CHAPTER 2

Traffic Management Before Differentiated Services

Because engineering, so far, is a human activity, the evolution of traffic management is similar to that of any other human effort. Thomas Kuhn (1922–1996) presented one of the most popular theories describing this process of theory development in the early 1960s. His central statement is that science does not progress in an orderly fashion from lesser to greater truth, but rather remains fixated on a particular explanation (Kuhn 1996). Only with great difficulty can this explanation, or paradigm, be replaced by a new one.

Engineering, an application of science, is quite similar to science in this respect: There is always a prevalent notion about how a certain engineering problem should be solved and such notions are difficult to change, requiring a lot of time and effort. The additional factor affecting engineering is the rapid evolution of environment; paradigm changes are necessary purely because of altered problems. Differentiated Services may in fact turn out to be a new paradigm. To design a new one, however, the old paradigms—including their strengths and their weaknesses—must be understood.

2.1 Fundamental Concepts, Models, and Technologies

This chapter outlines the basic vocabulary and concepts of telecommunication networks and services, and shows which are the most appropriate. In the best case, vocabulary provides a useful framework for developing new ideas and analyzing old ones. In the worst case, inappropriate concepts may limit our thoughts; for instance, mathematical concepts may lead to an idea that everything essential can be expressed in mathematical formulae.

The viewpoint in this chapter is mainly that of an Internet service provider with a goal related more to the service business than to technical excellence. Hence, the following pages introduce the basic concepts and some basic issues related to the Internet services provision; the purpose is to prepare your thoughts for the thorough examination of the Differentiated Services approach in the rest of this book.

2.1.1 Customer Service

First, you have to decide what kind of service you are selling and to whom. For this discussion, three main customer groups are identified: residential, business, and academic. These groups differ considerably in certain aspects. In the academic environment, for instance, usage control has traditionally been relatively loose. As a consequence, quality control has been slight, also. Typical business customers are much more concerned about quality and performance because even a short service outage can cause significant losses.

In addition, it is important to notice that there are three network types:

- The public Internet
- A private network
- A virtual private network (VPN)

Private networks are physically separate from any public network. A *VPN* may use the same network resources as public services and other VPNs as well. Because the topic of this book is Differentiated Services for the Internet, the main concern is with public services; the secondary issue is VPNs, because they share the same resources and can, therefore, be thought to be a part of the public network. In contrast, although the same principles of Differentiated Services can be applied in private networks, the special issues of private networks are not extensively discussed in this book.

Business and academic users, although with diverse needs and expectations, have some predictable characteristics; residential customers, on the other hand, as a group of Internet users, comprise a mostly unexplored field of business. It is unclear what quality most users really need, or how much they are willing to pay for better quality. You may personally assess whether you would rather have a predictable price of service (a flat-rate charge) than have a predictable quality of service (guaranteed service with time-dependent pricing). Based on practical business experience so far, Internet service providers have found that most residential customers prefer the first option.

It is not, however, obvious that this inference is valid with regard to new services, such as video or audio multicasting. Could it be possible to combine quality differentiation with flat-rate pricing? It seems that if you can realize that kind of service, you may have a significant business advantage. Section 5.1.3, “Pricing as a Tool for Controlling Traffic,” in Chapter 5, “Differentiation of Customer Service,” shows how you can use Differentiated Services to achieve this.

A *service-level agreement (SLA)* is the formal part of the relationship between a service provider and a customer. If you look at the Web pages of an ISP, you usually find general assurances for residential Internet service related to the following issues:

- Throughput—that is, the bit rate available at the access point of the network
- Network availability, with compensation in case of unavailability and reporting of unavailability within a specific time
- Time to install new services and to respond and repair faults
- Round-trip transmission delay within the operator's domain, and possibly to some other destinations as well

In addition to these assurances, there could be more complex guarantees relating to more specific technological issues. A potential problem, however, is that ISPs' SLAs are often convoluted, and "[I]f you are not an educated buyer, you may not understand what you are really getting," as stated in *PC Weeks* online article (Neil 1998). This is just one question that this book seeks to address thoroughly.

2.1.2 Network with Services

If you have successfully acquired a sufficient customer base for your business and have made appropriate service-level agreements, you need a network to satisfy the customer needs and meet your obligations under the SLAs. This section addresses some of the attendant technical issues of network services.

A packet network consists of nodes that are practically computers with some special hardware and links between them. A node that handles packets is usually called a router. Depending on the capabilities of the node, however, it may instead be called a bridge or a switch.

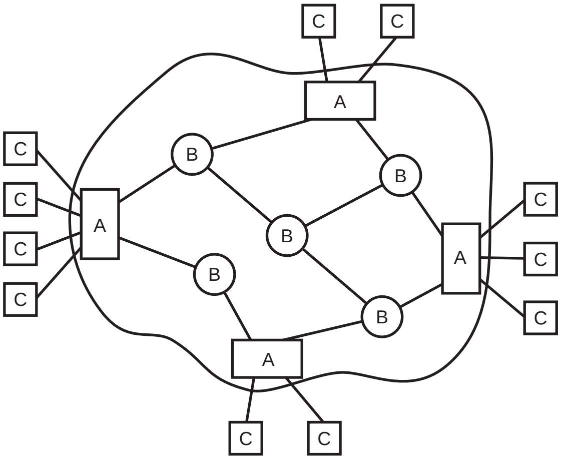
Note

The term *node* is used as a general term, covering bridges, routers, and switches, and the capabilities of a node are defined as exactly needed in each case. This book focuses primarily on networks with point-to-point links—that is, each link connects only two nodes.

Although the concepts are overlapping and there is not any single criterion to distinguish them, it is fair to say that a bridge has much less knowledge about network topology than a router or a switch. The key property of switching is that a switch makes the forwarding decision (for a packet or some other information unit) based on a label rather than the full destination address. The main advantage of this arrangement is that it allows better exploitation of hardware, which means it can accelerate forwarding functions (Davie, Doolan, and Rekhter 1998).

A large network can usually be divided functionally into two sections: the access network and the backbone (or core) network (see Figure 2.1). The main tasks of the access network are to physically connect your customers to the network and to provide appropriate tools, such as pricing capabilities, to manage the relationship between operator and customer. For these reasons, the total capacity of a boundary node connecting access and core networks, measured in bit rate, is usually much smaller than that of an interior node. On the other hand, however, a boundary node has more "sophisticated" tools that enable it to control and measure individual flows. Interior nodes, for their part, govern large bundles of aggregate traffic. Therefore, an interior node's main task is to efficiently transmit high-speed traffic.

Figure 2.1 The main building blocks are boundary nodes (A), interior nodes (B), and customer equipment (C).



2.1.3 *Network Operation and Management*

The quality of network service depends basically on two issues: the sufficiency of network resources and the capability of a traffic-handling mechanism to efficiently utilize the available resources. If there are not enough resources, even advanced traffic-handling mechanisms cannot solve all problems. From an operator viewpoint, therefore, it is extremely important to manage the resources appropriately. The following section, "Traffic Handling," briefly introduces the principles of resource allocation, or network dimensioning, in circuit-switched and packet-switched networks.

The goal of network dimensioning is to make certain that the network has enough capacity to keep customers satisfied with the service. In a circuit-switched network, such as a telephone network, the main effect of insufficient capacity is that some of the call requests must be discarded. The standard measure of this quality parameter is call-blocking probability. Call blocking can be either directly measured or estimated by using a mathematical modeling.

Direct measurement—that is, counting the number of calls rejected because of insufficient capacity and the number of successful calls—is a useful tool to estimate the quality level of a real network. This approach is obviously not suitable for determining the required capacity of a future network. (How can you measure an imaginary network?) For such cases, you need an indirect approach.

The indirect approach is based on the measurement of traffic load and on certain assumptions about the statistical characteristics of traffic process. First, a base calculation must be determined. To do so (for this discussion), make the following assumptions:

- Every call reserves one channel.
- The average call arrival rate is λ (calls / s).
- The arrival process is Poisson—that is, the inter-arrival time between call attempts is exponential.
- The average call holding time is h (s).
- A link has together S channels.

Under these conditions, the call-blocking probability can be calculated by using the Erlang loss formula. (See Chapter 5, "Differentiation of Customer Service," for further information.) Although the underlying assumptions of the Erlang formula are seldom exactly valid, it can be used to illustrate certain important phenomenon of any service with capacity reservations.

If the call-blocking probability is fixed—say, to 0.1%—the allowed load level (A / S) depends largely on the number of channels (S). The main message is that, if you divide the link into fixed parts in a way that each part has its own traffic that can use only that part of the link capacity, the allowed link utilization may decrease dramatically, as shown in the examples in Table 2.1.

Table 2.1 Link Utilization for a 0.1% Call-Blocking Standard

Link Parts	Number of Channels	Theoretical Load Level
1	500	448 calls (90%)
10	50	325 calls (65%)
100	5	76 calls (15%)

Further, it should be noted that you must be able to divide the load evenly among the 100 parts to get even the figures in Table 2.1. From an efficiency point of view, therefore, it is always questionable to divide the available capacity into a large number of fixed parts.

A similar phenomenon is noticeable in packet networks as well, although the effect is not as prominent because the buffering of packets softens some problems—but only provided that the link capacity can be divided proportionally to the load level of each part. If the load of one of the parts exceeds the capacity of that part, the corresponding buffer eventually overflows even though the other buffers are empty. In this case, also, a fixed division of link capacity (without the possibility to use the other part of the link capacity) may lead to a significant waste of resources.

This kind of division approach could be reasonable, but only if there are clear reasons behind it (for instance, because different user groups must be tightly separated for security reasons). The same dilemma arises with regard to some standardization efforts by the Internet Engineering Task Force (IETF), such as Multiprotocol Label Switching (MPLS) and Resource Reservation Protocol (RSVP).

2.1.4 Traffic Handling

When you have successfully finished the network dimensioning phase, you have enough network resources to handle the traffic demand. The next step is to ascertain that you have appropriate traffic-handling mechanisms in your network because, in reality, the performance of traffic handling determines the quality of packet flows.

The first, and fundamental, requirement is that your network must be able to transmit the packet to the required destination. This fundamental task of a packet network is actually done by two processes: routing and forwarding. *Routing* is a mechanism implemented in the network nodes to collect, maintain, and distribute information about paths to different destinations in the network. In other words, routing does not directly concern packets, but it enables an efficient packet *forwarding* mechanism that is in charge of conveying packets to the right destination. These two processes together make sure that packets reach the right destination, provided that there are enough resources for the transmission.

Another aspect of traffic handling is the treatment of the packets inside the network nodes. Traffic handling can be done on different levels of aggregation. The lowest level in a packet network is one arriving packet as an independent entity without any information about any other earlier packets in the network. In this case, traffic handling must rely on the information available in the packet.

From a traffic-handling viewpoint, the main fields in an Internet Protocol version 4 (IPv4) packet header are the source address, destination address, type of service (ToS), and protocol. The node may treat the packets differently based on these fields; for instance, a node may immediately drop all packets coming from a certain source address, or may give fast delivery for all packets that use a certain protocol. Although routers may, in principle, apply quite complex rules, the implementation and management of the system may become impossible when millions of packets are arriving every second. Further, any system without any knowledge about past traffic is inherently limited because it does not take into account the amount of resources used by different flows in the past.

