NTIA Report 83-125

American National Standard X3.102
User Reference Manual

N. B. Seitz
D. S. Grubb

U.S. DEPARTMENT OF COMMERCE
Malcolm Baldrige, Secretary
David J. Markey, Assistant Secretary
for Communications and Information

October 1983

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

AMERICAN NATIONAL STANDARD X3.102 USER REFERENCE MANUAL

N. B. Seitz and D. S. Grubb*

American National Standard X3.102 defines a set of 21 standard parameters that provide a uniform means of specifying the performance of data communication systems and services as seen by users. This report is basically an explanation and elaboration of that standard. The report first outlines the benefits of using the standard from the viewpoint of the end user, the communication provider, and the communication manager. The report then summarizes the standard's overall approach and content in informal, non-technical terms. Finally, the report examines the meaning and importance of each standard parameter in a series of tutorial parameter descriptions. Typical parameter values are presented and their design implications are discussed.

1. INTRODUCTION

1.1 Background

On February 22, 1983, the Board of Standards Review of the American National Standards Institute (ANSI) voted to approve publication of American National Standard (ANS) X3.102, "Data Communication Systems and Services: User-Oriented Performance Parameters." The purpcse of the standard is stated in its opening paragraph:

> "...to establish a uniform means of specifying, assessing, and comparing the performance of data communication systems and services from the point of view of the data communication user."

The essence of the ANS X3.102 approach is summed up in the phrase "from the point of view of the data communication user." The ANS X3.102 parameters focus on user performance concerns rather than on engineering design considerations; and they describe end-to-end services rather than particular transmission or switching facilities. Because they are user-oriented, the parameters are also system independent--i.e., they may be applied to any digital communication system or service, irrespective of transmission medium, network topology, or control protocol. This property makes the parameters useful in performance comparison, user requirements specification, and top-down design.

*N. B. Seitz is with the Institute for Telecommunication Sciences, National Telecommunications and Information Administration, U.S. Department of Commerce, Boulder, CO 80303. D. S. Grubb is with the Institute for Computer Sciences and Technology at the National Bureau of Standards, Washington, DC 20234.

The X3.102 standard was developed by the Data Communication Performance Task Group, X3S35. That group had produced two earlier data communication performance standards, X3.44 (ANSI, 1974) and X3.79 (ANSI, 1980). These standards differ from X3.102 primarily in their focus on particular communication protocols.

American National Standard X3.102 evolved from an earlier standard, Interim Federal Standard 1033 (GSA, 1979). That standard was developed under the National Communications System's Federal Telecommunications Standards Program (Bodson, 1978) and has been applied in several trial Federal procurements of packet-switched services (e.g., see EPA, 1980). Standard X3.102 is similar in approach and content to Interim 1033 and is expected to eventually replace it as a Federal Standard.

Substantial progress has been made towards the development of standard measurement methods for the ANS X3.102 parameters. The National Telecommunications and Information Administration's Institute for Telecommunication Sciences (NTIA/ITS) has defined detailed measurement methods for the Interim 1033 parameters in proposed Federal Standard 1043 (Seitz et al., 1981a, 1981b). NTIA/ITS and the Institute for Computer Sciences and Technology at the National Bureau of Standards (NBS/ICST) have verified and demonstrated these methods via experimental measurements on the ARPANET (Wortendyke et al., 1982). Task Group X3S35 is developing a proposed ANSI measurement standard, X3S35/135, based on the 1043 model. The draft measurement standard is expected to be completed early in 1984.

1.2 Purpose and Scope of Report

The purpose of this report is to encourage and facilitate use of American National Standard X3.102 by providing an informal, nontechnical presentation of its objectives and content. The report is divided into three major sections. Section 2 outlines the benefits of using the standard from the viewpoint of the end user, the communication provider, and the communication manager. Section 3 summarizes the standard's overall approach and content. Section 4 provides a tutorial essay on the meaning and importance of each standard parameter. An annotated bibliography of technical reports and papers dealing with related performance assessment issues is provided in Section 5.

The explanatory, user-oriented nature of this report necessarily imposes limitations. The report takes the standard parameters as a given starting point, and provides little discussion of how or why they were selected.

Appendix B of ANS X3.102 identifies some of the alternatives considered in parameter selection. The report defines the standard parameters in an informal, narrative style. More rigorous definitions are provided in the standard itself. Finally, the report does not address the complex subject of performance measurement or related issues such as sampling strategy. These topics are being addressed in the measurement standard and in supporting reports. As an example, comprehensive procedures for sample size determination have been defined in a series of ITS reports by Crow (1974, 1978, 1979), and Crow and Miles (1976a, 1976b, 1977).

2. BENEFITS OF USING ANS X3.102

This section summarizes the benefits of using ANS X3.102 from three points of view: end user, communication provider, and communication manager. The section is divided into two parts. The first identifies a common problem with traditional data communication procurement methods--the mixing of user requirements analysis and system design. The second shows how the user-oriented performance parameters defined in ANS X3.102 can eliminate that problem and improve data communication procurement by providing a system-independent framework for functional specification.

2.1 A Common Procurement Problem

Communication system specification involves two basic steps:

1. Specifying what the system must do, in terms of a set of required functions and associated performance levels;

2. Specifying how the system will achieve these objectives, in terms of specific components, interconnections, and operations.

The first step is called a "user requirements analysis." Properly conducted, it is a careful examination of the user function the system must support. It determines the relationship between communication performance and user effectiveness, and thereby defines the objectives of system design. As an example, communication delay would have vastly different impacts on the user functions of inventory control and aircraft position reporting, and the user requirements in the two cases would differ accordingly. The output of the requirements analysis step is called a "functional specification" to emphasize the fact that it defines what is needed, but not how the need is to be met.

