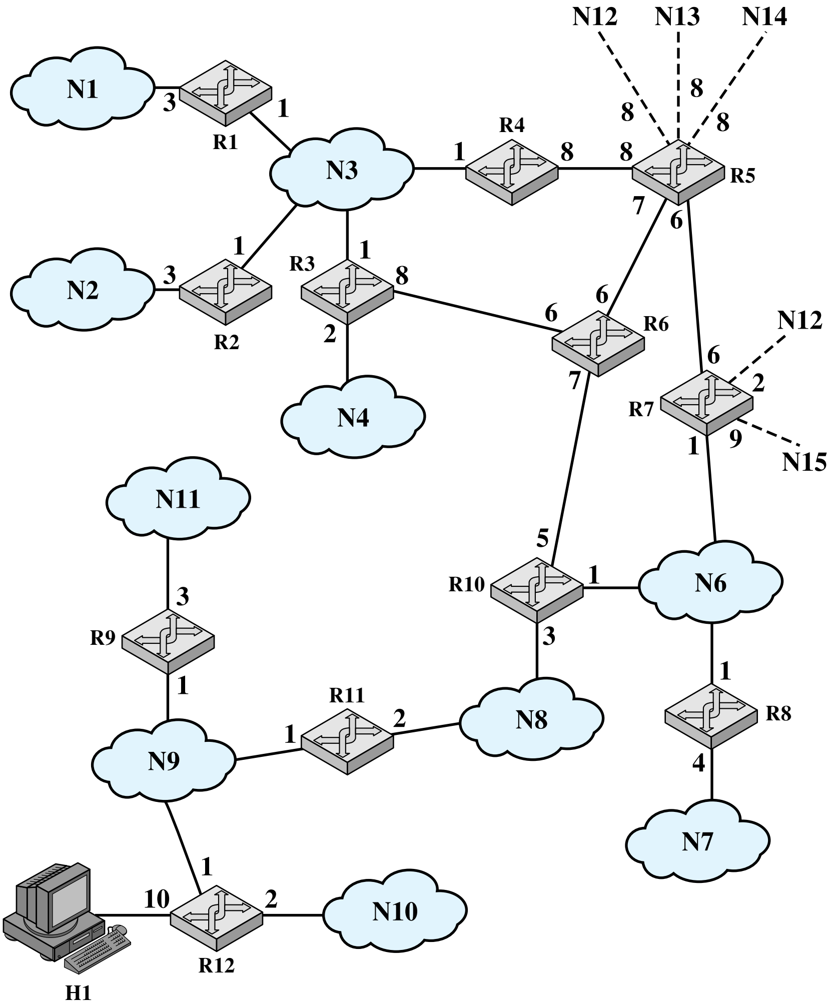


Figure 19.7 A Sample Autonomous System

A cost is associated with the output side of each router interface. This cost is configurable by the system administrator. Arcs on the graph are labeled with the cost of the corresponding router output interface. Arcs having no labeled cost have a cost of 0. Note that arcs leading from networks to routers always have a cost of 0.

A database corresponding to the directed graph is maintained by each router. It is pieced together from link state messages from other routers in the internet. Using Dijkstra's algorithm (see Section 12.3), a router calculates the least-cost path to all destination networks. The result for router 6 of Figure 19.7 is shown as a tree in Figure 19.9, with R6 as the root of the tree. The tree gives the entire route to any destination network or host. However, only the next hop to the destination is used in the forwarding process. The resulting routing table for router 6 is shown in Table 19.3. The table includes entries for routers advertising external routes (routers 5 and 7). For external networks whose identity is known, entries are also provided.

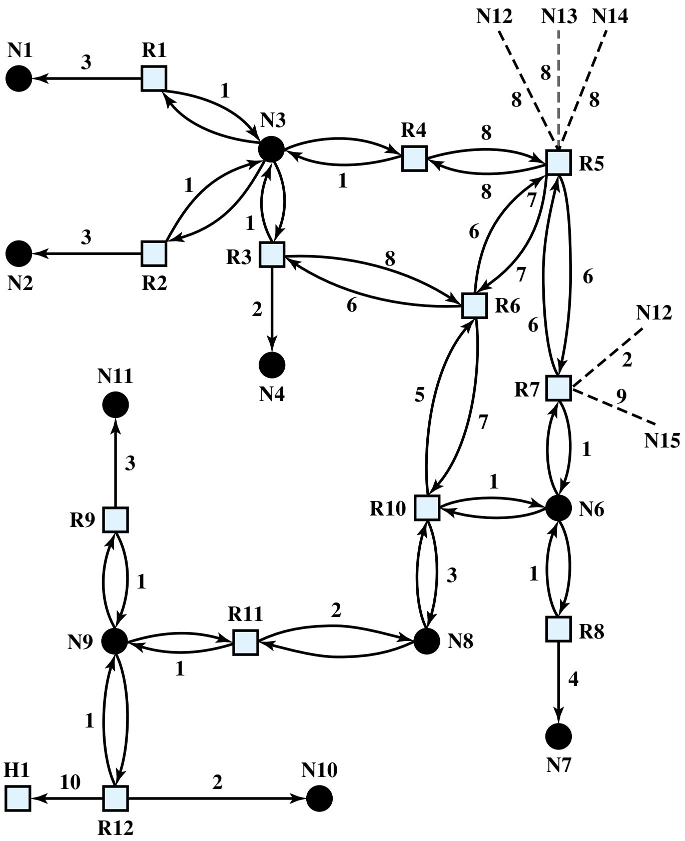


Figure 19.8 Directed Graph of Autonomous System of Figure 19.7

19.3 INTEGRATED SERVICES ARCHITECTURE

To meet the requirement for QoS-based service, the IETF is developing a suite of standards under the general umbrella of the Integrated Services Architecture (ISA). ISA, intended to provide QoS transport over IP-based internets, is defined in overall terms in RFC 1633, while a number of other documents are being developed to fill in the details. Already, a number of vendors have implemented portions of the ISA in routers and end-system software.

This section provides an overview of ISA.

Internet Traffic

Traffic on a network or internet can be divided into two broad categories: elastic and inelastic. A consideration of their differing requirements clarifies the need for an enhanced internet architecture.

Elastic Traffic Elastic traffic is that which can adjust, over wide ranges, to changes in delay and throughput across an internet and still meet the needs of its applications. This is the traditional type of traffic supported on TCP/IP-based internets and

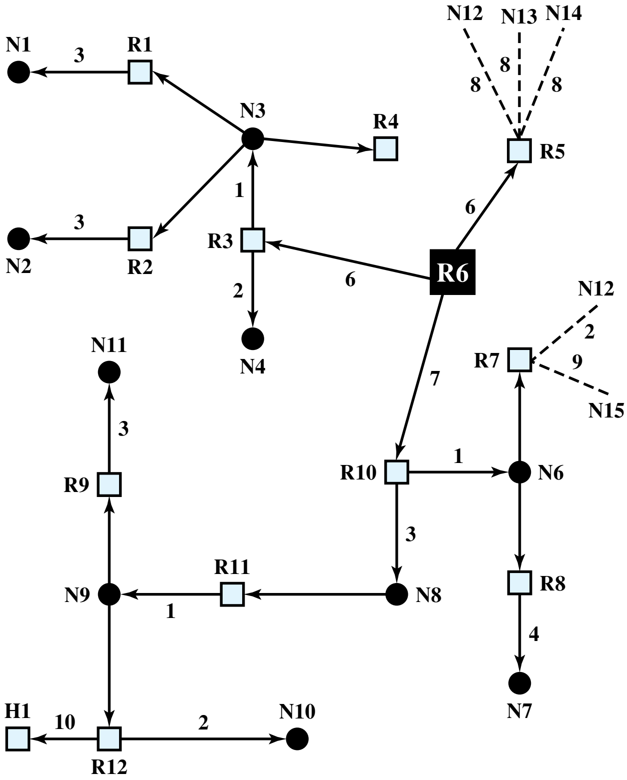


Figure 19.9 The SPF Tree for Router R6

is the type of traffic for which internets were designed. Applications that generate such traffic typically use TCP or UDP as a transport protocol. In the case of UDP, the application will use as much capacity as is available up to the rate that the application generates data. In the case of TCP, the application will use as much capacity as is available up to the maximum rate that the end-to-end receiver can accept data. Also with TCP, traffic on individual connections adjusts to congestion by reducing the rate at which data are presented to the network; this is described in Chapter 20.

Applications that can be classified as elastic include the common applications that operate over TCP or UDP, including file transfer (FTP), electronic mail (SMTP), remote login (TELNET), network management (SNMP), and Web access (HTTP). However, there are differences among the requirements of these applications. For example,

- E-mail is generally insensitive to changes in delay.
- When file transfer is done interactively, as it frequently is, the user expects the delay to be proportional to the file size and so is sensitive to changes in throughput.
- With network management, delay is generally not a serious concern. However, if failures in an internet are the cause of congestion, then the need for

Table 19.3 Routing Table for R6

Destination	Next Hop	Distance
N1	R3	10
N2	R3	10
N3	R3	7
N4	R3	8
N6	R10	8
N7	R10	12
N8	R10	10
N9	R10	11
N10	R10	13
N11	R10	14
H1	R10	21
R5	R5	6
R7	R10	8
N12	R10	10
N13	R5	14
N14	R5	14
N15	R10	17

SNMP messages to get through with minimum delay increases with increased congestion.

- Interactive applications, such as remote logon and Web access, are sensitive to delay.

It is important to realize that it is not per-packet delay that is the quantity of interest. As noted in [CLAR95], observation of real delays across the Internet suggest that wide variations in delay do not occur. Because of the congestion control mechanisms in TCP, when congestion develops, delays only increase modestly before the arrival rate from the various TCP connections slow down. Instead, the QoS perceived by the user relates to the total elapsed time to transfer an element of the current application. For an interactive TELNET-based application, the element may be a single keystroke or single line. For a Web access, the element is a Web page, which could be as little as a few kilobytes or could be substantially larger for an image-rich page. For a scientific application, the element could be many megabytes of data.