Therefore, it is often desirable to classify packets into groups, follow the traffic process of the group, and make the required decision based on the information collected in this way. Basically, there are two different approaches to accomplish the task:

- Make the measurements at every node for every individual group.
- Make the measurements only at certain points in the network (usually at the edge of the network), and convey the needed information somehow through the network.

Both approaches have their advantages and disadvantages. If the number of groups is very large, the measurement system in interior nodes could limit the system performance (and note that the same measurement must be done at every node). Further, in many control schemes, it is not enough just to measure the traffic; some information related to the groups is also required—for instance, how much each customer is paying for his service. Because this kind of information is usually available at the boundary node, one reasonable approach is to make the necessary measurement only once in the boundary node and then transfer the relevant information to other nodes. Now there are two options:

- The information can be placed in every IP packet.
- The information can be transferred by using special control packets.

The first option is often more realistic in packet networks (although the second option is possible as well). In particular, if the information content can be expressed by a couple of bits, and it is changing frequently, the transmission of additional packets with large IP headers is not an efficient solution. Therefore, it is desirable to reserve some bits in the packet header to transmit relevant information in every packet. Actually there is one octet, ToS, reserved for this purpose in an IPv4 packet, although it is not yet widely used. The basic philosophy of Differentiated Services is to utilize the ToS octet in a way that enables service differentiation throughout the network without keeping track of all flows at every node.

2.1.5 Traffic Models for the Internet

The network dimensioning of a packet network is traditionally based on delay characteristics. This dimensioning problem can be divided into the capacity allocation problem of each individual link in the network. Because packet networks inherently rely on statistical multiplexing, at least one queue is needed for each outgoing link. (The actual queuing systems are discussed further in Chapter 5, "Differentiation of Customer Service.")

In the simplest queue model using the *first in, first out (FIFO)* discipline, Poisson arrival process, and exponential service-time distribution, there is a simple formula, as shown in Formula 2.1, that connects the average load (ρ) and average waiting time in the queue (D), and average service time (h).

Formula 2.1

$$D = h\rho / (1-\rho)$$

You can apply this formula to a case where packets arrive at a buffer and are sent to a link with a certain speed. The service time of a packet is determined by the packet size and link speed. If the average packet size is 500 bytes, link rate is 100Mbps, and the average load is 0.5 (that is, 50Mbps), for example, the average waiting time of a packet according to Formula 2.1 is only 0.04 milliseconds. Even if the average load is as high as 0.99, the average, theoretical, queuing delay is less than 4 milliseconds.

This simple calculation may indicate that delay is not any problem in high-speed networks. Unfortunately, this is not a right conclusion because of several reasons. First, even if all other assumptions were valid but the average load increases by 1% from 0.99 to 1, the theoretical delay grows to infinity. Therefore, information about average load only is not sufficient for making practical conclusions.

Second, extensive studies have shown that the Poisson assumption is not valid for modeling Internet traffic, as noticed in the studies made at Bellcore in the early 1990s (Leland *et al*. 1993, 183–193). In particular, the aggregate arrival process of packets is not Poisson, but it contains a long-term correlation process that essentially changes the characteristics of traffic process. It is said that the traffic process is *self-similar*. Self-similarity in this context means that there are similar traffic variations on every time scale from milliseconds to weeks. Because of this fundamental nature, it is almost impossible to calculate any exact delay or packet-loss ratio for typical Internet traffic.

Moreover, even with the right formula, it is difficult to measure the required traffic parameters; one characteristic of self-similar traffic is that extremely long measurement periods are needed to acquire accurate results. Even if you did have both the formula and the parameters, a relatively small change in some of the parameters could result in a remarkable effect on the delay or loss figures. Therefore, this kind of approach may give some understanding about the system, but probably not any definite numbers for resource-management purposes.

Note

One additional warning is also valid: On the Internet, there is no such thing as traffic process independent of the network resources. This is evident if you consider the nature of TCP, which adjusts the bit rate of each connection based on the load situation in the network. (See section 2.3.2, "Basic Best-Effort Service Based on TCP," later in this chapter.) Consequently, analytical formulae are seldom useful. Instead, cumbersome simulations are usually needed to investigate Internet performance issues.

Simplistic models can be misleading. It is much easier to implement a network with three nodes and three links; you could have complete information about everything going on in this small network. If you have only three nodes, for instance, you can easily configure permanent connections between each pair of nodes and even reserve capacity for several different classes. On the contrary, if you have 1,000 boundary nodes and five service classes, this simple scenario is totally impractical.

If you want to establish a permanent connection with a specific bit rate between each boundary node pair for every service class, you need to manage 2,497,500 connections. Either you have a superb automatic management system or you have to forget the whole idea. Besides, as stated earlier, division of the link capacity into distinct parts is an inefficient way to utilize your resources. Therefore, you need a sensible, somewhat flexible, approach with some level of control over the traffic.

2.1.6 Technological Progress

The progress of optical transmission systems has been amazing during the past few years. The most advanced systems with *wavelength division multiplexing (WDM)* can provide bit rates as high as 100Gbps. To understand the real capacity of those systems, suppose that you have one transatlantic link with a capacity of 100Gbps in both directions. How many minutes of telephone calls can every inhabitant in the United States make during a day?

A straightforward calculation leads to this theoretical result: Each of the approximately 268,000,000 inhabitants could speak about 8.4 minutes every day if 64kbps coding is used. This is a considerable length, even though it is not realistic to suppose that the whole link capacity can be exploited by phone calls. On the other hand, if you are not using the standard PCM coding, but a more efficient coding scheme, you can lengthen the duration of 7.5 minutes up to even an hour!

Such a huge capacity means that the network nodes must be extremely capable. If the same link is used to transmit IP packets with an average size of 500 bytes, for instance, the nodes must be able to handle an average of 25 million packets coming from one link. That is certainly a hard task, although not impossible, even when taking into account the rapid development of information-processing technology. Therefore, although the transmission capacity may seem to be limitless, some bottlenecks will continue to occur in the foreseeable future (either inside network nodes or at access networks).

A consideration of the growth of network capacity, bottlenecks, and Internet traffic models leads to the conclusion that traffic engineering is needed even in networks with huge capacity. There are too many uncertainties to allow a feasible solution without any traffic control.

2.2 Traditional Telecommunication Approaches

The telephone network has a long tradition. Some significant changes in technology have occurred: first the emergence of automatic exchanges, and then digital transmission, and finally digital exchanges. All these developments have been important inside the network, and they have had certain effects on customer service as well. The operational principle has remained the same, however, in such a way that telephone networks have been able to smoothly evolve from one technological phase to another. What is the continuity of telephone networks? One apparent answer is the target—that is, to provide a medium for transmitting voice over long distances.

Telephone networks are now used for other purposes as well. These other uses are possible because the applications, such as fax and data connections, have adapted to the characteristics of the telephone networks. Certain limits do apply, however, with regard to this approach—for instance, a voice channel is too slow for many advanced applications. These issues are analyzed in Section 2.2.1, “Circuit-Switched Networks.”

The main solution for these telephone network problems is Asynchronous Transfer Mode (ATM). Because of the telephone background, some of the basic principles of telephone networks can be found in ATM as well; in particular, a connection should be established before any user traffic can be transmitted through the network. The main objective of Section 2.2.2, “ATM Networks,” is to provide an outline of the main strengths and weaknesses of the ATM approach.

2.2.1 Circuit-Switched Networks

In circuit-switched networks, a dedicated channel (or circuit) is established for the duration of a transmission. Telephone networks, the most universal circuit-switched networks, initially applied an utmost mode of circuit switching in which the network provided an unbroken, undivided electrical circuit for a specified frequency region between two telephones.

The technical evolution, including thorough digitalization of communication networks, has obscured this clear situation of electrical circuits: Very seldom anymore is there any fixed circuits between end terminals; instead, there is usually a certain type of transmission channel. Therefore, a circuit-switched network can be recognized by the following list of determinants:

- The network reserves certain fixed capacity for the information transmission for every channel.
- The network service provides a small additional delay to the fixed delay determined by speed of light, and a minimal end-to-end delay variation.
- The distinguishing of different channels during the transmission is based on the location of the information in the frame structure rather than the information inside the transmission channel.

Although the previous characteristics are typical for circuit-switched networks, you can find several deviations from the basic form of circuit switching in current networks. First, it is possible even in analog telephone networks to detect idle periods in a telephone conversation and to use these periods for transmitting some other information. As a consequence, although it appears for the user that he has a continuous connection to the other end, the connection could be of an on / off nature.

In digital networks, the possibilities are even more versatile. If all information is presented in digital form, for example, a basic circuit-switched network can manipulate information inside the network. Digital telephone exchanges, for instance, store all (or almost all) information for a short duration before it is transmitted forward. This is an unavoidable action, because it is possible that two different incoming channels that have exactly the same arrival time but a different incoming link will be multiplexed to the same outgoing link. Either of the channels has to be delayed.

Despite the development of circuit-switched technology, it is still evident that circuit-switching systems are primarily ideal for communication systems that require data to be transmitted in real-time during a relatively long period of time. Because they provide a tool for transmitting information from one place to another, however, they could, in principle, be used as a basis for any communication network. What does this actually mean if you have 1,000,000 end users who require transmission service for Internet traffic?

In a circuit-switched network, you typically can establish only connections with a predefined bit rate. Because the digital telephony network is based either on a bit rate of 56kbps or 64kbps, for instance, it is difficult to support a connection with an arbitrary bit rate because only certain multiples of the basic bit rate are usually supported. If, and when, your end users have various and continuously changing bit-rate needs (from some kilobits per second to several megabits per second), you could have big troubles with your customer and network services. Your customers will have to use certain predefined bit rates even though they might need something else most of the time.

You may be able to acquire an exceptional circuit-switching system capable of transmitting a very large number of different bit rates. For example:

n*1kbps where n = 1,2,3,..., 1,000,000

Does this kind of network solve your problem? Unfortunately, although it does solve a part of the problem, significant difficulties still exist. First, the network will always either establish a new connection or modify an old one when the required bit rate of a connection changes—and there will be a huge number of changes every second in your network. Consequently, your network must have a very advanced signaling system to transmit all information related to bit-rate changes—and a signaling system may require a considerable amount of transmission capacity.

The second problem is that each customer or application must be able to predict what bit rate the application needs within the next second, minute, or hour. This is definitely possible in the case of certain established applications, such as telephony calls in current networks. However, many Internet applications are not based on any fixed bit rate.

If your network did have an excellent signaling system and all applications were able to predict their bit rate, you would still encounter fundamental problems. If the bit rate changes, say, once a second, the network probably cannot update the capacity reservations in a way that no resources are wasted.

Finally, if you were able to solve all the previously discussed problems, you would have to dimension your network in such a way that your customers would remain satisfied. This is the same task you need to do if you are responsible for telephony service. You must go through the following phases for all links in the network:

Phase 1 Predict the offered traffic during busy hours for all the network links.

Phase 2 Specify quality of service target—for example, the allowed probability that a connection attempt is rejected because of insufficient link capacity.

Phase 3 Determine, based on the traffic prediction, the capacity required to satisfy QoS.

Phase 4 Find out what is the cheapest product that has at least the capacity calculated at Phase 3.

Phase 5 Order the required product (or update the current product).

Phase 6 Make the installation or update.