The second step in the system specification process is the detailed system design. In this step, the designer postulates various ways a system could be constructed to meet the user requirements and evaluates each relative to defined constraints. As an example, a specified delay requirement might be satisfied by a dedicated communication link with a relatively low transmission rate, by a packet switching network with a higher rate, or, conceivably, by a totally nontelecommunications solution such as express mail. The output of the system design step is called a "design specification" to emphasize its focus on how a previously stated need is to be met. In sum, the functional specification defines the required service; the design specification defines the system that provides that service.

A common problem with traditional data communication procurement methods is that they do not clearly distinguish between these two specification steps. As an example, existing Federal Property Management Regulations require that Federal agencies conduct a comprehensive "data communication study" before procuring data communication services (GSA, 1978). The required study output is "a written report detailing the data communications system which most economically and effectively satisfies the requirements of the proposed data processing system"--i.e., a detailed data communication system design. No intermediate functional specification is required or even suggested.

Such procurement methods tend to place the major responsibility for communication system design on the users, with unfavorable consequences for all participants. To most users, data communication is an "information transportation" service to be employed, like the mail, in moving information from one work place to another. Users have little interest in how information is physically moved; their concerns are with the ultimate performance and cost of the service. They feel, quite reasonably, that they should not have to understand how to design a system in order to use it.

Provider dissatisfaction with user design is just as strong. To a provider, the existence of a preconceived user design in a procurement specification means two things: (1) the provider's opportunity to choose the most efficient, economical method of meeting the user need has been usurped; and (2) the provider may have been precluded from seeking the user's business altogether if the products or services of a competitor have been specified. Most users have relatively little communication expertise, and their design specifications may be incomplete or misleading. Such specifications increase provider uncertainty and financial risk. Poorly prepared design specifications can also reduce provider revenue by delaying the installation and productive use of new services.

The communication manager operates as a broker between a group of users who require communication service and one or more providers who supply them. Communication managers are charged with two general responsibilities: (1) meet the communication needs of users at the lowest possible cost, and (2) ensure that communication procurements are conducted in accordance with organizational policy guidelines (e.g., maximum Federal reliance on the private sector).

Both communication management responsibilities are frustrated by a concentration of system design responsibility in user organizations. As documented in GAO (1977), user-developed designs tend to be costly, brute-force approaches--e.g., dedicated lines. Faced with the complex task of designing a communication system, users may give insufficient attention to the requirements analysis step, with the result that the procured system has little relationship to actual needs. Even when users are capable of efficient design, they are often not in a position to assess the economies of scale that might be realized by sharing transmission and switching resources with other user groups. Once a user group has committed itself to a particular system design, that design becomes the basis of discussion with the communication manager, with the result that little consideration may be given to other design alternatives or organizational policy guidelines.

To summarize, the absence of a clear distinction between user requirements specification and system design is detrimental to all participants in data communication procurement. Users are burdened with design responsibilities they are often unwilling or ill-equipped to fulfill. Providers may be denied the opportunity to compete in new data communication markets. Communication managers are unable to realize potential improvements in communication performance and economy. Substantial inefficiencies in the procurement and use of data communication services and equipment are the result.

2.2 ANS X3.102 as a Framework for Functional Specification

Figure 1 contrasts the traditional data communication procurement approach with a more effective functional approach. In the traditional approach, Figure 1a, the user requirements analysis and the system design are undifferentiated elements of an overall "data communication study." This