For very small elements, the total elapsed time is dominated by the delay time across the internet. However, for larger elements, the total elapsed time is dictated by the sliding-window performance of TCP and is therefore dominated by the throughput achieved over the TCP connection. Thus, for large transfers, the transfer time is proportional to the size of the file and the degree to which the source slows due to congestion.

It should be clear that even if we confine our attention to elastic traffic, a QoS-based internet service could be of benefit. Without such a service, routers are dealing evenhandedly with arriving IP packets, with no concern for the type of application and whether a particular packet is part of a large transfer element or a small one.

Under such circumstances, and if congestion develops, it is unlikely that resources will be allocated in such a way as to meet the needs of all applications fairly. When inelastic traffic is added to the mix, the results are even more unsatisfactory.

Inelastic Traffic Inelastic traffic does not easily adapt, if at all, to changes in delay and throughput across an internet. The prime example is real-time traffic. The requirements for inelastic traffic may include the following:

- **Throughput:** A minimum throughput value may be required. Unlike most elastic traffic, which can continue to deliver data with perhaps degraded service, many inelastic applications absolutely require a given minimum throughput.
- **Delay:** An example of a delay-sensitive application is stock trading; someone who consistently receives later service will consistently act later, and with greater disadvantage.
- **Jitter:** The magnitude of delay variation, called jitter, is a critical factor in real-time applications. Because of the variable delay imposed by the Internet, the interarrival times between packets are not maintained at a fixed interval at the destination. To compensate for this, the incoming packets are buffered, delayed sufficiently to compensate for the jitter, and then released at a constant rate to the software that is expecting a steady real-time stream. The larger the allowable delay variation, the longer the real delay in delivering the data and the greater the size of the delay buffer required at receivers. Real-time interactive applications, such as teleconferencing, may require a reasonable upper bound on jitter.
- **Packet loss:** Real-time applications vary in the amount of packet loss, if any, that they can sustain.

These requirements are difficult to meet in an environment with variable queuing delays and congestion losses. Accordingly, inelastic traffic introduces two new requirements into the internet architecture. First, some means is needed to give preferential treatment to applications with more demanding requirements. Applications need to be able to state their requirements, either ahead of time in some sort of service request function, or on the fly, by means of fields in the IP packet header. The former approach provides more flexibility in stating requirements, and it enables the network to anticipate demands and deny new requests if the required resources are unavailable. This approach implies the use of some sort of resource reservation protocol.

A second requirement in supporting inelastic traffic in an internet architecture is that elastic traffic must still be supported. Inelastic applications typically do not back off and reduce demand in the face of congestion, in contrast to TCP-based applications. Therefore, in times of congestion, inelastic traffic will continue to supply a high load, and elastic traffic will be crowded off the internet. A reservation protocol can help control this situation by denying service requests that would leave too few resources available to handle current elastic traffic.

ISA Approach

The purpose of ISA is to enable the provision of QoS support over IP-based internets. The central design issue for ISA is how to share the available capacity in times of congestion.

For an IP-based internet that provides only a best-effort service, the tools for controlling congestion and providing service are limited. In essence, routers have two mechanisms to work with:

- **Routing algorithm:** Most routing protocols in use in internets allow routes to be selected to minimize delay. Routers exchange information to get a picture of the delays throughout the internet. Minimum-delay routing helps to balance loads, thus decreasing local congestion, and helps to reduce delays seen by individual TCP connections.
- **Packet discard:** When a router's buffer overflows, it discards packets. Typically, the most recent packet is discarded. The effect of lost packets on a TCP connection is that the sending TCP entity backs off and reduces its load, thus helping to alleviate internet congestion.

These tools have worked reasonably well. However, as the discussion in the preceding subsection shows, such techniques are inadequate for the variety of traffic now coming to internets.

ISA is an overall architecture within which a number of enhancements to the traditional best-effort mechanisms are being developed. In ISA, each IP packet can be associated with a flow. RFC 1633 defines a flow as a distinguishable stream of related IP packets that results from a single user activity and requires the same QoS. For example, a flow might consist of one transport connection or one video stream distinguishable by the ISA. A flow differs from a TCP connection in two respects: A flow is unidirectional, and there can be more than one recipient of a flow (multicast). Typically, an IP packet is identified as a member of a flow on the basis of source and destination IP addresses and port numbers, and protocol type. The flow identifier in the IPv6 header is not necessarily equivalent to an ISA flow, but in future the IPv6 flow identifier could be used in ISA.

ISA makes use of the following functions to manage congestion and provide QoS transport:

- **Admission control:** For QoS transport (other than default best-effort transport), ISA requires that a reservation be made for a new flow. If the routers collectively determine that there are insufficient resources to guarantee the requested QoS, then the flow is not admitted. The protocol RSVP is used to make reservations.
- **Routing algorithm:** The routing decision may be based on a variety of QoS parameters, not just minimum delay. For example, the routing protocol OSPF, discussed in Section 19.2, can select routes based on QoS.
- **Queuing discipline:** A vital element of the ISA is an effective queuing policy that takes into account the differing requirements of different flows.
- **Discard policy:** A discard policy determines which packets to drop when a buffer is full and new packets arrive. A discard policy can be an important element in managing congestion and meeting QoS guarantees.

ISA Components

Figure 19.10 is a general depiction of the implementation architecture for ISA within a router. Below the thick horizontal line are the forwarding functions of the router; these are executed for each packet and therefore must be highly optimized.

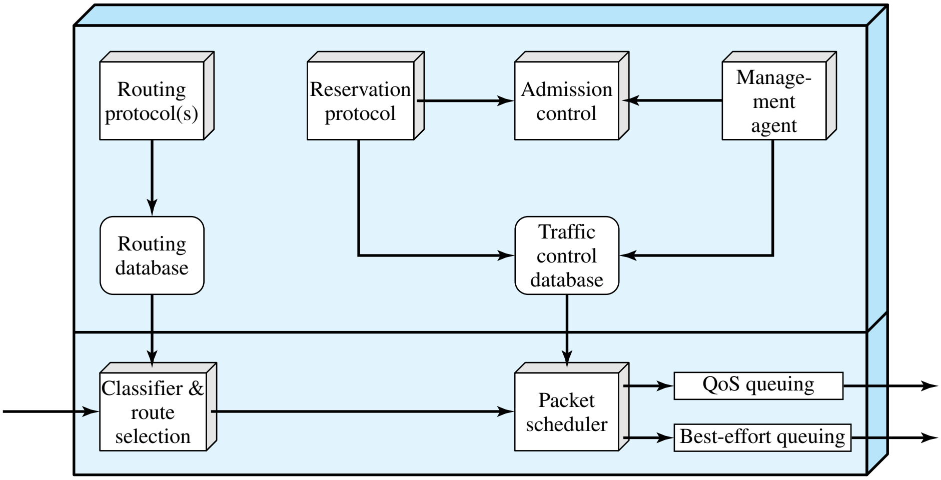


Figure 19.10 Integrated Services Architecture Implemented in Router

The remaining functions, above the line, are background functions that create data structures used by the forwarding functions.

The principal background functions are as follows:

- **Reservation protocol:** This protocol is to reserve resources for a new flow at a given level of QoS. It is used among routers and between routers and end systems. The reservation protocol is responsible for maintaining flow-specific state information at the end systems and at the routers along the path of the flow. RSVP is used for this purpose. The reservation protocol updates the traffic control database used by the packet scheduler to determine the service provided for packets of each flow.
- **Admission control:** When a new flow is requested, the reservation protocol invokes the admission control function. This function determines if sufficient resources are available for this flow at the requested QoS. This determination is based on the current level of commitment to other reservations and/or on the current load on the network.
- **Management agent:** A network management agent is able to modify the traffic control database and to direct the admission control module in order to set admission control policies.
- **Routing protocol:** The routing protocol is responsible for maintaining a routing database that gives the next hop to be taken for each destination address and each flow.

These background functions support the main task of the router, which is the forwarding of packets. The two principal functional areas that accomplish forwarding are the following:

- **Classifier and route selection:** For the purposes of forwarding and traffic control, incoming packets must be mapped into classes. A class may correspond to a single flow or to a set of flows with the same QoS requirements. For example, the packets of all video flows or the packets of all flows attributable to a particular organization may be treated identically for purposes of resource allocation and queuing discipline. The selection of class is based on fields in the IP header. Based on the packet's class and its destination IP address, this function determines the next-hop address for this packet.

- **Packet scheduler:** This function manages one or more queues for each output port. It determines the order in which queued packets are transmitted and the selection of packets for discard, if necessary. Decisions are made based on a packet's class, the contents of the traffic control database, and current and past activity on this outgoing port. Part of the packet scheduler's task is that of policing, which is the function of determining whether the packet traffic in a given flow exceeds the requested capacity and, if so, deciding how to treat the excess packets.

ISA Services

ISA service for a flow of packets is defined on two levels. First, a number of general categories of service are provided, each of which provides a certain general type of service guarantees. Second, within each category, the service for a particular flow is specified by the values of certain parameters; together, these values are referred to as a traffic specification (TSpec). Currently, three categories of service are defined:

- Guaranteed
- Controlled load
- Best effort

An application can request a reservation for a flow for a guaranteed or controlled load QoS, with a TSpec that defines the exact amount of service required. If the reservation is accepted, then the TSpec is part of the contract between the data flow and the service. The service agrees to provide the requested QoS as long as the flow's data traffic continues to be described accurately by the TSpec. Packets that are not part of a reserved flow are by default given a best-effort delivery service.

Before looking at the ISA service categories, one general concept should be defined: the token bucket traffic specification. This is a way of characterizing traffic that has three advantages in the context of ISA:

1. Many traffic sources can be defined easily and accurately by a token bucket scheme.
2. The token bucket scheme provides a concise description of the load to be imposed by a flow, enabling the service to determine easily the resource requirement.
3. The token bucket scheme provides the input parameters to a policing function.

A token bucket traffic specification consists of two parameters: a token replenishment rate R and a bucket size B. The token rate R specifies the continually sustainable data rate; that is, over a relatively long period of time, the average data rate to be supported for this flow is R. The bucket size B specifies the amount by which the data rate can exceed R for short periods of time. The exact condition is as follows: During any time period T, the amount of data sent cannot exceed $RT + B$.