A lot of problems can arise during all these phases. You cannot assume that your Internet traffic prediction is accurate: An increased traffic demand might be either 100% or 200% per year, resulting in a relatively high probability that you will either overestimate or underestimate the actual demand.

In telephony networks, Phase 3 is usually done with the aid of the Erlang loss formula, which gives the call-blocking probability as a function of offered traffic and the number of channels. Because your customers will have variable connection requests, Phase 3 is much more complicated than in the case of telephony networks. You need a more advanced tool; although several methods are available, they require some effort to be applied (Roberts, Mocci, and Virtamo 1996).

If you want to build your own network based on real products, you must cope with quite rough expansion steps. The available increments in a backbone network based on optical transmission systems, for example, are 155Mbps, 622Mbps, and 2.488Gbps. As an inescapable result, even if you are a skillful network planner, you are using your network resources inefficiently. As a target, a long-term average load of 20% is more ambitious than easy. Therefore, a traditional circuit-switched network does not seem to be a cost-efficient approach to transmitting Internet traffic; and because there is only one guaranteed service class (although with several bit rate levels), it is not versatile.

The most significant advantage of this approach is robustness: Customers can use exactly the bit rate they have requested (and paid for), but not a bit more than that. Further, there are no lost packets or bits inside the network, provided that your network is working properly. For the same reasons, the customer service can be considered fair.

Note

Some intricate issues make the assessment of fairness more complicated than what could be expected, however: The call-blocking probability may depend on the bit rate requested by the customer, on the time and date of the request, and on the number of links on the connection path. Although the question whether the result is fair is surely interesting, this discussion skips it because it seems that a pure circuit-switching network cannot provide a proper solution to the purpose of transmitting Internet traffic.

Although the preceding considerations seem to lead to an impractical result, such considerations serve one purpose: Identifying several key problems that most likely are solvable by any networking technology specifically designed for the transmission of Internet traffic. The problems can be summarized as follows:

- Many flows are of very short duration.
- Connection establishment tends to require complicated actions inside the network.
- Quality and capacity requirements of new applications vary within extremely wide bounds.
- Traffic characteristics of flows (or connections) are difficult to predict.
- The use of a resource-reservation principle tends to leave the majority of network capacity unused.

The third item in the preceding list relates to the fundamental attribute of versatility. The other items relate mainly to cost efficiency. With circuit-switching systems, no significant problems seem to be related to fairness or robustness. As discussed later, these attributes are the main concerns of some packet-switching systems.

You might be wondering why this discussion has so far focused on the evaluation of circuit-switched networks. The answer is that, if you want to improve the quality of service of a packet network by separate connections with fixed bit rate and quality, you will most likely encounter the same problems evident with circuit-switching technologies.

2.2.2 ATM Networks

Asynchronous Transfer Mode (ATM) has been regarded as a promising solution to some of the problems described in Chapter 1, "The Target of Differentiated Services,"—such problems as the lack of versatility in circuit-switched networks, for example. A wide standardization effort within the International Telecommunications Union (ITU) and ATM Forum has led to an extensive set of standards that specifies all the issues needed to build a workable ATM network.

Taking into account these facts, it is quite surprising that there are so few real demonstrations of customer service based on end-to-end ATM connections. (Although ATM has definitely been widely used as a backbone technology, that type of use exploits only a relatively small part of the whole set of ATM standards.) Chapter 3, "Differentiated Services Working Group," sheds light on this issue.

The target of the original project that led to the ATM technology was real-time cable TV using a high-speed digital transport (Coudreuse 1997). Therefore, the main challenges were high-speed switching (note that the project took place almost 20 years ago) and delay control. It soon became apparent, however, that a network with those characteristics could be useful for almost any imaginable purpose. So flexibility, or *versatility* in the terminology of this book, was given priority at an early stage of ATM development.

In the case of ATM, flexibility is achieved by an intrinsic property: All types of information are presented in the same form using equal-sized packets, called *cells*. The size of an ATM cell is 53 bytes (424 bits), of which 5 bytes are used for the header and 48 for user information. The size of the cell was a result of significant debate between two camps: those who wanted to keep the relative overhead of the cell header small by a large cell, and those who wanted to keep the packetization delay short by a small cell. Unfortunately, the final compromise of 48 bytes cannot meet either of these targets very well, as the following examples illustrate.

The packetization delay for an 8kbps audio stream can be calculated as shown in Formula 2.2.

Formula 2.2

$$\text{Delay} = 48*8 \text{ bit} \ / \ 8000 \text{ bit/s} = 48 \text{ ms}$$

This is the time needed to fill the information field of an ATM cell if a bit rate of 8kbps is used. Although 48 milliseconds is not a very long delay as such, it is a significant part of the allowed delay of a high-quality telephone call. According to Multimedia Communications Forum, delay shall be less than 160 milliseconds with echo controller, and a low delay limit of only 22 milliseconds is applicable when supporting connections to conventional telephones without supplementary echo control.

Because of the small size, the minimum overhead of an ATM cell is relatively large—that is, 5/53 = 9.4%. The real overhead when ATM is used to transmit IP packets is larger because IP packets should be adjusted into the cells with an extra protocol layer, the *ATM Adaptation Layer (AAL)*, that requires an additional byte in the ATM cell. In addition, because an IP packet is rarely a multiple of 47 bytes, one ATM cell per IP packet is partly unused. As a result, a typical overhead of ATM layers when used for transmitting IP packets is approximately 20%.

ATM can delicately solve one problem circuit-switched networks face: lack of versatility. Any user or application can transmit any bit rate whatsoever, limited only by the physical bit rate of the links, and the ATM network can aggregate any combination of different bit rates into one link, even without knowing what the real bit rate of each connection is. In that sense, ATM is surely versatile. The other side of this advantageous property is that every ATM node must have advanced tools to control traffic if and when it attempts to guarantee specific quality of service.

Virtual Connections

The key instruments of traffic handling in ATM networks are virtual circuits (VCs) and virtual paths (VPs). Because the quality control is based on connections, ATM is fundamentally a connection-oriented technology even when used to transmit IP packets. On the other hand, because ATM utilizes packet-based technology (rather than circuit switching) to transfer information, every packet (or cell) includes information about the destination address. Because the cell size is relatively small, it is not reasonable to convey the whole address in every cell; instead, a local identifier specifies the connection to which the cell belongs. Because of this locality, the required size of the identifier field is of moderate length—in ATM, 24 bits (assuming that the first 4 bits of the ATM header are not used for this purpose).

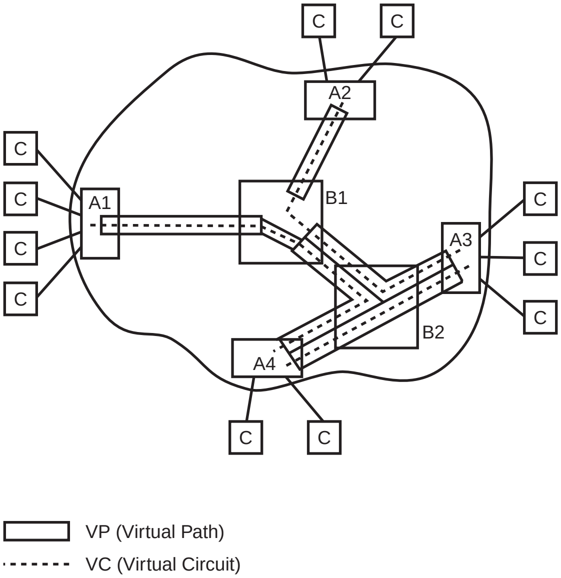
These 24 bits makes it possible to distinguish 16,777,216 connections on every link—that is, a large number. The downside of largeness, agreeable as such, is that it generates other strict requirements if really applied as a whole. Suppose, for example, that the average lifetime of a connection is three minutes. The node should be able to handle, in theory, 93,206 connection requests every second on every link. Although this could be realized technically, it would certainly bring about a serious operational and management burden unless the connection handling is truly straightforward.

To solve this dilemma, ATM uses the two levels of identifiers mentioned earlier: *virtual circuit (VC)* and *virtual path (VP)* identifiers. The basic idea of this arrangement is that a VP forms a relatively permanent pathway for cells between two ATM nodes, possibly far away from each other; and VCs can then be established and terminated without making any actions in the intermediate nodes. Figure 2.2 illustrates this system.

Node B2 in Figure 2.2 is a so-called VP cross-connect node that looks and takes into account only the VP identifiers, but leaves all VC identifiers unchanged. The other backbone node, B1, can support both VCs and VPs. The result is that virtual circuits can be established and modified inside a permanent VP from A1 to A4 through B1 and B2, without any actions in nodes B1 and B2.

Although VPs are generally useful, they have some negative effects as well. One of the key problems of VPs is that, because a VP usually needs a fixed bandwidth reservation, an excessive use of VPs tends to severely deteriorate the efficiency of statistical multiplexing. Note that if the bandwidth reserved for the VP is always changed when a new connection is established, you lose the fundamental advantage of VPs (because all intermediate nodes should be informed of all changes).

Figure 2.2 Virtual circuits and virtual paths in an ATM network.



To provide versatility in quality characteristics, a third set of building blocks for ATM traffic management is needed: the service categories. ATM Forum has so far defined six services with the following admission criteria and efficiency of statistical multiplexing:

- *Constant Bit Rate (CBR)*: Admission control is based on the peak rate of the connection, usually without statistical multiplexing.
- *Real-Time Variable Bit Rate (rt-VBR)*: Admission control is based on several parameters that make it possible to apply more efficient statistical multiplexing.
- *Non-Real-Time Variable Bit Rate (nrt-VBR)*: Admission control is similar to rt-VBR, but statistical multiplexing could be more efficient because of better possibility for buffering ATM cells.
- *Available Bit Rate (ABR)*: Connection-level admission control is based on a minimum bit rate; in addition, a cell-level admission control is based on load level inside the network. It provides high statistical multiplexing—at least in theory.
- *Unspecified Bit Rate (UBR)*: With no or minimal admission control, UBR provides very efficient statistical multiplexing.
- *Guaranteed Frame Rate (GFR)*: Admission control is based on a minimum bit rate available for a connection; GFR provides efficient statistical multiplexing. (The standardization is unfinished.)

Constant Bit Rate (CBR)

The CBR service category is intended for real-time connections that need tight synchronization between the traffic source and destination. Further, it is supposed that the source sends traffic with a constant bit rate or, actually, with a constant cell rate (because the source is sending cells, not individual bits). It is unrealistic, however, to require that a source send cells with exactly the same inter-arrival time (because the ATM network itself may generate jitter to any originally regular flow). Therefore, CBR service allows a small variation in cell rate, but not any persistent excess of cell rate.

Because network-management systems must rely on the assumption that CBR connections are really using a constant bit rate, or at least that the bit rate is below a specified limit, there must be tools for restricting offered traffic of every CBR connection. To guarantee robust service, therefore, two traffic control functions are used: *Usage Parameter Control (UPC)* and *Connection Admission Control (CAC)*.

Excessive cells are rejected by the UPC mechanism situated at the ingress ATM node. (A similar function at network-to-network interfaces is called *Network Parameter Control [NPC].*) CAC mechanism decides whether a new connection request can be accepted into the network without compromising quality of service of existing connections. The technical aspects of these control functions are discussed in Chapter 5, “Differentiation of Customer Service.”