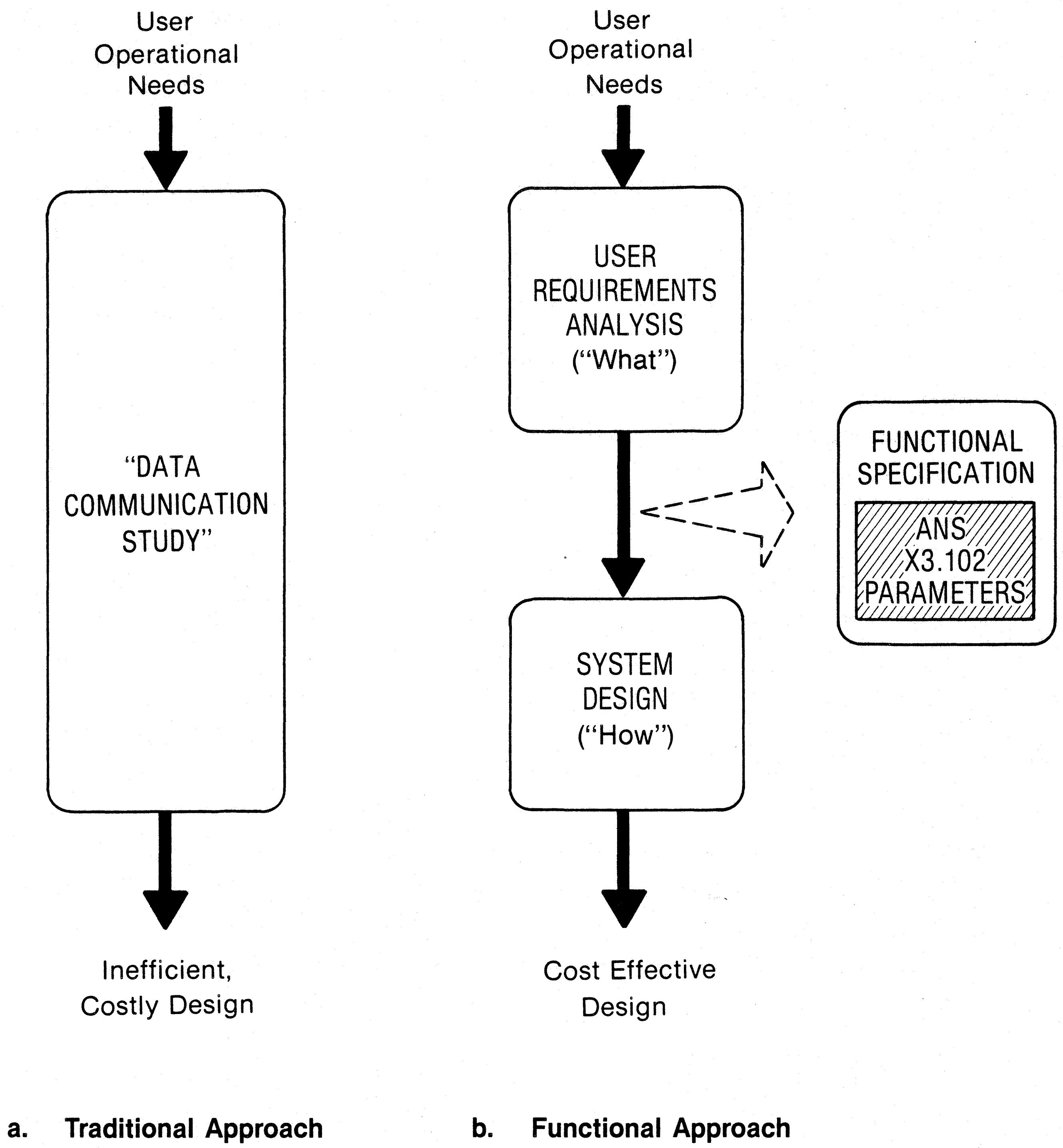


Figure 1. Traditional vs. functional procurement approaches.

approach encourages a mixing of "what" with "how," and often results in poor delineation of the user requirements, limited provider competition, and an inefficient, costly design. In the functional approach, Figure 1b, the user requirements analysis and the system design are clearly distinguished as separate, consecutive studies. This approach encourages more precise definition of the user requirements, facilitates provider competition, and often leads to a more cost effective design.

The key element that makes this separation possible is the functional specification: a precise statement of both user requirements and system design objectives expressed in system-independent terms. The functional specification must be comprehensive to ensure that any conforming system fully meets the user needs, but must not limit provider options by presupposing a particular design.

Functional specification of data communication services has been difficult in the past because of the lack of user-oriented, system-independent performance descriptors. American National Standard X3.102 provides such descriptors--in essence, a common language for relating the performance needs of end users with the capabilities of supplier systems. The standard will benefit users of data communication service in two ways:

1. By relieving them of a burdensome responsibility for communication system design. Users will be enabled and encouraged to regard data communication as an information transportation service--their natural inclination in the first place.

2. By allowing them to define their data communication needs more precisely. The X3.102 parameters will enable users to pinpoint the specific impacts of communication performance on their own operations, thereby minimizing procurement uncertainty and the risk of costly mistakes.

The standard may also assist users in evaluating the productivity improvement potential of proposed data communication enhancements.

American National Standard X3.102 will benefit data communication providers in three direct ways:[1]

[1] The discussion assumes provider design responsibility. Similar benefits accrue when communication managers are responsible for design, as long as a system-independent, functional specification is developed.

1. By enlarging the communication provider's participation in the design process. Under a functional procurement approach, the provider, rather than the user, determines the best way of meeting a stated communication need.

2. By offering a uniform method of specifying service performance. Providers will be able to develop a single basic performance specification applicable to many potential users.

3. By maximizing their opportunity to compete for user business. Expanded use of functional specifications in communication procurement will prevent the arbitrary exclusion of qualified bidders.[2]

Use of the standard will also benefit data communication providers in an indirect way, by improving their ability to assess existing or proposed new services from an end user perspective.

American National Standard X3.102 will assist communication managers in discharging both their user service and policy implementation responsibilities. To efficiently meet a user's data communication needs, the communication manager clearly must understand those needs--a difficult task if all discussion with the user is focused on a preconceived design. Similarly, implementing organizational procurement policies requires a certain authority over procurement decisions--and that authority must be exercised before the key design decisions are made. The X3.102 standard should be useful in three distinct phases of communication management: requirements specification, service acquisition, and service performance evaluation. The standard will be particularly useful to communication managers in matching end user requirements with offered systems and services.

All of these benefits are a result of the standard's "common denominator" property.

3. OVERVIEW OF THE STANDARD

This section summarizes the objectives and content of ANS X3.102. The section is divided into two parts. The first describes three performance description problems that influenced development of the standard. The second summarizes the overall approach used in defining the ANS X3.102 parameters and solving these problems. The 21 ANS X3.102 parameters are shown on Table 1. Refer to the standard and its supporting reports for further details.

[2] This is true of equipment providers as well as service providers, since subsystems can also be specified in functional terms.