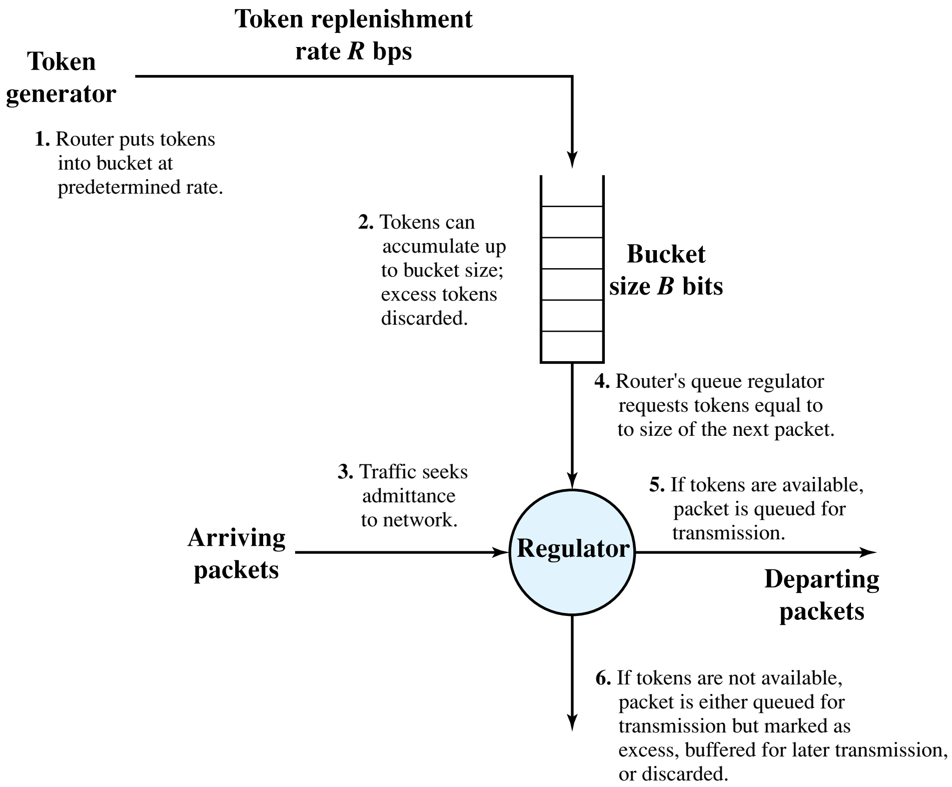

Token generator
Token replenishment rate *R* bps
1. Router puts tokens into bucket at predetermined rate.
2. Tokens can accumulate up to bucket size; excess tokens discarded.
Bucket size *B* bits
4. Router's queue regulator requests tokens equal to to size of the next packet.
3. Traffic seeks admittance to network.
Arriving packets
Regulator
5. If tokens are available, packet is queued for transmission.
Departing packets
6. If tokens are not available, packet is either queued for transmission but marked as excess, buffered for later transmission, or discarded.

Figure 19.11 Token Bucket Scheme

Figure 19.11 illustrates this scheme and explains the use of the term *bucket.* The bucket represents a counter that indicates the allowable number of octets of IP data that can be sent at any time. The bucket fills with *octet tokens* at the rate of *R* (i.e., the counter is incremented *R* times per second), up to the bucket capacity (up to the maximum counter value). IP packets arrive and are queued for processing. An IP packet may be processed if there are sufficient octet tokens to match the IP data size. If so, the packet is processed and the bucket is drained of the corresponding number of tokens. If a packet arrives and there are insufficient tokens available, then the packet exceeds the TSpec for this flow. The treatment for such packets is not specified in the ISA documents; common actions are relegating the packet to best-effort service, discarding the packet, or marking the packet in such a way that it may be discarded in future.

Over the long run, the rate of IP data allowed by the token bucket is *R*. However, if there is an idle or relatively slow period, the bucket capacity builds up, so that at most an additional *B* octets above the stated rate can be accepted. Thus, *B* is a measure of the degree of burstiness of the data flow that is allowed.

Guaranteed Service The key elements of the guaranteed service are as follows:

- The service provides assured capacity, or data rate.

- There is a specified upper bound on the queuing delay through the network. This must be added to the propagation delay, or latency, to arrive at the bound on total delay through the network.
- There are no queuing losses. That is, no packets are lost due to buffer overflow; packets may be lost due to failures in the network or changes in routing paths.

With this service, an application provides a characterization of its expected traffic profile, and the service determines the end-to-end delay that it can guarantee.

One category of applications for this service is those that need an upper bound on delay so that a delay buffer can be used for real-time playback of incoming data, and that do not tolerate packet losses because of the degradation in the quality of the output. Another example is applications with hard real-time deadlines.

The guaranteed service is the most demanding service provided by ISA. Because the delay bound is firm, the delay has to be set at a large value to cover rare cases of long queuing delays.

Controlled Load The key elements of the controlled load service are as follows:

- The service tightly approximates the behavior visible to applications receiving best-effort service under unloaded conditions.
- There is no specified upper bound on the queuing delay through the network. However, the service ensures that a very high percentage of the packets do not experience delays that greatly exceed the minimum transit delay (i.e., the delay due to propagation time plus router processing time with no queuing delays).
- A very high percentage of transmitted packets will be successfully delivered (i.e., almost no queuing loss).

As was mentioned, the risk in an internet that provides QoS for real-time applications is that best-effort traffic is crowded out. This is because best-effort types of applications employ TCP, which will back off in the face of congestion and delays. The controlled load service guarantees that the network will set aside sufficient resources so that an application that receives this service will see a network that responds as if these real-time applications were not present and competing for resources.

The controlled service is useful for applications that have been referred to as adaptive real-time applications [CLAR92]. Such applications do not require an a priori upper bound on the delay through the network. Rather, the receiver measures the jitter experienced by incoming packets and sets the playback point to the minimum delay that still produces a sufficiently low loss rate (e.g., video can be adaptive by dropping a frame or delaying the output stream slightly; voice can be adaptive by adjusting silent periods).

Queuing Discipline

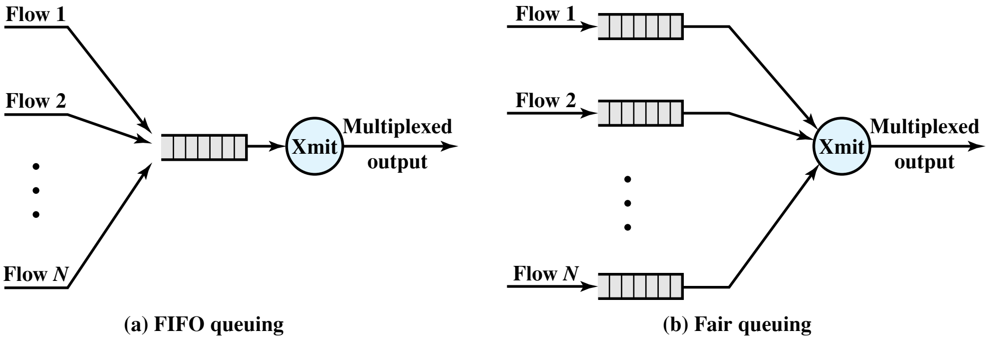
An important component of an ISA implementation is the queuing discipline used at the routers. Routers traditionally have used a first-in-first-out (FIFO) queuing discipline at each output port. A single queue is maintained at each output port. When a new packet arrives and is routed to an output port, it is placed at the end of the queue. As long as the queue is not empty, the router transmits packets from the queue, taking the oldest remaining packet next.

There are several drawbacks to the FIFO queuing discipline:

- No special treatment is given to packets from flows that are of higher priority or are more delay sensitive. If a number of packets from different flows are ready to be forwarded, they are handled strictly in FIFO order.
- If a number of smaller packets are queued behind a long packet, then FIFO queuing results in a larger average delay per packet than if the shorter packets were transmitted before the longer packet. In general, flows of larger packets get better service.
- A greedy TCP connection can crowd out more altruistic connections. If congestion occurs and one TCP connection fails to back off, other connections along the same path segment must back off more than they would otherwise have to do.

To overcome the drawbacks of FIFO queuing, some sort of fair queuing scheme is used, in which a router maintains multiple queues at each output port (Figure 19.12). With simple fair queuing, each incoming packet is placed in the queue for its flow. The queues are serviced in round-robin fashion, taking one packet from each nonempty queue in turn. Empty queues are skipped over. This scheme is fair in that each busy flow gets to send exactly one packet per cycle. Further, this is a form of load balancing among the various flows. There is no advantage in being greedy. A greedy flow finds that its queues become long, increasing its delays, whereas other flows are unaffected by this behavior.

A number of vendors have implemented a refinement of fair queuing known as weighted fair queuing (WFQ). In essence, WFQ takes into account the amount of traffic through each queue and gives busier queues more capacity without completely shutting out less busy queues. In addition, WFQ can take into account the amount of service requested by each traffic flow and adjust the queuing discipline accordingly.



(a) FIFO queuing

(b) Fair queuing

Figure 19.12 FIFO and Fair Queuing

Resource ReSerVation Protocol (RSVP)

RFC 2205 defines RSVP, which provides supporting functionality for ISA. This subsection provides an overview.

A key task, perhaps the key task, of an internetwork is to deliver data from a source to one or more destinations with the desired quality of service (QoS), such as throughput, delay, delay variance, and so on. This task becomes increasingly difficult on any internetwork with increasing number of users, data rate of applications, and use of multicasting. To meet these needs, it is not enough for an internet to react to congestion. Instead a tool is needed to prevent congestion by allowing applications to reserve network resources at a given QoS.

Preventive measures can be useful in both unicast and multicast transmission. For **unicast**, two applications agree on a specific quality of service for a session and expect the internetwork to support that quality of service. If the internetwork is heavily loaded, it may not provide the desired QoS and instead deliver packets at a reduced QoS. In that case, the applications may have preferred to wait before initiating the session or at least to have been alerted to the potential for reduced QoS. A way of dealing with this situation is to have the unicast applications reserve resources in order to meet a given quality of service. Routers along an intended path could then preallocate resources (queue space, outgoing capacity) to assure the desired QoS. If a router could not meet the resource reservation because of prior outstanding reservations, then the applications could be informed. The applications may then decide to try again at a reduced QoS reservation or may decide to try later.