In addition to end-to-end virtual connections, CBR service is regularly used in the case of VPs because statistical multiplexing between VPs that are used to transmit VBR VCs inside them is extremely difficult to manage. Therefore, even though the traffic inside a typical VP is anything but constant, VPs are usually supplied with constant resources.

Variable Bit Rate (VBR)

The next two categories, rt-VBR and nrt-VBR, are aimed at improving the statistical multiplexing of CBR service. As the name indicates, the basic difference between CBR and VBR is that VBR allows more fluctuations in traffic process than CBR service does. A VBR connection is characterized by three parameters: Peak Cell Rate (PCR), Sustained Cell Rate (SCR), and Maximum Burst Size (MBS). Based on these parameters and on information about network resources, ATM nodes calculate the required bandwidth for a set of connections. This task has turned out to be very difficult to carry out in real-time; in particular, it should be noted that the required parameter for any flow may depend essentially on both other connections and the available link rate.

The nrt-VBR service is applicable for those VBR connections that have no inherent need for time synchronization between source and destination. The rt-VBR service category was principally designed for transmitting compressed video traffic. There are two principles of video coding. With constant bit-rate coding, the output of video coder is constant-bit-rate and the quality of the picture is variable. In particular, scene changes generating high peaks of information to be transmitted are difficult to support with CBR coding without temporarily deteriorating picture quality. You can avoid this problem by variable bit-rate coding that makes possible a constant quality.

Available Bit Rate (ABR)

The ABR service category tries to combine definite quality guarantees with flexible use of network resources. This target is ambitious: How can a network give any guarantees if it at the same time allows users to send traffic with arbitrary bit rates? Actually it cannot; it must regulate the bit rates used by customers quite tightly.

The principal assumption behind ABR is that the applications using the service do not have any strict bit-rate requirements, but that they can benefit from increased bit rate. In addition, it supposes that packet losses are so harmful, either for users or for the network, that they should be avoided even at the expense of complicated control mechanisms. The control mechanism is designed to offer a fair share of network resources for each ABR connection, basically by dividing the available bandwidth at each bottleneck link according to a definite rule. Information about available bandwidth is then transmitted through the network by specific cells, called Resource Management (RM) cells.

ABR service is suited only for those systems and applications that can quickly adjust their bit rate. (Otherwise, a lot of cells might be lost at the ingress node before the cells enter the ABR network service.)

It is fair to say that ABR service is *versatile* in the sense that it provides various and variable bit rates; *robust*, because it tightly controls traffic sent by the user; and *fair*, because the available capacity is divided equitably. The major concern regarding ABR is whether it can be cost efficient because of its inherent complexity.

Unspecified Bit Rate (UBR)

The UBR service category differs fundamentally from the other ATM service categories in the sense that UBR sources neither specify nor receive any bit rate, delay, or loss guarantee. UBR service can be used by applications that can adjust their bit rate in case of lost or delayed cells.

The lack of guarantees and of strict control mechanisms bring about fairness problems; in fact, fairness issues are either left for upper-layer protocols, such as TCP/IP, or the network operator supposes that most of the time there are no critical fairness problems (for instance, because of low network utilization). The fairness problems related to UBR are basically the same as those with the best-effort service model in IP networks (see Section 2.3.2, “Basic Best-Effort Service Based on TCP”). In particular, greedy users capable of modifying protocols may get much more bandwidth than users relying on standard protocols.

Guaranteed Frame Rate (GFR)

The most recent development to ATM services is GFR. According to Report Q7/13 of the ITU documents, the main motivation behind GFR is that some applications may not be best suited for any of the ATM-transfer capabilities described earlier. Such applications are too bursty for CBR, have traffic characteristics that are not suitable for VBR, and cannot use explicit feedback as in ABR.

The main advantage of GFR over UBR is that it provides a minimum guaranteed frame rate for every connection. Furthermore, new signaling messages are needed for establishing the reservations. Although the standardization is still unfinished as of this writing, and it is not totally clear what the actual meaning of guarantee is with regard to GFR, it is likely that there will be strict rules for controlling GFR connections. Yet, the rules may be looser than those of other ATM services. Because of the inherent vagueness of the applications of this service, however, the design of an optimal and mathematically accurate control method might be a very laborious process. In general, it seems that a combination of applications with unpredictable traffic patterns, loosely defined control mechanisms, and guaranteed services is difficult, if not impossible, to realize.

A short summary of the basic engineering philosophy of ATM is as follows:

- For most of a network, the basic unit for traffic engineering is a connection—that is, a continuous flow of cells.
- ATM provides two levels of aggregation (VC and VP) that may facilitate traffic management.
- An ATM network offers guaranteed services for most of the connections.
- The rest of the capacity is divided among UBR, GFR, and ABR service categories, suitable for adaptive applications.
- An ATM network favors statistical multiplexing to improve network utilization even at the expense of complicated control architecture.

2.2.3 Evaluation of Connection-Oriented Approaches

It can be argued that if the philosophical basis of ATM is the right one, the overall result cannot be much better than what ATM technology offers independently of the amount of effort put in to develop the service architecture. It is impossible, however, to be certain that the starting point is totally relevant with regard to the Internet. Hence, it's important to consider the requisite attributes—fairness, versatility, robustness, and cost efficiency—when assessing this issue. Although this evaluation concerns mainly ATM networks, most of the issues are common to any connection-oriented technology.

Versatility

Several arguments can be made for the superior versatility of ATM technology:

- Five different service categories can meet, in principle, almost any imaginable service need. More specifically, the rt-VBR service (and CBR as a special case of it) can provide superb real-time characteristics, and ABR and UBR are designed for adaptive data applications.
- The network can distinguish each individual connection and give everyone a network service with user- or application-specific characteristics, including appropriate bit rate. Therefore, the bit-rate granularity problem of circuit-switching systems can be solved exquisitely by ATM.
- The advanced traffic-control functions make it possible for the operator to adjust and optimize the use of network resources in a flexible manner. Thus, the use of virtual paths enables the network operator to handle a lot of connections inside the network without detailed information about individual connections.

Further, it is possible, at least in theory, to provide various levels of cell-loss ratios within the VBR service categories. There is even a bit in every cell reserved for dividing cells of each connection into two cell-loss categories: Cell Loss Priority (CLP). It might also be possible to provide two different virtual paths within one link in a way that the cell-loss ratio is different. In practice, this kind of system is cumbersome to implement and manage. (This is discussed further in Chapter 5, "Differentiation of Customer Service.")

Although it can be argued that ATM operators or service providers will rarely actually use all these service categories and traffic-control features, the overall conclusion is that there are not many problems related to versatility. Besides, it seems that the standardization organizations are able to develop new standards if any deficiency is identified.

Robustness

Robustness is the other area to which ATM standardization has paid a lot of attention. If this discussion ignores, for a while, the UBR service category, all the other ATM services are designed in a way that definitely restricts the possibility of misuse of the network. The traffic contract between a user and a network specifies in detail, on the one hand, what the user is allowed to send into the network, and, on the other hand, what performance the network promises to offer for compliant connections. There is not much more room for misuse than in circuit-switching networks.

Of course, the principle of statistical multiplexing results in some level of uncertainty. It is theoretically possible, for example, that a large number of users exploiting VBR service will synchronize their transmission in such a way that the momentary load exceeds the network capacity even though every individual user is complying with the traffic contract. In general, any traffic-control mechanism relying on statistical properties of traffic behavior can be challenged by an intentional attack from malicious users. It is possible to limit this kind of threat with well-defined customer services and by appropriate network dimensioning.

Another possible, and perhaps more serious, concern relates to the inherent complexity of ATM traffic engineering. It is apparent that the more parameters to be specified, the more possibilities for errors. Errors can be made either by users when defining their requirements or by network operators when specifying the characteristics of networks services. Consider, for example, what you would think if you put an extra zero in the required bitrate box (say, 500kbps rather than 50kbps) and then received a bill 10 times more expensive than what you expected? Correspondingly, a reverse error made by a network operator may wreak havoc on the performance of a whole service category. If there are dozens of parameters, as ATM service categories in total have, it is very likely that something will go wrong (because of the complex architecture and the large number of parameters).

Finally, it is important to say something about the robustness of UBR service category. UBR is, in this respect, contrary to other ATM services: Complex service architecture does not induce any problems, but the lack of strict traffic control might. If all other service categories can be insulated from the effects of excessive UBR traffic—and ATM traffic management surely provides tools for doing that—the problems can be kept on an appropriate level. Further, the users of UBR service will comprehend that the cheapness of the service is directly related to the lack of any strict service guarantees.

Fairness

Fairness is an elusive term. Because this is the first time that this discussion is attempting to thoroughly assess the fairness of a service model, it is important to first consider this central issue more generally. It is possible to clarify some issues by limiting the viewpoint to a specific case. You have bought a service from a service provider for transmitting information through the network, for example. The structure of the service could be of any form, simple or complex. One thing is certain, however: You must pay for the service. For the sake of simplicity, assume that you get a monthly bill that consists of either a constant flat rate or a very complicated composite of separate fees. Other customers get similar bills, or perhaps dissimilar bills because of a different service model, every month. The most essential issues to consider when assessing the fairness are as follows:

- Total amount of payment
- The service you and other customers have obtained
- Clarity and predictability of both the service and the bill

Note

The viewpoint of this book is that the structure of the bill is of minor significance. This book assumes that the customer doesn't much care whether the monthly invoice consists of several itemized charges or of one flat rate (only that the content should be, in any case, understandable).

Another issue to consider is the relative services obtained by different customers compared to the service obtained by one customer and the costs related to realize that service. One would argue that the charge of a service could be unfair even though all customers get the same service with the same price. This is definitely possible—for example, the price of some telephony services seems to significantly exceed the real costs of the service.

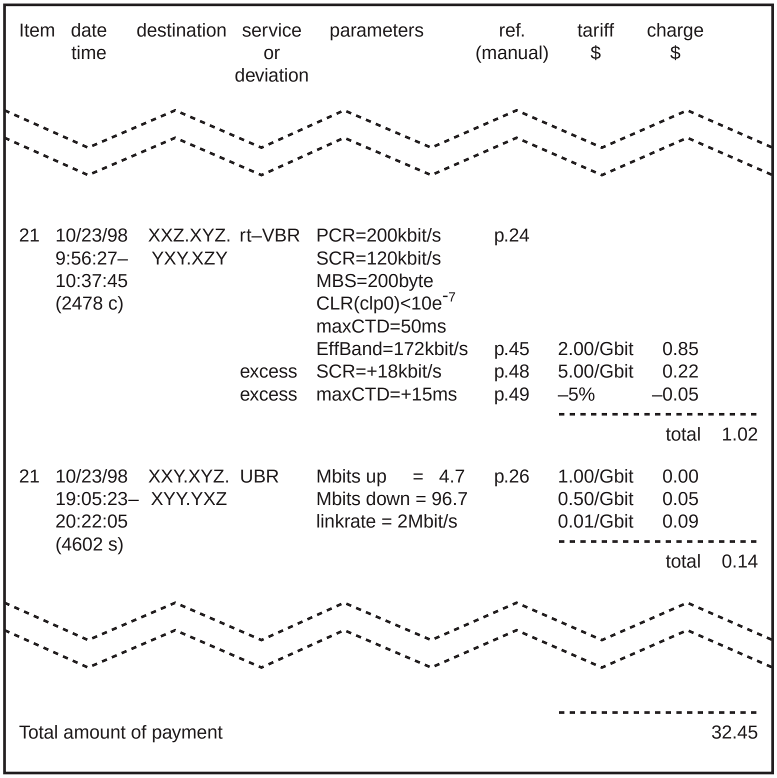
Nevertheless, these are new services with tough competition. It seems fair to suppose that the competition keeps the average price level of Internet services reasonable. In this case, it is important to ask the hard question: What is a fair price structure when service diversity is as wide as it is predicted to be in the future Internet?