SERVICE PERFORMANCE SPECIFICATION

Part A - Primary Parameters

1. Access Time.. ______ Seconds
2. Incorrect Access Probability.. .... ______ *
3. Access Denial Probability.. ______ *
4. Access Outage Probability.. ______ *

5. Bit Error Probability.. ______ *
6. Bit Misdelivery Probability.. ______ *
7. Bit Loss Probability .... ______ *
8. Extra Bit Probability .... ______ *

9. Block Transfer Time.. ______ Seconds
10. Block Error Probability.. ______ *
11. Block Misdelivery Probability.................................. ______ *
12. Block Loss Probability.. ______ *
13. Extra Block Probability... ______ *

14. User Information Bit Transfer Rate......................... ______ Bits/ Second

15. Disengagement Time .... ______ Seconds
16. Disengagement Denial Probability.............................. ______ *

17. Transfer Denial Probability.................................... ______ *

Part B - Ancillary Parameters

18. User Fraction of Access Time............................... ______ *
19. User Fraction of Block Transfer Time...................... ______ *
20. User Fraction of Sample Input/Output Time............. ______ *
21. User Fraction of Disengagement Time..................... ______ *

*Note: The probabilities and user performance time fractions are dimensionless numbers between zero and one.

Table 1. Summary of ANS X3.102 Parameters

3.1 Performance Description Problems

Development of ANS X3.102 required the solution of three fundamental performance description problems: system dependence, definitional precision, and user performance delay. Each problem is discussed and illustrated below.

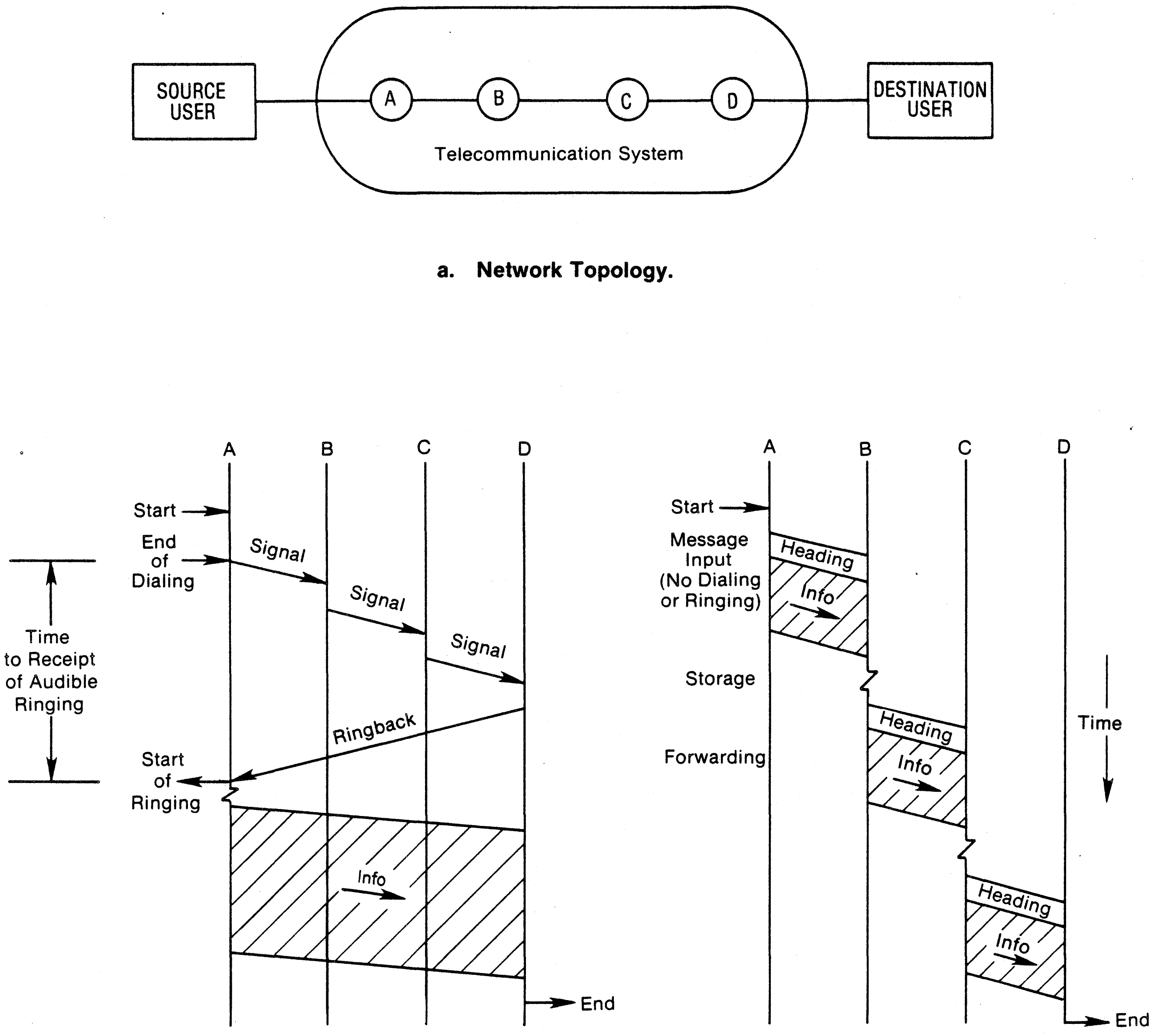
3.1.1 System Dependence

System dependence is that property of a performance parameter's definition that specializes (and restricts) its application to systems with particular design features. System-dependent parameters can be very useful to providers in design optimization, but they frustrate performance comparison and are thus undesirable in a user-oriented standard. A major challenge in developing ANS X3.102 was to avoid basing the parameter definitions on system design assumptions--e.g., a particular network topology or protocol architecture.

A good insight into the problem of system dependence can be obtained by considering the differences between traditional circuit-switched and message-switched networks. Figure 2a shows a simple network topology that might represent either. In a traditional circuit-switched network, an end-to-end path or circuit is established between users prior to the start of user information transfer (Figure 2b). The individual links that comprise the end-to-end circuit are allocated to that particular user pair for the duration of the connection, independent of usage; all links are used simultaneously during transfer. A familiar example is the public voice telephone network.