Multicast transmission presents a much more compelling case for implementing resource reservation. A multicast transmission can generate a tremendous amount of internetwork traffic if either the application is high-volume (e.g., video) or the group of multicast destinations is large and scattered, or both. What makes the case for multicast resource reservation is that much of the potential load generated by a multicast source may easily be prevented. This is so for two reasons:

1. Some members of an existing multicast group may not require delivery from a particular source over some given period of time. For example, there may be two "channels" (two multicast sources) broadcasting to a particular multicast group at the same time. A multicast destination may wish to "tune in" to only one channel at a time.
2. Some members of a group may only be able to handle a portion of the source transmission. For example, a video source may transmit a video stream that consists of two components: a basic component that provides a reduced picture quality, and an enhanced component. Some receivers may not have the processing power to handle the enhanced component or may be connected to the internetwork through a subnetwork or link that does not have the capacity for the full signal.

Thus, the use of resource reservation can enable routers to decide ahead of time if they can meet the requirement to deliver a multicast transmission to all designated multicast receivers and to reserve the appropriate resources if possible.

Internet resource reservation differs from the type of resource reservation that may be implemented in a connection-oriented network, such as ATM or frame relay.

An internet resource reservation scheme must interact with a dynamic routing strategy that allows the route followed by packets of a given transmission to change. When the route changes, the resource reservations must be changed. To deal with this dynamic situation, the concept of **soft state** is used. A soft state is simply a set of state information at a router that expires unless regularly refreshed from the entity that requested the state. If a route for a given transmission changes, then some soft states will expire and new resource reservations will invoke the appropriate soft states on the new routers along the route. Thus, the end systems requesting resources must periodically renew their requests during the course of an application transmission.

Based on these considerations, the specification lists the following characteristics of RSVP:

- **Unicast and multicast:** RSVP makes reservations for both unicast and multicast transmissions, adapting dynamically to changing group membership as well as to changing routes, and reserving resources based on the individual requirements of multicast members.
- **Simplex:** RSVP makes reservations for unidirectional data flow. Data exchanges between two end systems require separate reservations in the two directions.
- **Receiver-initiated reservation:** The receiver of a data flow initiates and maintains the resource reservation for that flow.
- **Maintaining soft state in the internet:** RSVP maintains a soft state at intermediate routers and leaves the responsibility for maintaining these reservation states to end users.
- **Providing different reservation styles:** These allow RSVP users to specify how reservations for the same multicast group should be aggregated at the intermediate switches. This feature enables a more efficient use of internet resources.
- **Transparent operation through non-RSVP routers:** Because reservations and RSVP are independent of routing protocol, there is no fundamental conflict in a mixed environment in which some routers do not employ RSVP. These routers will simply use a best-effort delivery technique.
- **Support for IPv4 and IPv6:** RSVP can exploit the Type-of-Service field in the IPv4 header and the Flow Label field in the IPv6 header.

19.4 DIFFERENTIATED SERVICES

The Integrated Services Architecture (ISA) and RSVP are intended to support QoS capability in the Internet and in private internets. Although ISA in general and RSVP in particular are useful tools in this regard, these features are relatively complex to deploy. Further, they may not scale well to handle large volumes of traffic because of the amount of control signaling required to coordinate integrated QoS offerings and because of the maintenance of state information required at routers.

As the burden on the Internet grows, and as the variety of applications grow, there is an immediate need to provide differing levels of QoS to different traffic flows. The differentiated services (DS) architecture (RFC 2475) is designed to provide a simple, easy-to-implement, low-overhead tool to support a range of network services that are differentiated on the basis of performance.

Several key characteristics of DS contribute to its efficiency and ease of deployment:

- IP packets are labeled for differing QoS treatment using the existing IPv4 (Figure 18.6) or IPv6 (Figure 18.11) DS field. Thus, no change is required to IP.
- A service level agreement (SLA) is established between the service provider (internet domain) and the customer prior to the use of DS. This avoids the need to incorporate DS mechanisms in applications. Thus, existing applications need not be modified to use DS.
- DS provides a built-in aggregation mechanism. All traffic with the same DS octet is treated the same by the network service. For example, multiple voice connections are not handled individually but in the aggregate. This provides for good scaling to larger networks and traffic loads.
- DS is implemented in individual routers by queuing and forwarding packets based on the DS octet. Routers deal with each packet individually and do not have to save state information on packet flows.

Today, DS is the most widely accepted QoS mechanism in enterprise networks.

Although DS is intended to provide a simple service based on relatively simple mechanisms, the set of RFCs related to DS is relatively complex. Table 19.4 summarizes some of the key terms from these specifications.

Services

The DS type of service is provided within a DS domain, which is defined as a contiguous portion of the Internet over which a consistent set of DS policies are administered. Typically, a DS domain would be under the control of one administrative entity. The services provided across a DS domain are defined in an SLA, which is a service contract between a customer and the service provider that specifies the forwarding service that the customer should receive for various classes of packets. A customer may be a user organization or another DS domain. Once the SLA is established, the customer submits packets with the DS octet marked to indicate the packet class. The service provider must assure that the customer gets at least the agreed QoS for each packet class. To provide that QoS, the service provider must configure the appropriate forwarding policies at each router (based on DS octet value) and must measure the performance being provided for each class on an ongoing basis.

If a customer submits packets intended for destinations within the DS domain, then the DS domain is expected to provide the agreed service. If the destination is beyond the customer's DS domain, then the DS domain will attempt to forward the packets through other domains, requesting the most appropriate service to match the requested service.

A draft DS framework document lists the following detailed performance parameters that might be included in an SLA:

- Detailed service performance parameters such as expected throughput, drop probability, latency

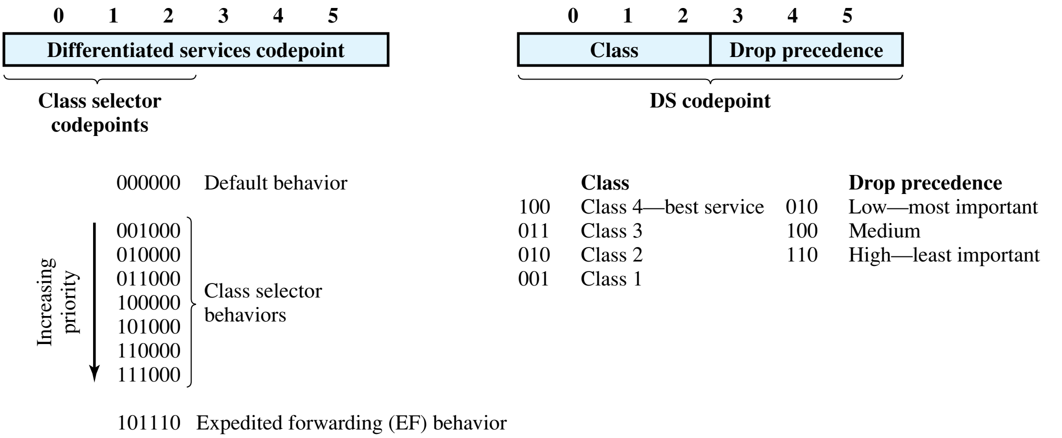
Table 19.4 Terminology for Differentiated Services

Behavior Aggregate	A set of packets with the same DS codepoint crossing a link in a particular direction.
Classifier	Selects packets based on the DS field (BA classifier) or on multiple fields within the packet header (MF classifier).
DS Boundary Node	A DS node that connects one DS domain to a node in another domain.
DS Codepoint	A specified value of the 6-bit DSCP portion of the 8-bit DS field in the IP header.
DS Domain	A contiguous (connected) set of nodes, capable of implementing differentiated services, that operate with a common set of service provisioning policies and per-hop behavior definitions.
DS Interior Node	A DS node that is not a DS boundary node.
DS Node	A node that supports differentiated services. Typically, a DS node is a router. A host system that provides differentiated services for applications in the host is also a DS node.
Dropping	The process of discarding packets based on specified rules; also called **policing**.
Marking	The process of setting the DS codepoint in a packet. Packets may be marked on initiation and may be re-marked by an en route DS node.
Metering	The process of measuring the temporal properties (e.g., rate) of a packet stream selected by a classifier. The instantaneous state of that process may affect marking, shaping, and dropping functions.
Per-Hop Behavior (PHB)	The externally observable forwarding behavior applied at a node to a behavior aggregate.
Service Level Agreement (SLA)	A service contract between a customer and a service provider that specifies the forwarding service a customer should receive.
Shaping	The process of delaying packets within a packet stream to cause it to conform to some defined traffic profile.
Traffic Conditioning	Control functions performed to enforce rules specified in a TCA, including metering, marking, shaping, and dropping.
Traffic Conditioning Agreement (TCA)	An agreement specifying classifying rules and traffic conditioning rules that are to apply to packets selected by the classifier.

- Constraints on the ingress and egress points at which the service is provided, indicating the scope of the service
- Traffic profiles that must be adhered to for the requested service to be provided, such as token bucket parameters
- Disposition of traffic submitted in excess of the specified profile

The framework document also gives some examples of services that might be provided:

1. Traffic offered at service level A will be delivered with low latency.
2. Traffic offered at service level B will be delivered with low loss.
3. Ninety percent of in-profile traffic delivered at service level C will experience no more than 50 ms latency.
4. Ninety-five percent of in-profile traffic delivered at service level D will be delivered.
5. Traffic offered at service level E will be allotted twice the bandwidth of traffic delivered at service level F.



(a) DS Field

(b) Codepoints for assured forwarding PHB

Figure 19.13 DS Field

6. Traffic with drop precedence X has a higher probability of delivery than traffic with drop precedence Y.

The first two examples are qualitative and are valid only in comparison to other traffic, such as default traffic that gets a best-effort service. The next two examples are quantitative and provide a specific guarantee that can be verified by measurement on the actual service without comparison to any other services offered at the same time. The final two examples are a mixture of quantitative and qualitative.

DS Field

Packets are labeled for service handling by means of the 6-bit DS field in the IPv4 header or the IPv6 header. The value of the DS field, referred to as the **DS codepoint**, is the label used to classify packets for differentiated services. Figure 19.13a shows the DS field.