What would be the result if you were to take as a starting point the monthly bill rather than the technical characteristics of service categories? You might get a monthly bill with an extremely detailed description of what you have used, something like the one depicted in Figure 2.3.

Note

It is not important to this example to understand all the terms appearing in the monthly statement in Figure 2.3. You can just suppose that an ordinary customer is not willing to read the handbook, and that it is too long to include here. There could be 100 items every month if the Internet service is used for Web browsing, telephone calls, file transfer, and various other applications. As a result, the total bill is hard to compress into fewer than 10 pages, and a customer needs to spend quite a lot of time to check the bill carefully.

Figure 2.3 Part of a fictitious ATM service bill.



Item	date time	destination	service or deviation	parameters	ref. (manual)	tariff $	charge $	
21	10/23/98	XXZ.XYZ.	rt–VBR	PCR=200kbit/s	p.24			
	9:56:27–	YXY.XZY		SCR=120kbit/s				
	10:37:45			MBS=200byte				
	(2478 c)			CLR(clp0)<$10e^{-7}$				
				maxCTD=50ms				
				EffBand=172kbit/s	p.45	2.00/Gbit	0.85	
			excess	SCR=+18kbit/s	p.48	5.00/Gbit	0.22	
			excess	maxCTD=+15ms	p.49	–5%	–0.05	
							total	1.02
21	10/23/98	XXY.XYZ.	UBR	Mbits up = 4.7	p.26	1.00/Gbit	0.00	
	19:05:23–	XYY.YXZ		Mbits down = 96.7		0.50/Gbit	0.05	
	20:22:05			linkrate = 2Mbit/s		0.01/Gbit	0.09	
	(4602 s)							
							total	0.14
Total amount of payment								32.45

What about fairness? Is there any problem? The service and the tariffs are designed carefully in a manner that could be considered as fair as possible. Every traffic and quality parameter has been taken into account; the price levels of different service categories have been pondered sincerely; and the hard competition takes care of the overall level of tariffs.

Although every detail seems to be fair, something in the whole system is inappropriate. Can you understand the bill in Figure 2.3 without looking at some ATM textbook? If not, don't be worried. A great majority of ordinary customers lose track in the first row of the bill and move immediately to the only figure that they certainly understand: the total amount to be paid.

Although the bill might be exhaustive, consisting of fair details, most customers cannot assess the service they get from their service providers or compare the price performance of different service providers. For these reasons, a scheme in which an ATM service provider offers all service categories to all users is not in reality a very likely approach.

Moreover, because one service provider can seldom offer connections to all required places, many important issues—such as the availability of service classes, pricing, and quality of service—depend on the approaches applied by other service providers and network operators. The reality of multiple providers makes the fairness assessment by an ordinary customer even more difficult.

What could be a solution to this disagreeable situation? There are, of course, various pricing approaches with different properties, as discussed in Section 5.1.3, “Pricing as a Tool for Controlling Traffic,” in Chapter 5, “Differentiation of Customer Service.” One feasible approach is to simplify the service construction as much as possible. The most concise, but still somehow feasible, structure is a combination of CBR and UBR (or GFR) services.

You can use UBR service whenever the network performance is high enough for your purposes; otherwise, you must use CBR service that is basically able to meet all imaginable quality requirements. In the simplest model, the price of the UBR service is based purely on your access rate into the network and CBR services are charged in the same way as ordinary telephone calls, except that the price depends on the required bit rate. Note that the basic philosophy of the GFR service model is quite similar to this combination of CBR and UBR services.

Can this simple scheme be fairer than the complex scheme with all ATM service categories? Let’s try to look at this fundamental question more thoroughly by supposing three groups of customers:

- Customers using CBR service
- Customers using UBR service with moderate demand of information transfer
- Customers using UBR service with high and continuous demand for transferring information

Although customers can change the group whenever they want, for this discussion suppose that customer groups are permanent during a month. Three main questions relate to fairness: Does this service structure enable relatively fair pricing for CBR customers, for the CBR-group and the UBR-groups, and for the two UBR-groups?

The first issue raises questions as to whether the tariff should depend on the destination address or on the date and time of the connection and whether the CBR tariff should be a linear function of the bit rate. Because the CBR service is similar to that of telephone service, many service providers with telephony background will answer *yes* to the first two questions. There is no evidently right or fair scheme, however, when looking at the technical costs of two CBR connections, a local connection, and a connection to a destination on the other side of the globe. The situation is too complicated and changeable to provide means for accurate evaluation. Markets, customer behavior, and regulatory issues will determine the situation in real networks.

There is no apparent answer to the linearity question either. A simple calculation shows that a linear tariff either limits the usefulness of CBR service for high-quality video transfer or offers free telephone calls:

- A three-minute telephone call with a bit rate of 20kbps generates 3.6Mb of traffic.
- A movie coded by 2Mbps and lasting 100 minutes generates 12Gb of traffic.

Consequently, a movie may generate 3,333 times more bits than a telephone call, which means that if the telephone call costs $0.003, transmitting the video will cost $10 when using the same linear tariff. Neither of these are reasonable: The cost of billing a telephone call is probably more than 0.3 cents, and $10 for transmitting a video through the network is a prohibitive price for most users. Therefore, although linear pricing could meet basic fairness criteria if judging by the technical realization of the service, it is not necessarily a reasonable solution in practice. (This issue is discussed further in the section titled "Price of Bandwidth" in Chapter 5, "Differentiation of Customer Service.")

The next topic is fairness of CBR pricing compared to UBR pricing. The same basic problem is encountered as in the case of different CBR calls: Customer willingness to pay depends much more on the usefulness or entertainment of the end application than on the number of bits transmitted through the network.

A tariff of $0.05 / minute for a 64kbps connection means $13 / Gb. If you assume that your UBR usage is moderate (say, on average 5MB per day), the same tariff means $15 / month. Because the price per bit for UBR service will be lower than that of CBR service due to lower quality, the result could be quite appropriate in this case—for instance, $5 / month for UBR services is a quite reasonable tariff.

But again, a linear-pricing model may bring about problems because customer willingness to pay is probably not a linear function of transmitted bits. How to avoid this problem? One possible approach is to ignore totally the transmitted bits and apply a pure flat rate. This approach may certainly solve some problems, but may also generate some new problems when assessing the fairness between the two UBR customer groups with light and heavy usage. This primary question of Internet services is addressed later in Section 2.3, "The Best-Effort Approach."

Cost Efficiency

It seems that a very versatile service provision, even though somehow desirable, is difficult to design and manage in a truly fair manner from an ordinary user's viewpoint. If you conclude that service structure should be as simple as possible, something like a combination of CBR and UBR categories, you have to ask whether ATM is the most efficient way to realize these services.

UBR service seems to be needed mainly for IP packet transmission. As mentioned earlier, the ATM overhead when transmitting IP packets is approximately 20%. In some cases, an overhead of this level is acceptable; sometimes, however, it is not acceptable. Another, probably more serious, source of extra costs is the management of an extra layer: ATM has its own signaling and routing and management systems. Therefore, if you consider only IP over UBR service, it is somewhat difficult to identify any compelling reason to use ATM between IP and the physical layers.

This picture changes significantly when you consider needs other than IP traffic—in particular, real-time applications such as voice and video. The cell-based switching and transmission and strict quality control of every connection are powerful tools to satisfy the most demanding requirements. The question now is whether this is enough to justify the large contraptions of ATM. If ATM, or any similar technology is used, you must first of all be certain that the quality-control scheme applied in ATM is exactly what you need.

Real Quality of ATM Services

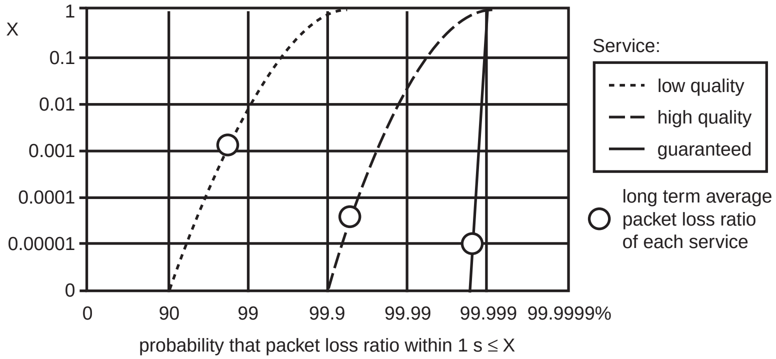
ATM traffic management and quality assurance are based on three cornerstones: virtual connections, capacity reservations, and quality guarantees. The terms and concepts of this system—connections, reservations, and quality—are easily misleading. (For example, reservation models and calculations are often based on certain assumptions that are not necessarily valid in reality.)

If you try to guarantee a definite cell-loss ratio, the actual result in practice could be like that shown in Figure 2.4. The figure illustrates a situation in which a service operator offers three service classes: low, high, and guaranteed. They are treated using two basic principles:

- All packets are delivered if possible.
- During congestion, packets belonging to higher service classes are delivered whenever possible (ahead of lower classes).

It is probable that most of the time (90% in the figure) there is enough capacity to transmit all packets. Consequently, if you consider a short period, the probability that there is no packet loss even on the lowest service level is 90%. Then during some high peaks of traffic load (or traffic variation), some packets belonging to the intermediate class should be discarded; all guaranteed service packets, on the other hand, can still be successfully transmitted. The highest class will suffer packet losses only if something exceptional happens—for instance, a cable break. Even so, the loss ratio is likely very high. Note that the unavailability value of 0.001% means only 5 minutes per year.

Figure 2.4 The difference in real service quality of three service classes.



Exceptional cases may occur, for instance, due to operational errors that are more likely when the system is complicated, as ATM is. If you want to provide several QoS classes, there will be a real difference between classes perhaps only 0.01% of the time. This is always a likely result when a network relies on reservations and preventive traffic engineering.

The difference in quality does not mean so much difference in cell-loss ratio, but rather differences in reliability or availability of the service. For most applications, the service is either available or not; seldom is the quality only moderate. For most end users, the real reason for the unavailability of service is usually of no importance. Most end users will want to know when the service will be available again, not the "fascinating" reason(s) for temporary unavailability—excessive load in the network, cable break, management error, and so on.

2.3 *The Best-Effort Approach*

There are actually two traffic management philosophies: In the first one, traffic management is needed only during congestion; in the other one, the main task of traffic management is to avoid congestion whenever possible. With ATM, the engineers had something permanent in mind, such as long video connections with relatively stable bit rate demand that are totally independent of what is occurring in the network. Congestion avoidance is a reasonable approach in that case.

On the contrary, in IP or Internet, the fundamental idea has been almost the opposite: Most of the traffic is anything but stable, and there are inherent relations between network capacity, load situation, and traffic demand. Therefore, the starting point of IP has been that if there is no congestion, no traffic-control actions are needed. It is important now to clarify the main reasons that have led to this traffic-control paradigm in IP networks.