In a traditional message-switched network, no end-to-end circuit is established prior to the start of user information transfer (Figure 2c). Instead, the user message is forwarded through the network link by link, and the entire message is stored for some time at each intermediate node. Individual links are allocated to a particular user only during actual transmission of that user's message. At all other times, the link may support other users. The General Services Administration's Automatic Record System (ARS) is one example of such a network.

One parameter that is commonly used in expressing the performance of circuit-switched networks is the Time to Receipt of Audible Ringing--the elapsed time from the end of dialing to the start of ringing (Figure 2b). No counterpart to this parameter is possible in message-switched systems. In such systems, the function of switching is performed by interpretation of the



a. **Network Topology.**

b. **Circuit-Switched Communication**

c. **Message-Switched Communication**

Figure 2. System dependence example.

message heading at each node, rather than by pre-transmission signaling. The interface events associated with dialing and ringing are just not applicable.

Clearly, a user wishing to compare the performance of circuit-switched and message-switched services cannot do so in terms of Time to Receipt of Audible Ringing. Other, system-independent descriptors of performance are required. The absence of such system-independent performance descriptors has been a major difficulty with performance comparison in the past.

3.1.2 Definitional Precision

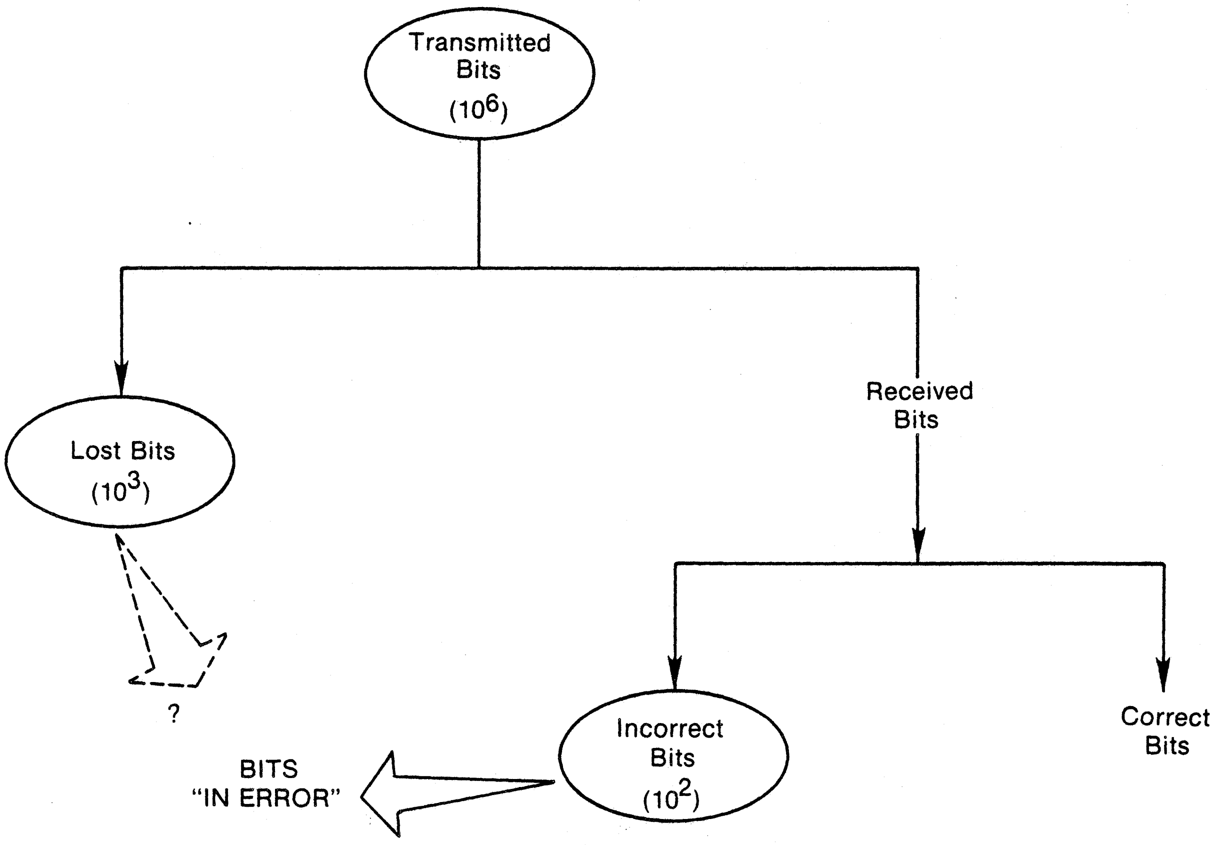
The second performance description problem encountered in developing ANS X3.102 was the problem of definitional precision. A review of candidate parameter definitions revealed that many left major decisions open to user interpretation. The result, of course, is a potential for differences of opinion about the parameter values. The intended use of ANS X3.102 in performance comparison required that it eliminate most, if not all, such ambiguities.

As an example of the precision problem, consider the familiar accuracy parameter Bit Error Probability (Figure 3). This parameter is often described as the number of bits in error divided by the number of bits transmitted, with no mention of whether (or how) bits lost in transmission should be counted. Two obvious choices, both consistent with the narrative definition, would be (1) to count lost bits and received incorrect bits in calculating Bit Error Probability, and (2) to considered only received incorrect bits in calculating Bit Error Probability.