With a 6-bit codepoint, there are in principle 64 different classes of traffic that could be defined. These 64 codepoints are allocated across three pools of codepoints, as follows:

- Codepoints of the form xxxxx0, where x is either 0 or 1, are reserved for assignment as standards.
- Codepoints of the form xxxx11 are reserved for experimental or local use.
- Codepoints of the form xxxx01 are also reserved for experimental or local use but may be allocated for future standards action as needed.

Within the first pool, several assignments are made in RFC 2474. The codepoint 000000 is the default packet class. The default class is the best-effort forwarding behavior in existing routers. Such packets are forwarded in the order that they are received as soon as link capacity becomes available. If other higher-priority packets in other DS classes are available for transmission, these are given preference over best-effort default packets.

Codepoints of the form xxx000 are reserved to provide backward compatibility with the IPv4 precedence service. To explain this requirement, we need to digress to an explanation of the IPv4 precedence service. The IPv4 type of service (TOS) field includes two subfields: a 3-bit precedence subfield and a 4-bit TOS subfield. These subfields serve complementary functions. The TOS subfield provides guidance to the IP entity (in the source or router) on selecting the next hop for this datagram, and the precedence subfield provides guidance about the relative allocation of router resources for this datagram.

The precedence field is set to indicate the degree of urgency or priority to be associated with a datagram. If a router supports the precedence subfield, there are three approaches to responding:

- **Route selection:** A particular route may be selected if the router has a smaller queue for that route or if the next hop on that route supports network precedence or priority (e.g., a token ring network supports priority).
- **Network service:** If the network on the next hop supports precedence, then that service is invoked.
- **Queuing discipline:** A router may use precedence to affect how queues are handled. For example, a router may give preferential treatment in queues to datagrams with higher precedence.

RFC 1812, Requirements for IP Version 4 Routers, provides recommendations for queuing discipline that fall into two categories:

- **Queue service**
  - **(a)** Routers SHOULD implement precedence-ordered queue service. Precedence-ordered queue service means that when a packet is selected for output on a (logical) link, the packet of highest precedence that has been queued for that link is sent.
  - **(b)** Any router MAY implement other policy-based throughput management procedures that result in other than strict precedence ordering, but it MUST be configurable to suppress them (i.e., use strict ordering).
- **Congestion control.** When a router receives a packet beyond its storage capacity, it must discard it or some other packet or packets.
  - **(a)** A router MAY discard the packet it has just received; this is the simplest but not the best policy.
  - **(b)** Ideally, the router should select a packet from one of the sessions most heavily abusing the link, given that the applicable QoS policy permits this. A recommended policy in datagram environments using FIFO queues is to discard a packet randomly selected from the queue. An equivalent algorithm in routers using fair queues is to discard from the longest queue. A router MAY use these algorithms to determine which packet to discard.
  - **(c)** If precedence-ordered queue service is implemented and enabled, the router MUST NOT discard a packet whose IP precedence is higher than that of a packet that is not discarded.

(d) A router MAY protect packets whose IP headers request the maximize reliability TOS, except where doing so would be in violation of the previous rule.

(e) A router MAY protect fragmented IP packets, on the theory that dropping a fragment of a datagram may increase congestion by causing all fragments of the datagram to be retransmitted by the source.

(f) To help prevent routing perturbations or disruption of management functions, the router MAY protect packets used for routing control, link control, or network management from being discarded. Dedicated routers (i.e., routers that are not also general purpose hosts, terminal servers, etc.) can achieve an approximation of this rule by protecting packets whose source or destination is the router itself.

The DS codepoints of the form xxx000 should provide a service that at minimum is equivalent to that of the IPv4 precedence functionality.

DS Configuration and Operation

Figure 19.14 illustrates the type of configuration envisioned in the DS documents. A DS domain consists of a set of contiguous routers; that is, it is possible to get from any router in the domain to any other router in the domain by a path that does not include routers outside the domain. Within a domain, the interpretation of DS codepoints is uniform, so that a uniform, consistent service is provided.

Routers in a DS domain are either boundary nodes or interior nodes. Typically, the interior nodes implement simple mechanisms for handling packets based

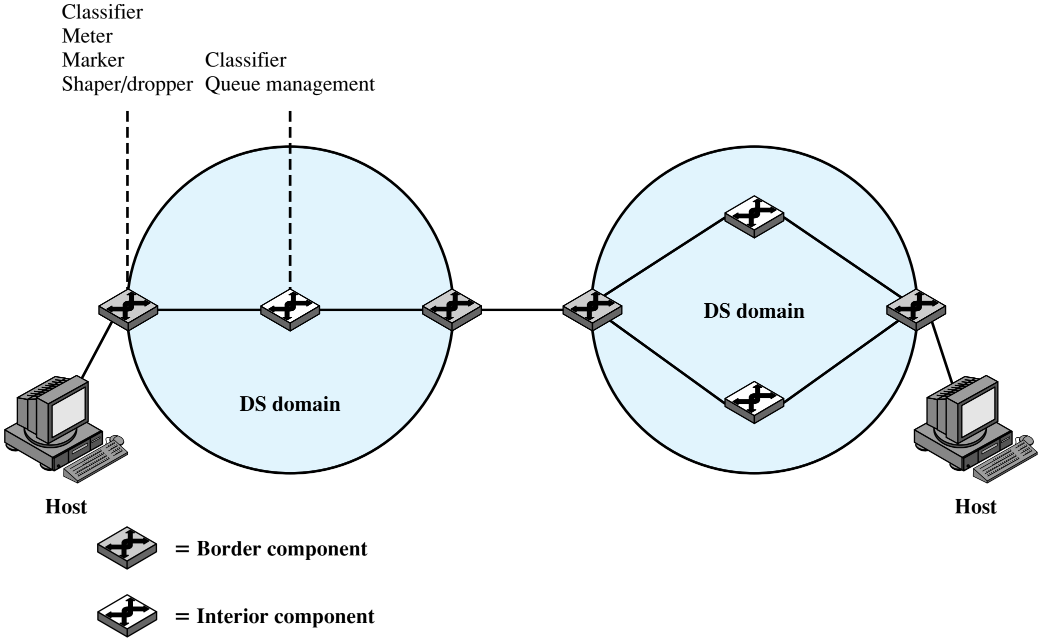


Figure 19.14 DS Domains

on their DS codepoint values. This includes queuing discipline to give preferential treatment depending on codepoint value, and packet-dropping rules to dictate which packets should be dropped first in the event of buffer saturation. The DS specifications refer to the forwarding treatment provided at a router as per-hop behavior (PHB). This PHB must be available at all routers, and typically PHB is the only part of DS implemented in interior routers.

The boundary nodes include PHB mechanisms but more sophisticated traffic conditioning mechanisms are also required to provide the desired service. Thus, interior routers have minimal functionality and minimal overhead in providing the DS service, while most of the complexity is in the boundary nodes. The boundary node function can also be provided by a host system attached to the domain, on behalf of the applications at that host system.

The traffic conditioning function consists of five elements:

- **Classifier:** Separates submitted packets into different classes. This is the foundation of providing differentiated services. A classifier may separate traffic only on the basis of the DS codepoint (behavior aggregate classifier) or based on multiple fields within the packet header or even the packet payload (multifield classifier).
- **Meter:** Measures submitted traffic for conformance to a profile. The meter determines whether a given packet stream class is within or exceeds the service level guaranteed for that class.
- **Marker:** Re-marks packets with a different codepoint as needed. This may be done for packets that exceed the profile; for example, if a given throughput is guaranteed for a particular service class, any packets in that class that exceed the throughput in some defined time interval may be re-marked for best effort handling. Also, re-marking may be required at the boundary between two DS domains. For example, if a given traffic class is to receive the highest supported priority, and this is a value of 3 in one domain and 7 in the next domain, then packets with a priority 3 value traversing the first domain are remarked as priority 7 when entering the second domain.
- **Shaper:** Delays packets as necessary so that the packet stream in a given class does not exceed the traffic rate specified in the profile for that class.
- **Dropper:** Drops packets when the rate of packets of a given class exceeds that specified in the profile for that class.

Figure 19.15 illustrates the relationship between the elements of traffic conditioning. After a flow is classified, its resource consumption must be measured. The metering function measures the volume of packets over a particular time interval to determine a flow's compliance with the traffic agreement. If the host is bursty, a simple data rate or packet rate may not be sufficient to capture the desired traffic characteristics. A token bucket scheme, such as that illustrated in Figure 19.11, is an example of a way to define a traffic profile to take into account both packet rate and burstiness.

If a traffic flow exceeds some profile, several approaches can be taken. Individual packets in excess of the profile may be re-marked for lower-quality handling and allowed to pass into the DS domain. A traffic shaper may absorb a burst of

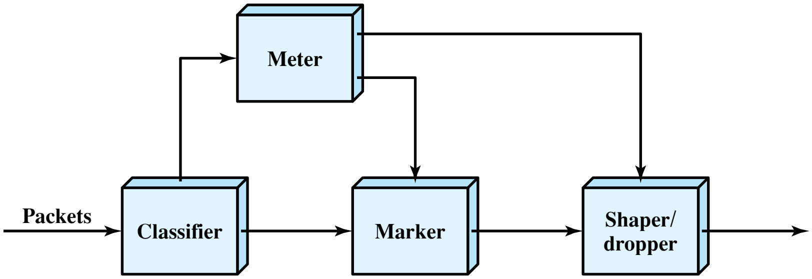


Figure 19.15 DS Traffic Conditioner

packets in a buffer and pace the packets over a longer period of time. A dropper may drop packets if the buffer used for pacing becomes saturated.

Per-Hop Behavior

As part of the DS standardization effort, specific types of PHB need to be defined, which can be associated with specific differentiated services. Currently, two standards-track PHBs have been issued: expedited forwarding PHB (RFCs 3246 and 3247) and assured forwarding PHB (RFC 2597).

Expedited Forwarding PHB RFC 3246 defines the expedited forwarding (EF) PHB as a building block for low-loss, low-delay, and low-jitter end-to-end services through DS domains. In essence, such a service should appear to the endpoints as providing close to the performance of a point-to-point connection or leased line.