2.3.1 Service Model

The foundation of Internet technology has been the assumption that packet switching is much more suitable than circuit switching for computer networks. The Internet has shown that this assumption is valid. However, the technological differences between packet and circuit switching do not totally explain the remarkable differences in the history of the Internet and telecommunication networks.

Part of the difference stems from the amount of time that each has taken to develop. There has been much time for building bureaucratic standardization and development processes since the invention of telephony in 1876. In contrast, the development of the Internet during the first 20 years was a much less bureaucratic, and a much more flexible, process.

The Development of the Internet

The Internet started as a research project connecting four computers in 1969. The experimental network, called ARPAnet, was funded by Advanced Research Projects Agency (ARPA), now called Defense Advanced Research Projects Agency (DARPA), an agency of the U.S. Department of Defense. Since then, a lot has occurred:

- The number of computers has grown steadily, by approximately 75% per year.
- The Request for Comments (RFC) series was established in 1969.
- The first email system was introduced in 1972.
- Wide deployment of TCP/IP began January 1, 1983.
- First IETF meeting took place in January 1986 with 21 attendees.
- Tim Berners-Lee at CERN (*Conseil European pour la Recherche Nucleaire*, translated as European Laboratory for Particle Physics Research) invented the World Wide Web in 1990, which added graphics capability to the Internet and positioned the network to become a vehicle of commerce.
- The Internet Society was founded in 1991.
- The number of computers connected to the Internet exceeded 1 million in 1992.
- During the past few years, TCP/IP has become the dominating networking protocol.
- The number of attendees to the forty-second IETF meeting, in August 1998, was 2,106.

continues

A few key principles have guided the evolution of the Internet:

- Open architecture means that the network architecture does not dictate the use of any network technology, but rather the provider may select it freely.
- The simplicity and robustness of the system has been promoted by specifying that the network nodes do not keep any information about the individual flows of packets passing through.
- The Internet has not been designed for just one application, but as a general infrastructure.

These principles distinguish the Internet from most other networking standards.

For further information about the history of the Internet, see `http://www.isoc.org/internet/history/`.

In traditional telecommunication networks and services, the specification and implementation phases are clearly and separately defined. With regard to the Internet, however, specification work and implementation proceed parallel. This is explicitly stated in RFC 2026: “An Internet Standard is a specification that is stable and well-understood, is technically competent, has multiple, independent, and interoperable implementations with substantial operational experience, enjoys significant public support, and is recognizably useful in some or all parts of the Internet.”

Therefore, an Internet document may reach a standard status only after there are independent implementations. In addition, it should be noted that the standardization body, the Internet Engineering Task Force (IETF), is a loosely self-organized group of people who make technical and other contributions rather than a hierarchical organization with official representatives from different organization. Basically, the same people who are the most intensive users of the Internet are participating in the standardization effort (and may as well be involved with the operation of the network). Although this situation has changed somewhat as the user population has expanded, it is safe to say that Internet engineers are still developing standards for themselves.

It is, therefore, somewhat artificial to speak about customer service in the case of former IP networks. The engineering philosophy was based on the model of a homogeneous community that had common interest to design a workable network rather than on a model of service providers and customers.

The fairness of the Internet service, or more generally the fairness of the whole Internet, has relied on the assumption that there is in essence one user group consisting of all

Internet users. In that case, the fairest situation is when everyone is allowed to use the network for any sensible purpose, and only when there is not enough capacity for all demand, would there be a need for controlling or limiting the traffic sent to the network.

Even during congestion, it is supposed that all or at least most users behave agreeably. Agreeable behavior could be that users stop transferring enormous files if they notice any performance problems in the network, or even better, if everyone sends only truly necessary information through the network. The situation was earlier eased by the fact that transferring information through the network was a much more complex operation than nowadays; back then, only persons with some level of experience in the field of data transmission sent much data through the network. It is evident that this kind of approach has serious limitations when the population contain tens of millions of users and the use of the network becomes a simple task for anyone (even those without any knowledge about data networks and protocols).

The next step has been to specify protocols that automatically adjust the sent traffic to the network. If everyone is using a similar protocol and does not evade the adjustment control by using other more greedy protocols, the system could partly solve the problem of different user behaviors, because most users are neither able nor willing to modify any traffic-control protocols.

Within these limits, any user who has been connected to the network has been allowed to utilize any available network resources independent of the actual purpose of the application or information. The network then provides a service that is called *best effort* because the network tries to transmit as many packets as possible and as soon as possible but does not give any guarantees. As a result, the realization of best-effort service consists of three main parts:

- The network transmitting packets
- The TCP protocol controlling the bit rate
- The application capable of working in changing conditions

2.3.2 Basic Best-Effort Service Based on TCP

Jon Postel wrote the Transmission Control Protocol specification, RFC 793, in 1981. It is worth noticing what was said about the objective: “This document focuses its attention primarily on military computer communication requirements, especially robustness in the presence of communication unreliability and availability in the presence of congestion, but many of these problems are found in the civilian and government sector as well.”

Because of this background, TCP provides an effective tool to recover from data that is damaged, lost, duplicated, or delivered out of order. This is achieved by assigning a sequence number to all data transmitted in the network, and requiring a positive acknowledgment (ACK) from the receiving TCP. If the ACK is not received within a timeout interval, the data is retransmitted. As a result, if all TCP implementations function properly and the Internet does not become completely partitioned, TCP is able to recover from transmission errors.

Moreover, TCP provides a means for the receiver to control the amount of data sent by the sender. This property is achieved by returning a “window” indicating a range of acceptable sequence numbers. The window indicates an allowed number of octets that the sender may transmit before receiving further permission.

These characteristics are specified in the original TCP document. The basic TCP scheme does not, however, provide reasonable tools for efficiently avoiding or alleviating congestion situations inside the network. In a worst-case scenario, a combination of retransmissions and a rapidly growing load in congested links may lead to a so-called congestion collapse.

The situation may start when a new file transfer begins to fill a buffer assigned to an already loaded link. When this buffer fills up, the round-trip time for all connections rises quickly. In that case, TCP connections suppose that packets are lost, and retransmit them. Finally, several copies of the same packet may exist at the same time in the network. Consequently, the throughput of the network is permanently reduced to a small fraction of normal. This problem was addressed by RFC 896 in 1984 and various proposals to solve it, such as RFC 2001, have been presented thereafter.

2.3.3 *Improvements to the Basic TCP Behavior*

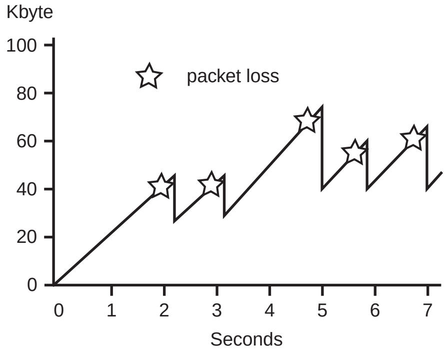
The fundamental problem of the old TCP implementations based on RFC 793 is that the sender may start a connection by sending lots of data up to the window size advertised by the receiver. Although this simple scheme may work in small networks with large capacity, it may be harmful in large networks with several routers and possibly low or highly loaded links.

Slow-start is a solution to this problem. In essence slow-start means that after connection establishment, the sender is allowed to send only one packet before getting acknowledgement from receiver (supposing that the sender is using the packet size announced by the receiver). When the sender receives ACK, the sender can double the amount of packets to be sent until a packet is discarded and the sender notices that the maximum available capacity in the network is reached.

Congestion can be alleviated by going into a slow-start when the sender notices a congestion situation in the network. Several different schemes are used to increase packet rate after congestion. This discussion does not address them further, but it should be noted that they all induce a sawtooth pattern in which the window size (and also the bit rate) goes regularly up and down, as illustrated in Figure 2.5.

All connections also encounter intermittent packet losses when the total load exceeds link capacity and the buffers get full. Note that there is a delay between packet loss and window size reduction because of the round-trip delay. Both the sawtooth and packet losses are intrinsic characteristics of TCP and usually are insignificant to most end applications.

Figure 2.5 Sawtooth pattern of a TCP connection.



An elementary part of the congestion problem is that the network nodes have applied a pure first in, first out (FIFO) principle in the buffers—packets are discarded only when the buffer is totally full. If most network nodes are built with FIFO buffers, TCP flow-control algorithms are about the best that can be done (Keshav 1998, 421). Although a FIFO principle may seem to be the most efficient and fair in general, in certain situations it is both inefficient and unfair.

A FIFO buffer yields a similar packet-loss ratio to every connection at certain point in time. When measured during a short period when the buffer is full, the packet-loss ratio could be very high, and consequently, almost all senders get notice of congestion at the same time. If they are also reacting at the same time, the total traffic will drop dramatically. Then for a certain period of time, the congested link will be underutilized because every connection begins to increase its packet rate from a low value.

One possible solution to this problem is that some randomly selected packets are discarded even before the buffer becomes full. In that way, some senders are informed about the imminent congestion in the network. Because the senders are not synchronized, it could be possible to keep the bottleneck link utilized most of the time. The selection of discarded packets can be totally random, or some more complicated procedure can be applied. These mechanisms are discussed thoroughly in section 5.3, “Traffic Handling Functions in Interior Nodes,” in Chapter 5, “Differentiation of Customer Service.”

Although it seems possible to develop TCP protocol and buffering algorithms in a way that maximizes the network utilization, there remain some serious difficulties with fairness. If and when well-behaved TCP connections must live with other connections with different behavior, the final result could be that TCP connections are continuously in the slow-start phase, and aggressive connections without any adjusting mechanism seize most of the capacity.

Is there any means to alleviate this situation? Yes, if you are ready to distinguish individual flows somehow inside the network. During a congestion situation, you should discard packets from those connections that have exploited most of the resources and leave other connections alone. There is not any obvious way to select the discarded packets, however, particularly if you want to share the network resources based on individual traffic contracts rather than on even shares. Actually, this is one of the fundamental questions of the Differentiated Services approach.

2.3.4 Evaluation of the Best-Effort Approach

The best-effort approach has been a successful service model for the flourishing Internet. Why should we change one of the cornerstones of such a successful technology? One plausible opinion is that we actually should not do that, but we only have to increase the network capacity as quickly as possible without changing the best-effort service model. The reasoning behind any new, likely more complicated model has to be strong and clear; a mere vague idea that best effort is not satisfactory for the future is not enough.

Chapter 1, “The Target of Differentiated Services,” introduced “attributes” exactly for this purpose—that is, to facilitate the analyzing of different approaches. The attributes—cost efficiency, versatility, robustness, and fairness—are used in the next four sections to look at the best-effort approach from different viewpoints. Cost efficiency gives emphasis to the economical aspects; versatility stresses the various needs of future applications; robustness and fairness shed light on the issues related to the intrinsic weakness of a service model based on the TCP protocol.

Cost Efficiency

One potential efficiency problem of the best-effort service model using TCP as a control method is that at the bottleneck node, some packets are always lost because the algorithm detects overload situations using discarded packets. You would argue that a lost packet always means wasted resources. In a sense, you are right: Some resources are used to transmit the packet to the bottleneck node. Despite this fact, it is fair to infer that in a simple situation with only one bottleneck, no significant resources are wasted.