This ambiguity can have a substantial effect on measured parameter values. Assume that of a million (10^6) bits transmitted during a test, a thousand (10^3) are lost and a hundred (10^2) are inverted in transmission. The measured Bit Error Probability values under the two choices are 1.1 x 10^{-3} and 10^{-4}--an order of magnitude error in interpreting the meaning of a narrative parameter definition! Data loss is actually more common than data error in many services (e.g., see AT&T, 1971; Wortendyke et al., 1982).

3.1.3 User Delay

The third performance description problem encountered in developing ANS X3.102 was the problem of user delay. In most cases, the communication process involves a sequence of interactions between the users and the system. The observed performance is therefore influenced by user performance as well



$$\left\{\begin{matrix}\text{Bit Error}\\ \text{Probability}\end{matrix}\right\} = \left\{\begin{matrix}\text{Bits “In Error”}\\ \text{per Bits Transmitted}\end{matrix}\right\}$$

Choice 1 — Lost Bits Counted:

$$\left\{\begin{matrix}\text{Bit Error}\\ \text{Probability}\end{matrix}\right\} = \frac{10^2 + 10^3}{10^6} = 1.1 \text{ X } 10^{-3}$$

(Disparity in Values)

Choice 2 — Lost Bits Not Counted:

$$\left\{\begin{matrix}\text{Bit Error}\\ \text{Probability}\end{matrix}\right\} = \frac{10^2}{10^6} = 10^{-4}$$

Figure 3. Parameter definition example.

as system performance. There is an obvious problem in employing user-dependent parameters in specifying the required performance of the system: the service provider has no control over user performance and thus cannot ensure that user-dependent parameter values will be met.

Telephone circuit establishment provides a simple example of the user delay problem (Figure 4). In placing a call, a typical telephone user is concerned with how soon conversation can begin, i.e., the total delay between his off-hook action and the called party's answer. The performance parameter Access Time describes this delay; but its value depends on the users' speed in dialing and answering as well as the system's speed in signaling and switching.

Common carriers have traditionally avoided the user delay problem by specifying parameters that describe unilateral system performance. Examples are Dial Tone Delay (the time from off-hook to dial tone) and Time to Receipt of Audible Ringing (the time from the end of dialing until the calling party hears ringing). Unfortunately, such parameters have three major disadvantages from the user point of view. First, they are system dependent (as noted above). Second, they often do not answer the questions of major concern to users. As an example, a user wishing access to a remote computer data base is concerned with the total access delay, including the response time of the remote computer, rather than with the particular delay components under network provider control.[3] Finally, they do not reflect differences in the "functional burden" placed on the users by otherwise equivalent services. For example, if one service requires seven-digit dialing, while another permits abbreviated dialing with only three digits, the effect on the user delay is considerable.

In general, then, it appears that user-oriented performance specifications must consider both total (observed) delays and their user and system components. A method of distinguishing user and system delays is described below as one element of the ANS X3.102 approach.

[3] Providers must also consider total delay in network design. As an example, dial receivers in a circuit switch are shared between lines, and the number required depends on the user's signalling rate.