In an internet or packet-switching network, a low-loss, low-delay, and low-jitter service is difficult to achieve. By its nature, an internet involves queues at each node, or router, where packets are buffered waiting to use a shared output link. It is the queuing behavior at each node that results in loss, delays, and jitter. Thus, unless the internet is grossly oversized to eliminate all queuing effects, care must be taken in handling traffic for EF PHB to assure that queuing effects do not result in loss, delay, or jitter above a given threshold. RFC 3246 declares that the intent of the EF PHB is to provide a PHB in which suitably marked packets usually encounter short or empty queues. The relative absence of queues minimizes delay and jitter. Furthermore, if queues remain short relative to the buffer space available, packet loss is also kept to a minimum.

The EF PHB is designed to configuring nodes so that the traffic aggregate[3] has a well-defined minimum departure rate. (*Well-defined* means "independent of the dynamic state of the node." In particular, independent of the intensity of other traffic at the node.) The general concept outlined in RFC 3246 is this: The border nodes control the traffic aggregate to limit its characteristics (rate, burstiness) to some predefined level. Interior nodes must treat the incoming traffic in such a way that queuing effects do not appear. In general terms, the requirement on interior nodes is that the aggregate's maximum arrival rate must be less than the aggregate's minimum departure rate.

[3] The term *traffic aggregate* refers to the flow of packets associated with a particular service for a particular user.

RFC 3246 does not mandate a specific queuing policy at the interior nodes to achieve the EF PHB. The RFC notes that a simple priority scheme could achieve the desired effect, with the EF traffic given absolute priority over other traffic. So long as the EF traffic itself did not overwhelm an interior node, this scheme would result in acceptable queuing delays for the EF PHB. However, the risk of a simple priority scheme is that packet flows for other PHB traffic would be disrupted. Thus, some more sophisticated queuing policy might be warranted.

Assured Forwarding PHB The assured forwarding (AF) PHB is designed to provide a service superior to best-effort but one that does not require the reservation of resources within an internet and does not require the use of detailed discrimination among flows from different users. The concept behind the AF PHB was first introduced in [CLAR98] and is referred to as explicit allocation. The AF PHB is more complex than explicit allocation, but it is useful to first highlight the key elements of the explicit allocation scheme:

1. Users are offered the choice of a number of classes of service for their traffic. Each class describes a different traffic profile in terms of an aggregate data rate and burstiness.
2. Traffic from a user within a given class is monitored at a boundary node. Each packet in a traffic flow is marked *in* or *out* based on whether it does or does not exceed the traffic profile.
3. Inside the network, there is no separation of traffic from different users or even traffic from different classes. Instead, all traffic is treated as a single pool of packets, with the only distinction being whether each packet has been marked *in* or *out.*
4. When congestion occurs, the interior nodes implement a dropping scheme in which *out* packets are dropped before *in* packets.
5. Different users will see different levels of service because they will have different quantities of *in* packets in the service queues.

The advantage of this approach is its simplicity. Very little work is required by the internal nodes. Marking of the traffic at the boundary nodes based on traffic profiles provides different levels of service to different classes.

The AF PHB defined in RFC 2597 expands on the preceding approach in the following ways:

1. Four AF classes are defined, allowing the definition of four distinct traffic profiles. A user may select one or more of these classes to satisfy requirements.
2. Within each class, packets are marked by the customer or by the service provider with one of three drop precedence values. In case of congestion, the drop precedence of a packet determines the relative importance of the packet within the AF class. A congested DS node tries to protect packets with a lower drop precedence value from being lost by preferably discarding packets with a higher drop precedence value.

This approach is still simpler to implement than any sort of resource reservation scheme but provides considerable flexibility. Within an interior DS node, traffic from the four classes can be treated separately, with different amounts of resources (buffer space, data rate) assigned to the four classes. Within each class, packets are handled based on drop precedence. Thus, as RFC 2597 points out, the level of forwarding assurance of an IP packet depends on

- How much forwarding resources has been allocated to the AF class to which the packet belongs
- The current load of the AF class, and, in case of congestion within the class
- The drop precedence of the packet

RFC 2597 does not mandate any mechanisms at the interior nodes to manage the AF traffic. It does reference the RED algorithm as a possible way of managing congestion.

Figure 19.13b shows the recommended codepoints for AF PHB in the DS field.

19.5 SERVICE LEVEL AGREEMENTS

A service level agreement (SLA) is a contract between a network provider and a customer that defines specific aspects of the service that is to be provided. The definition is formal and typically defines quantitative thresholds that must be met. An SLA typically includes the following information:

- **A description of the nature of service to be provided:** A basic service would be IP-based network connectivity of enterprise locations plus access to the Internet. The service may include additional functions such as Web hosting, maintenance of domain name servers, and operation and maintenance tasks.
- **The expected performance level of the service:** The SLA defines a number of metrics, such as delay, reliability, and availability, with numerical thresholds.
- **The process for monitoring and reporting the service level:** This describes how performance levels are measured and reported.

The types of service parameters included in an SLA for an IP network are similar to those provided for frame relay and ATM networks. A key difference is that, because of the unreliable datagram nature of an IP network, it is more difficult to realize tightly defined constraints on performance, compared to the connection-oriented frame relay and ATM networks.

Figure 19.16 shows a typical configuration that lends itself to an SLA. In this case, a network service provider maintains an IP-based network. A customer has a number of private networks (e.g., LANs) at various sites. Customer networks are connected to the provider via access routers at the access points. The SLA dictates service and performance levels for traffic between access routers across the provider network. In addition, the provider network links to the Internet and thus provides Internet access for the enterprise. For example, for the Internet Dedicated Service provided by MCI (**http://global.mci.com/terms/us/products/dsl**), the SLA includes the following items:

- **Availability:** 100% availability.
- **Latency (delay):** Average round-trip transmissions of ≤45 ms between access routers in the contiguous United States. Average round-trip transmissions of ≤90 ms between an access router in the New York metropolitan

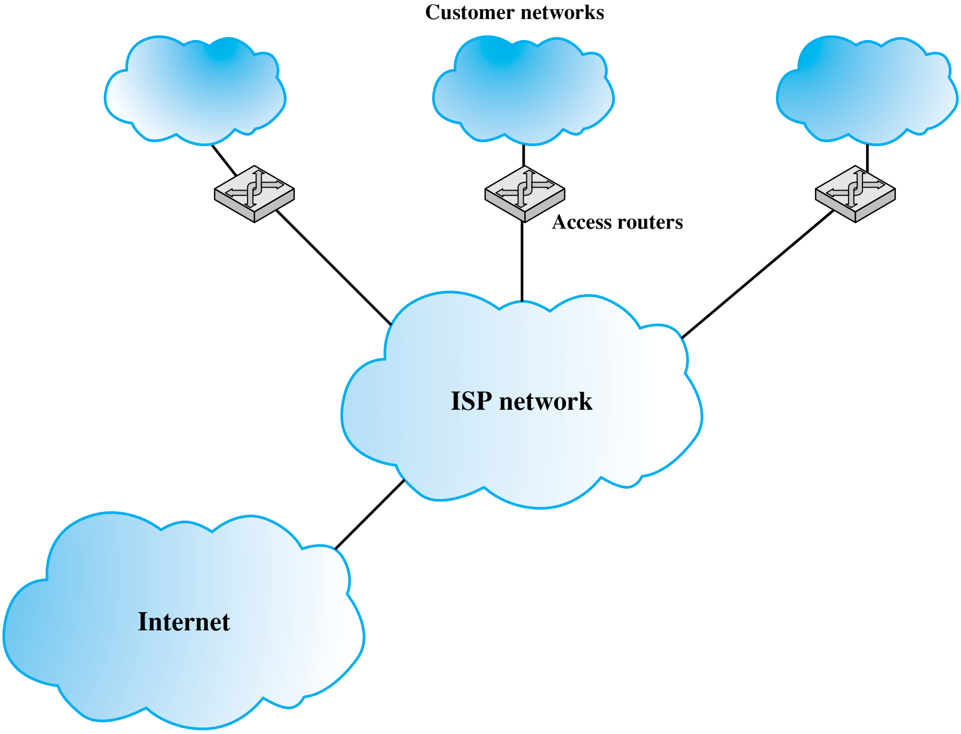


Figure 19.16 Typical Framework for Service Level Agreement

area and an access router in the London metropolitan area. Latency is calculated by averaging sample measurements taken during a calendar month between routers.

- **Network packet delivery (reliability):** Successful packet delivery rate of ≥99.5%.
- **Denial of service (DoS):** Responds to DoS attacks reported by customer within 15 minutes of customer opening a complete trouble ticket. MCI defines a DoS attack as more than 95% bandwidth utilization.
- **Network jitter:** Jitter is defined as the variation or difference in the end-to-end delay between received packets of an IP or packet stream. Jitter performance will not exceed 1 ms between access routers.

An SLA can be defined for the overall network service. In addition, SLAs can be defined for specific end-to-end services available across the carrier's network, such as a virtual private network, or differentiated services.

19.6 IP PERFORMANCE METRICS

The IPPM Performance Metrics Working Group (IPPM) is chartered by IETF to develop standard metrics that relate to the quality, performance, and reliability of Internet data delivery. Two trends dictate the need for such a standardized measurement scheme:

1. The Internet has grown and continues to grow at a dramatic rate. Its topology is increasingly complex. As its capacity has grown, the load on the Internet has grown at an even faster rate. Similarly, private internets, such as corporate intranets and extranets, have exhibited similar growth in complexity, capacity, and load. The sheer scale of these networks makes it difficult to determine quality, performance, and reliability characteristics.
2. The Internet serves a large and growing number of commercial and personal users across an expanding spectrum of applications. Similarly, private networks are growing in terms of user base and range of applications. Some of these applications are sensitive to particular QoS parameters, leading users to require accurate and understandable performance metrics.