In any modern telecommunication system, the actual costs are practically independent of the traffic load if the infrastructure and amount of customers are fixed (personnel, electricity, and so on) and costs are constant. What is the real nature of costs in telecommunication networks then? Definitely they depend somehow on the traffic, and lost packets are considered part of the traffic load. Traffic load can be related to costs is two ways.

First, the network dimensioning is based on the offered traffic load, and perhaps on the packet-loss ratio as well. If the load exceeds a certain limit, you update the network by acquiring more capacity—and that definitely entails costs. Because you are aware of the nature of the TCP mechanism, however, you should not be too hurried to buy new capacity if a moderate amount of packets are lost. A "normal" packet-loss ratio is acceptable and does not imply a need to expand the network. Only if the packet-loss ratio exceeds a certain higher threshold is it an indication of insufficient capacity. Therefore, there is not necessarily any direct relation between wasted packets and costs.

Second, a potentially more important issue is that a packet lost in the bottleneck node has used link and buffer capacity somewhere else in the network and, therefore, may give rise to an unnecessary packet discarding in those points. But that happens only if there is another bottleneck in the route of the packet, and at the same time there is a suitable packet to be transmitted through the network. Although this kind of situation may induce additional costs, it seems that under normal traffic conditions the total effect is negligible. This issue is discussed further in Chapter 7, "Per-Hop Behavior Groups," and Chapter 8, "Interworking Issues," because it is common to most of the Differentiated Services schemes.

It is fair to conclude that best-effort service based on TCP control makes possible highly efficient networks. In addition, the network costs seems to be low because no signaling is required, and a relatively simple buffering system gives satisfactory results; even a pure FIFO is workable. But this assessment is valid only with adaptive applications that can utilize the intrinsic characteristics of the service.

A lot of applications cannot do that, however; if you want to satisfy the needs of those applications, you must keep the overall load level in the network so low that packet losses are rare and delay variations small. In that case, best-effort service is not technically efficient because of low utilization; it can be more cost efficient than a complicated system, however, because of low implementation and management costs.

Versatility

The lack of versatility is one of the key questions related to best-effort service—and one of the fundamental questions of the whole effort of Differentiated Services. Versatility can be divided into several aspects: bit rates, delays, packet-loss ratios, and network environment.

As to the bit rate, best effort can be applied with any bit rate, low or high, constant or variable; there are no definite limits for granularity. The problems are related to the other aspects. It could be possible to devise a real-time best-effort service applying a similar mechanism to TCP. Unfortunately, some fundamental problems arise with this approach. A workable best-effort implementation requires that buffers be big enough to handle the bursty TCP connections; with very small buffers, the system does not work efficiently. However, if a large buffer is really used, it also means long delay unless the bit rate is very high.

Therefore, the basic best-effort service cannot properly support truly real-time connections except if the load level is so low that buffers are continuously almost empty. In practice, real-time service requires additional tools to be feasible, such as its own buffers and proper buffer management inside the nodes. Because TCP counts on packet losses to adjust bit rate, it cannot offer loss-free service or different levels of loss ratios. This kind of service is beyond the scope of the basic best-effort model, but surely belongs to the field of Differentiated Services.

It is also reasonable to ask whether TCP is suitable in all network environments. In most cases, it is; this fact is comprehensible if you remember the basic target of TCP including potentially unreliable networks. Nonetheless, one area of networks causes problems to TCP connections: wireless networks. In most current transmission systems, the bit error rate is very small. Therefore, the main reason for lost packets is congestion, just as the TCP mechanism assumes. On the contrary, in wireless networks bit error rate could be occasionally high and cause packet losses because every packet with bit errors is discarded. Consequently, TCP supposes there is severe congestion and moves into slow-start phase. Chapter 8, “Interworking Issues,” addresses this issue.

Robustness

One severe problem of TCP-based traffic management is that the TCP protocol is usually running in customer equipment and, therefore, not within the direct control of the network operator or service provider. As a result, the boundaries between network service and applications are considerably blurred, which makes it difficult to provide a consistent network service.

The current situation is that a main part of the traffic on the Internet utilizes only a couple of different TCP implementations, and that a large majority of users are using them without any modifications. Unfortunately, this situation leaves the field open for rogue users who try to maximize the bandwidth they attain from the network—and in a worst-case scenario, intentionally interfere with the normal network operation. Therefore, although best-effort service works well in many conditions, the whole service structure is susceptible to rogue users and new applications with different requirements.

Fairness

When you want to offer higher-quality connections for some customers, you need tools to at least limit the effect of different TCP implementations on the best-effort service class, and if possible, to also limit the effect of mischievous users within that class.

Internet users can be divided into two primary groups to assess fairness: ordinary users with no or minor knowledge about Internet technology, and skillful users with considerable ability to tune their computer systems. The latter can still be divided into two subgroups: friendly and harmful. Friendly users, even though they possess harmful potential, are chiefly interested in just getting somewhat more capacity than ordinary users from time to time, but without a desire to damage the network. Harmful users, who are unfortunately not unknown on the Internet, may instead try to abuse networks resources (sometimes even regardless of how much real benefit they actually get themselves).

As for the best-effort service, the group of unskillful users is usually not problematic; and similarly, most users belonging to the friendly expert group are not a threat as such. If every user is behaving appropriately, the best-effort service is a feasible approach within its intrinsic limits. The main threat seems to be that a programmer devises an innovative product that does not need much expertise to use but that significantly improves the bandwidth the user is getting compared to other users. This kind of product could become so popular that most experts, friendly or not, will exploit it.

In the worst case, this may decline the service of ordinary users and, therefore, impede overall customer service. Unfortunately, this seems to be possible because of weak or nonexistent control mechanisms at the user-network interface. If this happens on a large scale, it does not only deteriorate overall fairness but also deteriorates the service of all users. One of the areas in which this may happen is multicasting applications sending real-time audioand video streams.

2.4 Integrated Services Model

The history of *Integrated Services* can be traced to the Birds of Feather (BOF) session, “The Real-Time Packet Forwarding and Admission Control BOF,” in November 1993. The first sentences of the BOF minutes stated: “The demand for multimedia communication and the success of IETF audio / videocasts will soon create an urgent requirement for resource reservation and control in the Internet. From an architectural viewpoint, this represents a new Internet service model." (For more information, visit the Integrated Services mailing list archive at `ftp://ftp.isi.edu/int-serv/int-serv.mail`.)

This statement defines the main area of concern: real-time audio and video multicasting services. It was recognized that these services could not be properly supported by the basic best-effort mechanisms. From the very start, some fundamental questions were discussed:

- Why do we need a new service model?
- What should the fundamental nature of the service model be, explicit or implicit?
- Is admission control necessary?

As to the last issue, the primary philosophy of the Working Group was that occasional blocking of a connection request is a more economical approach than vast over-provisioning. That is the whole point to resource reservations and the guaranteed service model adopted by the Integrated Services Working Group.

It was observed that behavioral characterization of functionality is a very difficult intellectual problem, and that it was important that the community not get bogged down in this exercise. It seems, unfortunately, that this very intellectual problem is still unresolved. In the Differentiated Services effort, the behavioral characterization of functionality is one of the fundamental issues, and yet real experience is required in the same way it was required during the first phase of Integrated Services five years ago.

The Integrated Services Working Group focused on defining a minimal set of global requirements that would transform the Internet into a robust, integrated-service communication infrastructure, including the following issues:

- Defining the services to be provided
- Defining the interfaces between application and network service, routers, and subnetworks
- Developing router validation requirements

In January 1994, Bob Braden expressed a concern about poor activity in the Integrated Services mailing list; this was a somewhat premature concern, because four years later the mailing list archive consisted of more than one million words. In addition, a part of the effort, namely the Resource Reservation Protocol (RSVP), has been discussed on a separate mailing list. The following sections outline the results of these activities.

2.4.1 Customer Service

One interesting theme of discussion when the Working Group started was the importance of convincing the public at large that IP is suitable for Integrated Services. Although making major technological developments is difficult, it can be much more difficult to change public opinion. When the public has experienced a moderate level of Internet service with regard to quality and reliability for several years, you may encounter severe difficulties when attempting to ensure people that some Internet services can be both reliable and of a good quality.

If you want to offer high-quality telephony service over the Internet, for example, you will certainly meet a lot of doubts about the reliability of the service. Your customer is not likely to assess technical details; instead, he or she will compare the current telephony service with the current Internet service in general—and, right now, customers perceive a big difference. Although you may deem this unjust from an operator's viewpoint, you must face reality (and reality does not consist of technical facts only, but also of opinions).

So what is the right reference point for high-quality Internet service? The service that all Internet users are familiar with is ordinary telephony service. The current situation, in most developed countries, is that you practically always get a telephone connection with the same quality. The quality, albeit definitely sufficient for most purposes, is not actually very high; this is evident if you listen to classical music on the telephone. The strongest feature of digital telephone service is predictability: You can obtain the same service independent of time, date, location, or distance.

Note

The characteristics of mobile services are somewhat different, with some reliability and quality problems. The success of mobile services strongly indicates that users can cope with a lower level of quality, provided that the service can offer something unique. In this case, the uniqueness is mobility. Therefore, each ISP must find and define the uniqueness of its service offering.

Customer service—composed of both high- and moderate-quality parts—must, consequently, be credible in its good characteristics. One possibility is to build the highest-quality service on a mathematically provable basis. If you select this option, you clearly are aiming to compete with the current services with their own field. It will be very hard to surpass the delay or loss characteristics of circuit-switching networks or CBR service in ATM networks even with mathematical proofs.

Do the basic attributes—fairness, robustness, versatility, and cost efficiency—offer any clue about what could be the competitive strength of Integrated Services as a *customer service*?

Perhaps *fairness* could be the issue. Telephone operators now have several years of experience with customer expectations and competitive markets. So, there is probably not much opportunity to gain a marketing edge in this area. Because the Integrated Services must rely on the same infrastructure as the current Internet, it is not likely that *robustness* can be the main marketing point of integrated services on the Internet.

As to *versatility*, it may indeed offer real possibilities. Although the telephone network is very reliable, it is also inflexibly in the sense that totally new service features, if feasible at all, require complicated and cumbersome standardization processes. The problem of the ATM network is the lack of much real end-to-end ATM services. What is the meaning of versatility if it does not reflect on customer services? On the contrary, the Internet and the applications used through it are famous for rapid and innovative development.

The other potential advantage of the integrated-service model is *cost efficiency*. This advantage, however, remains unclear (because of the difficulties of identifying and assessing all the associated costs) until there are widespread implementations. The technical foundation of Integrated Services is likely to be at least as cost efficient as any other corresponding technology; whether it can offer cost savings related to the major cost sources, such as network operation, management, customer care and billing, is not so sure.

2.4.2 Implementation of Integrated Services

RFC 2215 defines the set of general control and characterization parameters used in the Integrated Services framework. Each parameter has a common definition across all QoS control services. For instance, `NON_IS_HOP` provides information about the presence of network nodes that do not support QoS control, and `AVAILABLE_PATH_BANDWIDTH` provides information about the available bandwidth along the path.