A standardized and effective set of metrics enables users and service providers to have an accurate common understanding of the performance of the Internet and private internets. Measurement data is useful for a variety of purposes, including

- Supporting capacity planning and troubleshooting of large complex internets
- Encouraging competition by providing uniform comparison metrics across service providers
- Supporting Internet research in such areas as protocol design, congestion control, and quality of service
- Verification of service level agreements

Table 19.5 lists the metrics that have been defined in RFCs at the time of this writing. Table 19.5a lists those metrics which result in a value estimated based on a sampling technique. The metrics are defined in three stages:

- **Singleton metric:** The most elementary, or atomic, quantity that can be measured for a given performance metric. For example, for a delay metric, a singleton metric is the delay experienced by a single packet.
- **Sample metric:** A collection of singleton measurements taken during a given time period. For example, for a delay metric, a sample metric is the set of delay values for all of the measurements taken during a one-hour period.
- **Statistical metric:** A value derived from a given sample metric by computing some statistic of the values defined by the singleton metric on the sample. For example, the mean of all the one-way delay values on a sample might be defined as a statistical metric.

The measurement technique can be either active or passive. **Active techniques** require injecting packets into the network for the sole purpose of measurement. There are several drawbacks to this approach. The load on the network is increased. This, in turn, can affect the desired result. For example, on a heavily loaded network, the injection of measurement packets can increase network delay, so that the measured delay is greater than it would be without the measurement traffic. In addition, an active measurement policy can be abused for denial-of-service attacks disguised as legitimate measurement activity. **Passive techniques** observe and extract metrics from existing traffic. This approach can expose the contents of Internet traffic to unintended recipients, creating security and privacy concerns. So far, the metrics defined by the IPPM working group are all active.

Table 19.5 IP Performance Metrics

(a) Sampled metrics

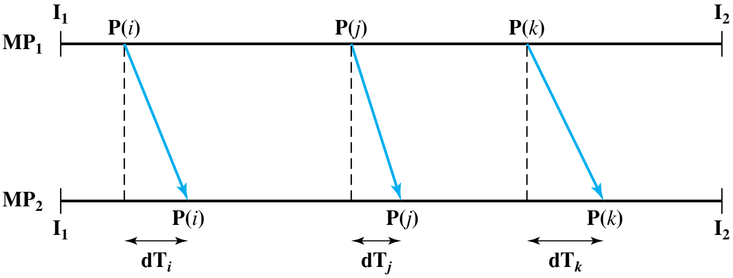
Metric Name	Singleton Definition	Statistical Definitions
One-Way Delay	Delay = dT, where Src transmits first bit of packet at T and Dst received last bit of packet at T + dT	Percentile, median, minimum, inverse percentile
Round-Trip Delay	Delay = dT, where Src transmits first bit of packet at T and Src received last bit of packet immediately returned by Dst at T + dT	Percentile, median, minimum, inverse percentile
One-Way Loss	Packet loss = 0 (signifying successful transmission and reception of packet); = 1 (signifying packet loss)	Average
One-Way Loss Pattern	Loss distance: Pattern showing the distance between successive packet losses in terms of the sequence of packets Loss period: Pattern showing the number of bursty losses (losses involving consecutive packets)	Number or rate of loss distances below a defined threshold, number of loss periods, pattern of period lengths, pattern of interloss period lengths.
Packet Delay Variation	Packet delay variation (pdv) for a pair of packets with a stream of packets = difference between the one-way-delay of the selected packets	Percentile, inverse percentile, jitter, peak-to-peak pdv

Src = IP address of a host
Dst = IP address of a host

(b) Other metrics

Metric Name	General Definition	Metrics
Connectivity	Ability to deliver a packet over a transport connection.	One-way instantaneous connectivity, two-way instantaneous connectivity, one-way interval connectivity, two-way interval connectivity, two-way temporal connectivity
Bulk Transfer Capacity	Long-term average data rate (bps) over a single congestion-aware transport connection.	BTC = (data sent)/(elapsed time)

For the sample metrics, the simplest technique is to take measurements at fixed time intervals, known as periodic sampling. There are several problems with this approach. First, if the traffic on the network exhibits periodic behavior, with a period that is an integer multiple of the sampling period (or vice versa), correlation effects may result in inaccurate values. Also, the act of measurement can perturb what is being measured (for example, injecting measurement traffic into a network alters the congestion level of the network), and repeated periodic perturbations can drive a network into a state of synchronization (e.g., [FLOY94]), greatly magnifying what might individually be minor effects. Accordingly, RFC 2330 (*Framework for IP*



I_1, I_2 = times that mark that beginning and ending of the interval in which the packet stream from which the singleton measurement is taken occurs
MP$_1$, MP$_2$ = source and destination measurement points
P(i) = ith measured packet in a stream of packets
dT$_i$ = one-way delay for P(i)

Figure 19.17 Model for Defining Packet Delay Variation

Performance Metrics) recommends Poisson sampling. This method uses a Poisson distribution to generate random time intervals with the desired mean value.

Most of the statistical metrics listed in Table 19.5a are self-explanatory. The percentile metric is defined as follows: The xth percentile is a value y such that x% of measurements $\geq y$. The inverse percentile of x for a set of measurements is the percentage of all values $\leq x$.

Figure 19.17 illustrates the packet delay variation metric. This metric is used to measure jitter, or variability, in the delay of packets traversing the network. The singleton metric is defined by selecting two packet measurements and measuring the difference in the two delays. The statistical measures make use of the absolute values of the delays.

Table 19.5b lists two metrics that are not defined statistically. Connectivity deals with the issue of whether a transport-level connection is maintained by the network. The current specification (RFC 2678) does not detail specific sample and statistical metrics but provides a framework within which such metrics could be defined. Connectivity is determined by the ability to deliver a packet across a connection within a specified time limit. The other metric, bulk transfer capacity, is similarly specified (RFC 3148) without sample and statistical metrics but begins to address the issue of measuring the transfer capacity of a network service with the implementation of various congestion control mechanisms.

19.7 RECOMMENDED READING AND WEB SITES

A number of worthwhile books provide detailed coverage of various routing algorithms: [HUIT00], [BLAC00], and [PERL00]. [MOY98] provides a thorough treatment of OSPF.

Perhaps the clearest and most comprehensive book-length treatment of Internet QoS is [ARMI00]. [XIAO99] provides an overview and overall framework for Internet QoS as well as integrated and differentiated services. [CLAR92] and [CLAR95] provide valuable surveys of the issues involved in internet service allocation for real-time and elastic applications, respectively. [SHEN95] is a masterful analysis of the rationale for a QoS-based internet architecture. [ZHAN95] is a broad survey of queuing disciplines that can be used in an ISA, including an analysis of FQ and WFQ.

[ZHAN93] is a good overview of the philosophy and functionality of RSVP, written by its developers. [WHIT97] is a broad survey of both ISA and RSVP.

[CARP02] and [WEIS98] are instructive surveys of differentiated services, while [KUMA98] looks at differentiated services and supporting router mechanisms that go beyond the current RFCs. For a thorough treatment of DS, see [KILK99].

Two papers that compare IS and DS in terms of services and performance are [BERN00] and [HARJ00].

[VERM04] is an excellent surveys of service level agreements for IP networks. [BOUI02] covers the more general case of data networks. [MART02] examines limitations of IP network SLAs compared to data networks such as frame relay.

[CHEN02] is a useful survey of Internet performance measurement issues. [PAXS96] provides an overview of the framework of the IPPM effort.

ARMI00 Armitage, G. *Quality of Service in IP Networks.* Indianapolis, IN: Macmillan Technical Publishing, 2000.

BERN00 Bernet, Y. “The Complementary Roles of RSVP and Differentiated Services in the Full-Service QoS Network.” *IEEE Communications Magazine*, February 2000.

BLAC00 Black, U. *IP Routing Protocols: RIP, OSPF, BGP, PNNI & Cisco Routing Protocols.* Upper Saddle River, NJ: Prentice Hall, 2000.

BOUI02 Bouillet, E.; Mitra, D.; and Ramakrishnan, K. “The Structure and Management of Service Level Agreements in Networks.” *IEEE Journal on Selected Areas in Communications*, May 2002.

CARP02 Carpenter, B., and Nichols, K. “Differentiated Services in the Internet.” *Proceedings of the IEEE*, September 2002.

CHEN02 Chen, T. “Internet Performance Monitoring.” *Proceedings of the IEEE*, September 2002.

CLAR92 Clark, D.; Shenker, S.; and Zhang, L. “Supporting Real-Time Applications in an Integrated Services Packet Network: Architecture and Mechanism” *Proceedings, SIGCOMM ’92*, August 1992.

CLAR95 Clark, D. *Adding Service Discrimination to the Internet.* MIT Laboratory for Computer Science Technical Report, September 1995. Available at **http://ana-www.lcs.mit.edu/anaWeb/papers.html**

HARJ00 Harju, J., and Kivimaki, P. “Cooperation and Comparison of DiffServ and IntServ: Performance Measurements.” *Proceedings, 23rd Annual IEEE Conference on Local Computer Networks*, November 2000.

HUIT00 Huitema, C. *Routing in the Internet.* Upper Saddle River, NJ: Prentice Hall, 2000.

KILK99 Kilkki, K. *Differentiated Services for the Internet.* Indianapolis, IN: Macmillan Technical Publishing, 1999.

KUMA98 Kumar, V.; Lakshman, T.; and Stiliadis, D. “Beyond Best Effort: Router Architectures for the Differentiated Services of Tomorrow’s Internet.” *IEEE Communications Magazine*, May 1998.

MART02 Martin, J., and Nilsson, A. “On Service Level Agreements for IP Networks.” *Proceeding IEEE INFOCOMM* ’02, 2002.

MOY98 Moy, J. *OSPF: Anatomy of an Internet Routing Protocol.* Reading, MA: Addison-Wesley, 1998.

PAXS96 Paxson, V. "Toward a Framework for Defining Internet Performance Metrics." *Proceedings*, *INET '96,* 1996. **http://www-nrg.ee.lbl.gov**

PERL00 Perlman, R. *Interconnections: Bridges, Routers, Switches, and Internetworking Protocols.* Reading, MA: Addison-Wesley, 2000.

SHEN95 Shenker, S. "Fundamental Design Issues for the Future Internet." *IEEE Journal on Selected Areas in Communications*, September 1995.

VERM04 Verma, D. "Service Level Agreements on IP Networks." *Proceedings of the IEEE,* September 2004.

WEIS98 Weiss, W. "QoS with Differentiated Services." *Bell Labs Technical Journal*, October–December 1998.

WHIT97 White, P., and Crowcroft, J. "The Integrated Services in the Internet: State of the Art." *Proceedings of the IEEE*, December 1997.

XIAO99 Xiao, X., and Ni, L. "Internet QoS: A Big Picture." *IEEE Network*, March/April 1999.

ZHAN93 Zhang, L.; Deering, S.; Estrin, D.; Shenker, S.; and Zappala, D. "RSVP: A New Resource ReSerVation Protocol." *IEEE Network*, September 1993.

ZHAN95 Zhang, H. "Service Disciplines for Guaranteed Performance Service in Packet-Switching Networks." *Proceedings of the IEEE*, October 1995.

Recommended Web sites:

- **Inter-Domain Routing working group:** Chartered by IETF to revise BGP and related standards. The Web site includes all relevant RFCs and Internet drafts.
- **OSPF working group:** Chartered by IETF to develop OSPF and related standards. The Web site includes all relevant RFCs and Internet drafts.
- **RSVP Project:** Home page for RSVP development.
- **IP Performance Metrics working group:** Chartered by IETF to develop a set of standard metrics that can be applied to the quality, performance, and reliability of Internet data delivery services. The Web site includes all relevant RFCs and Internet drafts.

19.8 KEY TERMS, REVIEW QUESTIONS, AND PROBLEMS

Key Terms

autonomous system (AS)
Border Gateway Protocol (BGP)
classifier
broadcast address
Differentiated Services (DS)
distance-vector routing
dropper
elastic traffic
exterior router protocol
inelastic traffic
Integrated Services Architecture (ISA)
interior router protocol
Internet Group Management Protocol
jitter
link-state routing
marker
meter
multicast address
multicasting
neighbor acquisition
neighbor reachability

network reachability
Open Shortest Path First (OSPF)
path-vector routing
per-hop behavior (PHB)
quality of service (QoS)
queuing discipline
Resource ReSerVation Protocol (RSVP)
shaper
unicast address

Review Questions

19.1 List some practical applications of multicasting.
19.2 Summarize the differences among unicast, multicast, and broadcast addresses.
19.3 List and briefly explain the functions that are required for multicasting.
19.4 What operations are performed by IGMP?
19.5 What is an autonomous system?
19.6 What is the difference between an interior router protocol and an exterior router protocol?
19.7 Compare the three main approaches to routing.
19.8 List and briefly explain the three main functions of BGP.
19.9 What is the Integrated Services Architecture?
19.10 What is the difference between elastic and inelastic traffic?
19.11 What are the major functions that are part of an ISA?
19.12 List and briefly describe the three categories of service offered by ISA.
19.13 What is the difference between FIFO queuing and WFQ queuing?
19.14 What is the purpose of a DS codepoint?
19.15 List and briefly explain the five main functions of DS traffic conditioning.
19.16 What is meant by per-hop behavior?

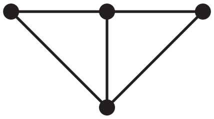
Problems

19.1 Most operating systems include a tool named "traceroute" (or "tracert") that can be used to determine the path packets follow to reach a specified host from the system the tool is being run on. A number of sites provide Web access to the "traceroute" tool, for example,

http://www.supporttechnique.net/traceroute.ihtml
http://www.t1shopper.com/tools/traceroute

Use the "traceroute" tool to determine the path packets follow to reach the host williamstallings.com.

19.2 A connected graph may have more than one spanning tree. Find all spanning trees of this graph:



19.3 In the discussion of Figure 19.1, three alternatives for transmitting a packet to a multicast address were discussed: broadcast, multiple unicast, and true multicast. Yet another alternative is flooding. The source transmits one packet to each neighboring router. Each router, when it receives a packet, retransmits the packet on all outgoing interfaces except the one on which the packet is received. Each packet is labeled with a unique identifier so that a router does not flood the same packet more than once. Fill out a matrix similar to those of Table 19.1 and comment on the results.

19.4 In a manner similar to Figure 19.3, show the spanning tree from router **B** to the multicast group.

19.5 IGMP specifies that query messages are sent in IP datagrams that have the Time to Live field set to 1. Why?

19.6 In IGMPv1 and IGMPv2, a host will cancel sending a pending membership report if it hears another host claiming membership in that group, in order to control the generation of IGMP traffic. However, IGMPv3 removes this suppression of host membership reports. Analyze the reasons behind this design decision.

19.7 IGMP Membership Queries include a "Max Resp Code" field that specifies the maximum time allowed before sending a responding report. The actual time allowed, called the Max Resp Time, is represented in units of 1/10 second and is derived from the Max Resp Code as follows:

If MaxRespCode $<$ 128, MaxRespTime $=$ Max Resp Code

If MaxRespCode $\geq$ 128, MaxRespTime is a floating-point value as follows:

0	1	2	3	4	5	6	7
1	exp			mant			

$$\text{MaxRespTime} = (\text{mant}|0\text{x}10) \ll (\text{exp} + 3) \quad \text{in C notation}$$

$$\text{MaxRespTime} = (\text{mant} + 16) \times 2^{(\text{exp}+3)}$$

Explain the motivation for the smaller values and the larger values.

19.8 Multicast applications call an API function on their sockets in order to ask the IP layer to enable or disable reception of packets sent from some specific IP address(es) to a specific multicast address.

For each of these sockets, the system records the desired multicast reception state. In addition to these per-socket multicast reception states, the system must maintain a multicast reception state for each of its interfaces, which is derived from the per-socket reception states.

Suppose four multicast applications run on the same host, and participate in the same multicast group, M1. The first application uses an EXCLUDE{A1, A2, A3} filter. The second one uses an EXCLUDE{A1, A3, A4} filter. The third one uses an INCLUDE{A3, A4} filter. And the fourth one uses an INCLUDE{A3} filter. What's the resulting multicast state (multicast address, filter mode, source list) for the network interface?

19.9 Multicast applications commonly use UDP or RTP (Real-Time Transport Protocol; discussed in Chapter 24) as their transport protocol. Multicast application do not use TCP as its transport protocol. What's the problem with TCP?

19.10 With multicasting, packets are delivered to multiple destinations. Thus, in case of errors (such as routing failures), one IP packet might trigger multiple ICMP error packets, leading to a packet storm. How is this potential problem avoided? *Hint:* Consult RFC 1122.

19.11 BGP's AS_PATH attribute identifies the autonomous systems through which routing information has passed. How can the AS_PATH attribute be used to detect routing information loops?

19.12 BGP provides a list of autonomous systems on the path to the destination. However, this information cannot be considered a distance metric. Why?

19.13 RFC 2330 (*Framework for IP Performance Metrics*) defines percentile in the following way. Given a collection of measurements, define the function F(x), which for any x gives the percentage of the total measurements that were $\leq x$. If x is less than the minimum value observed, then F(x) $=$ 0%. If it is greater or equal to the maximum value observed, then F(x) $=$ 100%. The yth percentile refer to the smallest value of x for which F(x) $\geq y$. Consider that we have the following measurements: -2, 7, 7, 4, 18, -5. Determine the following percentiles: 0, 25, 50, 100.

19.14 For the one-way and two-way delay metrics, if a packet fails to arrive within a reasonable period of time, the delay is taken to be undefined (informally, infinite). The threshold of reasonable is a parameter of the methodology. Suppose we take a sample of one-way delays and get the following results: 100 ms, 110 ms, undefined, 90 ms, 500 ms. What is the 50th percentile?

19.15 RFC 2330 defines the median of a set of measurements to be equal to the 50th percentile if the number of measurements is odd. For an even number of measurements, sort the measurements in ascending order; the median is then the mean of the two central values. What is the median value for the measurements in the preceding two problems?

19.16 RFC 2679 defines the inverse percentile of x for a set of measurements to be the percentage of all values $\leq x$. What is the inverse percentile of 103 ms for the measurements in Problem 19.14?

19.17 When multiple equal-cost routes to a destination exist, OSPF may distribute traffic equally among the routes. This is called *load balancing*. What effect does such load balancing have on a transport layer protocol, such as TCP?

19.18 It is clear that if a router gives preferential treatment to one flow or one class of flows, then that flow or class of flows will receive improved service. It is not as clear that the overall service provided by the internet is improved. This question is intended to illustrate an overall improvement. Consider a network with a single link modeled by an exponential server of rate $T_s = 1$, and consider two classes of flows with Poisson arrival rates of $\lambda 1 = \lambda 2 = 0.25$ and that have utility functions $U_1 = 4 - 2T_{q1}$ and $U_2 = 4 - T_{q2}$, where T_{qi} represents the average queuing delay to class i. Thus, class 1 traffic is more sensitive to delay than class 2. Define the total utility of the network as $V = U_1 + U_2$.

- **a.** Assume that the two classes are treated alike and that FIFO queuing is used. What is *V?*
- **b.** Now assume a strict priority service so that packets from class 1 are always transmitted before packets in class 2. What is *V?* Comment.

19.19 Provide three examples (each) of elastic and inelastic Internet traffic. Justify each example's inclusion in their respective category.

19.20 Why does a Differentiated Services (DS) domain consist of a set of contiguous routers? How are the boundary node routers different from the interior node routers in a DS domain?

19.21 The token bucket scheme places a limit on the length of time at which traffic can depart at the maximum data rate. Let the token bucket be defined by a bucket size B octets and a token arrival rate of R octets/second, and let the maximum output data rate be M octets/s.

- **a.** Derive a formula for *S*, which is the length of the maximum-rate burst. That is, for how long can a flow transmit at the maximum output rate when governed by a token bucket?
- **b.** What is the value of S for $B = 250$ KB, $R = 2$ MB/s, and $M = 25$ MB/s?

Hint: The formula for S is not so simple as it might appear, because more tokens arrive while the burst is being output.

19.22 In RSVP, because the UDP/TCP port numbers are used for packet classification, each router must be able to examine these fields. This requirement raises problems in the following areas:

- **a.** IPv6 header processing
- **b.** IP-level security

Indicate the nature of the problem in each area, and suggest a solution.