From the traffic management viewpoint, the key parameter is `TOKEN_BUCKET_TSPEC` (or the shorter `TSpec`) that describes traffic parameters using a token-bucket mechanism. (For more details, see the section titled "Measuring Principles" in Chapter 5, "Differentiation of Customer Service.") Data senders use this parameter to describe the traffic they expect to generate; the purpose is exactly the same as that of traffic parameters in ATM networks. `TSpec` uses the parameters shown in Table 2.2.

Table 2.2 Tspec Parameters

Parameter	Description
`b`	Token bucket with a bucket depth
`r`	Bucket rate
`p`	Peak rate
`m`	Minimum policed unit
`M`	Maximum datagram size

There are two IP-specific parameters: resource allocation and policing. All IP datagrams less than size `m` are treated as being of size `m`, and maximum packet size defines the biggest packet that can conform to the traffic specification.

Guaranteed Service

One of the first Internet drafts already stated that a guaranteed service shall provide firm, mathematically provable guarantees that the end-to-end delay experienced by packets in a flow will not exceed a set limit. This basic philosophy has been realized by RFC 2212, "Specification of Guaranteed Quality of Service."

A *guaranteed* QoS flow is specified by two sets of parameters: traffic parameters (`TSpec`) and service-level parameters (`RSpec`). The reservation specification, `RSpec`, consists of a data rate (`R`) and a slack term (`S`). In addition, two error terms, `C` and `D`, which describe the accuracy of the implementation compared to a perfect one, characterize the implementation of guaranteed service. Users can compute the maximum delay for a packet transmitted through the path by combining the parameters from the various service elements in a path. This discussion does not, however, go into the details of this calculation because it is quite complicated.

As a result, if the QoS control defined in RFC 2212 is deployed widely enough in the network, guaranteed service gives applications considerable control over their delay. Delay has two parts: a fixed delay and a queuing delay. The *fixed delay* is a property of the chosen path, which is determined not by guaranteed service but by the setup mechanism. Only *queuing delay* is determined by guaranteed service. In other words, an application can usually accurately estimate, *a priori*, what queuing delay guaranteed service will likely promise. If the delay is larger than expected, the application can modify traffics token bucket and data rate to achieve a lower delay.

Controlled-Load Service

The key pronouncement of the controlled-load service specification, as stated in RFC 2211, is the following: "Controlled-load service provides the client data flow with a quality of service closely approximating the QoS that same flow would receive from an unloaded network element, but uses capacity (admission) control to assure that this service is received even when the network element is overloaded."

What does this actually mean, and what is the motivation for this somewhat peculiar definition of a service? As already discussed, best-effort service may offer high quality provided that the network is slightly loaded. It is impossible to give any exact generally applicable numbers, but probably a 5% load is low enough even if the traffic variations are high. (A higher load is acceptable if variations are moderate.) Therefore, if you can give a higher priority for certain flows and limit the traffic and traffic variations of those flows, you might be able to offer a high, albeit not perfect, service with a relatively simple mechanism. Basically three components are needed:

- A prioritization mechanism to separate controlled-load flows from pure best-effort flows
- A mechanism to allocate appropriate resources inside the network to the flows
- Traffic control to limit traffic and traffic variations

Users requesting controlled-load service give an estimation of the data traffic they will generate: the `TSpec`. The service provider ensures that a very high percentage of transmitted packets are delivered successfully and that the delay does not greatly exceed the minimum delay experienced. The controlled-load service does not make use of specific target values for control parameters—such as delay or loss—so the service philosophy is better than best effort, but without any hard guarantees.

If the traffic of a flow exceeds the limits specified by `TSpec`, the flow obtains a similar service, but not necessarily exactly the same, as best-effort flows with the possibility of long delays and dropped packets. So the transition from *best-effort* to *controlled-load service* is a relatively easy operation for users. Moreover, because the specification is quite spacious, the network implementation may either rely on low utilization, traffic measurements to predict traffic behavior or on strict traffic control and accurate calculations.

Resource Reservation Protocol (RSVP)

The Integrated Services architecture enables users to request a higher quality than that of the best-effort service. In addition to the service specifications including the requirements for network element behavior, there has to be a mechanism to communicate the requirements to the network nodes along the transmission path. The Resource Reservation Protocol (RSVP) is designed for that purpose.

Basically, RSVP is doing the same task that is accomplished in connection-oriented networks by signaling. Several significant differences stem from the different starting points (the Internet is connectionless while traditional signaling is used in connection-oriented networks) and the different primary uses of the reservations (multicast applications, in the case of RSVP, but most connections in traditional networks are point-to-point). When compared to signaling in connection-oriented networks, such as ATM, the most prominent characteristics of RSVP are as follows:

- In RSVP, the receiver rather than the sender generates reservations.
- RSVP requires that the reservation be refreshed about once every 30 seconds; a permanent reservation, on the other hand, is explicitly finished.

- An RSVP receiver may modify the requested QoS at any time; usually the QoS is permanent for the life of the connection.
- In RSVP, the establishment of the route is an independent process; traditionally, however, the reservation and routing are concurrent.
- RSVP allows heterogeneity in trafficparameters, a characteristic not usually provided in any traditional networks.

Although all these characteristics could be reasonable and useful, they may complicate the cooperation with other networking technologies, such as ATM. For more information about RSVP, see the corresponding RFCs: 2205, 2210, and 2380.

2.4.3 Evaluation of the Integrated Services Model

Although the overall characteristics of Integrated Services are similar to those of corresponding ATM services, it is important to briefly assess the main attributes of the Integrated Services model.

Versatility

There seem to be no major problems, although there is a kind of gap between guaranteed services and best-effort services. In general, a technical standardization body like IETF is not necessarily the best organization to define services because its viewpoint could be too limited. Service providers and customers should have a more integral role in the service specification. Nevertheless, the Internet is such aversatile and flexible technology and environment that the possible gaps can likely be filled (one of the tasks left to Differentiated Services).

Fairness

One of the main problems with the Integrated Services model is that it seems to be difficult to build a reasonable—that is understandable and consistent—customer service. This is a complicated task with both technological and marketing challenges. The additional concern with the Internet is that a large part of the public does not deem the Internet reliable and, as a result, may have serious doubts about high-quality service offered as part of service selection.

Robustness

Because of the inherent mathematical basis, if the service is properly implemented and managed, significant problems are not probable. If the operators and service providers do not possess enough experience in this field, however, they may have big problems with reliability, quality of service, and network performance.

Cost Efficiency

Cost efficiency seems to be the main concern of the integrated-service model. The original intent was to solve primarily a rather limited problem of audio and video multicasting services. As time went on and the Internet changed, however, the objective apparently became more extensive. It is not, however, reasonable to assume that the relatively complex and heavy Integrated Services system with all the parameters and per-flow reservations can be used with most of the millions of flows traversing the Internet continuously. In short, the Integrated Services model has scalability problems.

2.5 Targets for Differentiated Services

The assessment of the other technologies—ATM, best effort, and integrated service—offers a good basis for considering the targets of Differentiated Services. It is important that Differentiated Services can provide a consistent and efficient model on different levels of realization: customer service, network services, operation and management, and traffic handling. Before entering into these special areas, however, it is important to define the general meaning of Differentiated Services:

> Differentiated Services refer to a simple service structure that provides quality differentiation mainly by the means of packet marking.

This definition consists of three parts:

- Differentiated Services is a target model rather than a specification that contains detailed information about the required implementation. (This target is evaluated in the following sections.)
- From the service perspective, Differentiated Services provides a moderate level of quality differentiation without strict guarantees.
- The distinctive technical characteristic is that the quality of service is not attained by reserving capacity for each individual flow or connection, but by marking packets at the network boundaries.

2.5.1 Customer Service

The primary goal of customer service is for most (preferably all) customers to consider it fair. Traditionally, this issue was left out of the standardization of networking technologies.

It is too easy to just remark that service providers are allowed to adopt any existing or new pricing scheme or customer-care system. Unfortunately, the freedom is often superficial, because the underlying service model dictates to a large extent the structure of customer service.

The guaranteed service model, for instance, requires that you understand the essence of the service in a way that you can select the proper service level, request it, and assess whether the service you obtained satisfies the service contract. In addition, if the provision of guaranteed service is based on per-connection pricing, you have to be able to understand the bill. If you want to avoid all these tight requirements, you had better not to apply guaranteed service as your main service paradigm.

It could be better to take a different approach. Users naturally have expectations about the service. You should not to create too high expectations; those might be too expensive and difficult to realize. Instead, you should control customer expectations in a "soft" manner and keep customers so satisfied that they are willing to pay more than the basic flat rate. This requires a predictable pricing and understandable service structure.

2.5.2 Network Services

The fundamental attribute of network service is that it must be robust. This means that the service provider or network operator must control the function of the actual service. This is the main problem with the current best-effort service based on TCP: even though it works surprisingly well most of the time, it is vulnerable to attacks by malicious users.

ATM and IETF's Integrated Services model provides examples of inherently robust service models. The robustness is achieved through the use of advanced control mechanisms, including a lot of traffic parameters, resource allocation and reservation tools, and tight control over the traffic sent by the user. The drawback to this kind of system is that it is prone to errors because of the overall complexity and large number of parameters needed to manage it.

Differentiated Services should be able to combine inherent robustness achieved by traffic control and simple service structure without excessive parameters. Although this apparently is a big challenge, it is target that must remembered all the time when designing the Differentiated Services architecture.

2.5.3 Operation and Management

Operation and management of a network can be costly. Therefore, cost efficiency is a major concern. The rapid progress of information technology makes it possible to develop complicated systems that can work under very hard, real-time requirements. The productivity of human labor, on the other hand, has improved only slowly. Therefore, one of the main targets should be to minimize the human actions needed to manage Internet traffic.

One apparently labor-intensive task is to solve fault situations. As the possible reasons for one fault type increase, the difficulty of fixing the fault also increases. If one connection encounters excessive packet-loss ratio, for instance, there are numerous explanations:

- One of the traffic parameters of that connection is incorrectly set.
- One of the quality parameters of that connection is incorrectly set.
- The service class is inappropriate.
- The application sends more packets than the user supposes.
- The operator has not reserved enough capacity for an aggregate stream.
- The operator has installed some of the service classes incorrectly; then, there are several operators, and so on.

The possibilities are almost limitless.

The overall structure of Differentiated Services should be so clear that the management burden remains limited. Therefore, a consistent, robust set of automatic functions is highly recommendable.

2.5.4 Traffic Handling

The previous aspects emphasize the need for simplicity. If you have no tools to build the service differentiation, however, you end up either with the current best-effort model or with a simple connection-oriented model. What is needed, therefore, is one consistent set of traffic-handling mechanisms that allows different treatment of packets.

This versatile set of mechanisms has to be sufficient to support a variety of network services. Consistency makes it possible to build an effective system with inexpensive network management and customer care. Finally, overall efficiency means that you can provide services that are not too expensive, but that still give reasonable profit for service providers and network operators.

Summary

As a summary of this relatively long assessment of other networking technologies, consider the following list of questions for Differentiated Services:

1. How can you sell a service package to ordinary customers without any technical background?

2. What kind of billing system do you need to support your service model and to make it fair?
3. Do you understand all interactions between the building blocks of services, and do they allow efficient troubleshooting?
4. How efficient is the model when used in a large network with millions of users?
5. Is the service model robust enough to limit the effects of intentional misuse of network resources?
6. Does the service model provide a realistic evolution path from the current best-effort network?

If a service model can acceptably answer all these questions, it has a good chance of being successful.