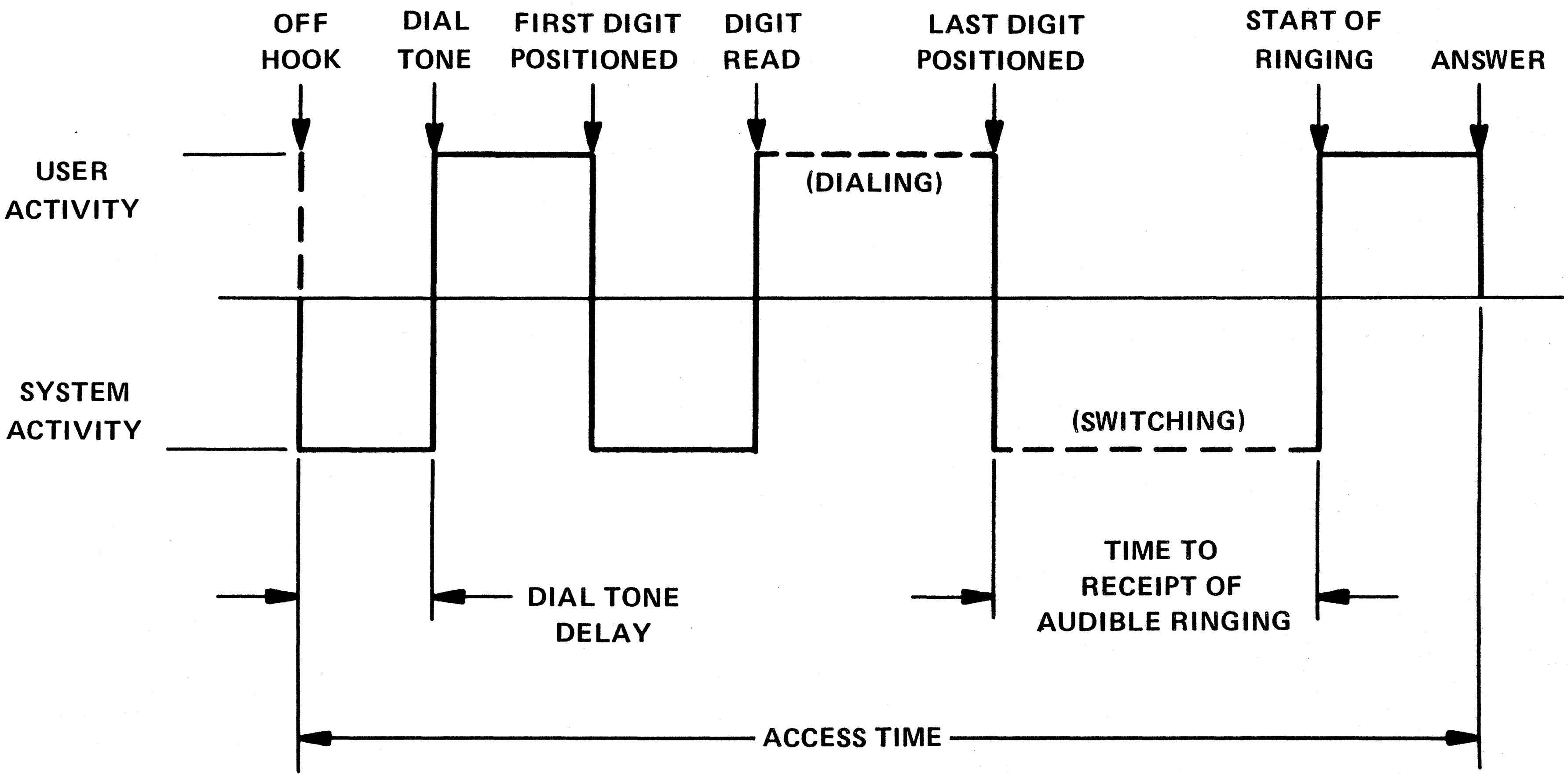


Figure 4. User and system components of telephone network access time.

3.2 ANS X3.102 Approach

Figure 5 summarizes the overall approach used by ANSI Task Group X3S35 in developing the ANS X3.102 performance parameters. The parameter development process consisted of four major steps:

1. Model Development. Existing and proposed data communication services were reviewed and certain universal performance characteristics shared by all were identified. These characteristics were consolidated in a simple, user-oriented model which provided a system-independent basis for the performance parameter definitions.

2. Function Definition. Three primary communication functions were selected and defined on the basis of the user-oriented model. These functions (access, user information transfer, and disengagement) provided a specific focus for the performance description effort.

3. Outcome Definition. Each primary function was analyzed to determine the possible outcomes an individual trial performance might encounter. Possible outcomes were grouped into three general categories: successful performance, incorrect performance, and nonperformance. These categories correspond to the three general performance concerns most frequently expressed by end users: speed, accuracy, and reliability.

4. Parameter Selection. Each primary function was considered relative to each performance outcome in matrix fashion. One or more specific parameters were selected to represent performance relative to each function/outcome pair. Parameters were selected on the basis of expressed user interest. These parameters consisted of probabilities, waiting times, and time rates. The matrix approach ensured that no significant aspect of data communication performance would be overlooked in the parameter selection process.

The following subsections describe the results of these steps in more detail.

3.2.1 Model Development

In order to describe data communication performance as seen by the end user, it is necessary to develop a user-oriented view of the data communication process. What is the nature of the interface between an end user and a data communication system? How is information transferred across such interfaces? How can the process of data communication be described in a way that is meaningful and familiar to the end user, but not restricted to a particular type of interface or a particular interaction sequence? These are questions ANS X3.102 answers with the aid of a user-oriented model of the data communication process.

The model described in ANS X3.102 defines the end user of a data communication system or service as one of the following types of entities:

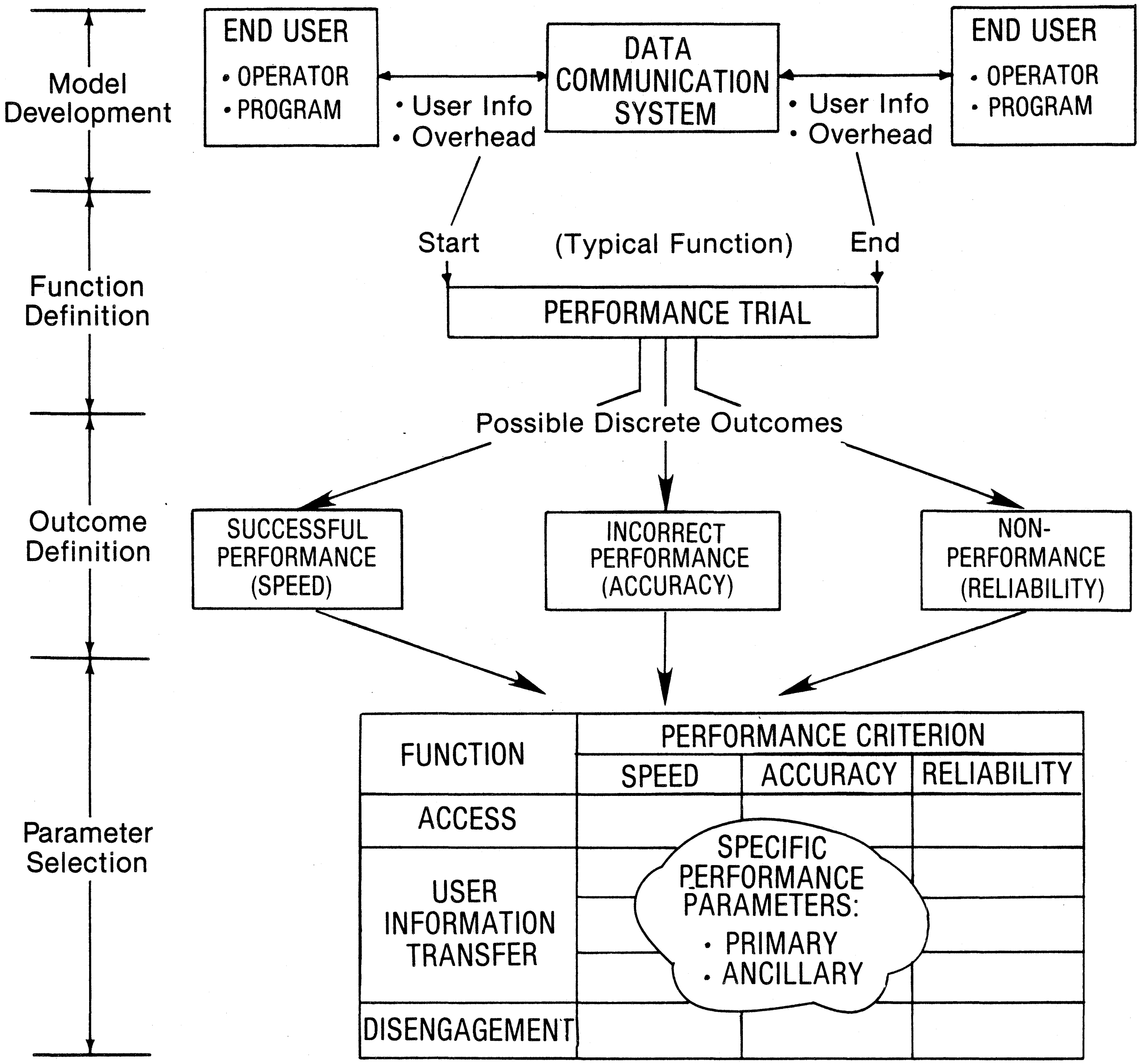


Figure 5. Overview of ANS X3.102

1. The human operator of a data terminal.

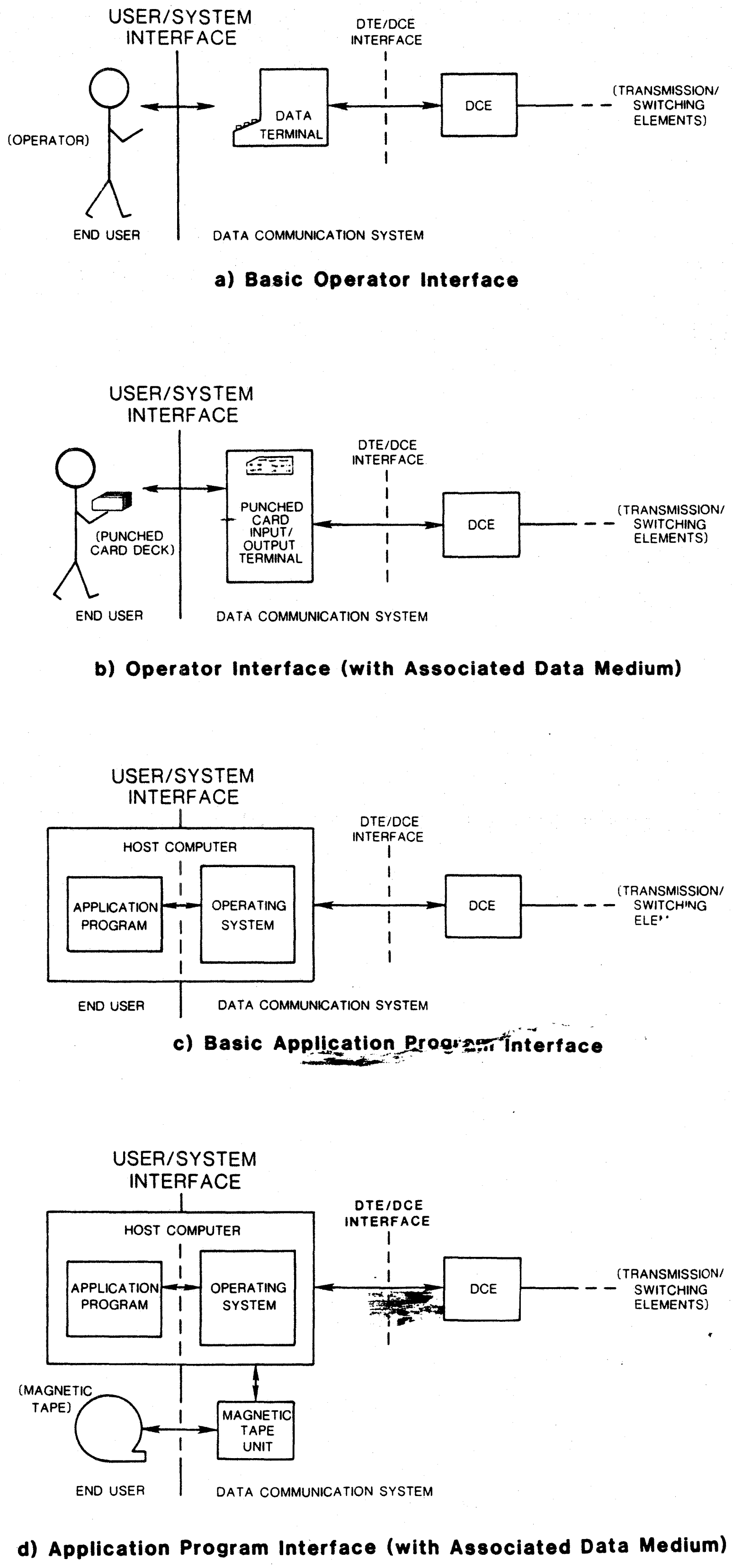
2. A computer application program that processes communicated information.

An example of the first type of end user is a person operating an automated banking terminal. The terminal converts information from a form that is meaningful to the user into transmittable form (e.g., encoded binary bits) and vice versa (e.g., displaying received characters on a CRT). In all cases where the end user is a human operating a terminal, the end-to-end data communication system is defined to include the data terminal and all elements of the information transfer channel on its line side. The user/system interface then corresponds to the physical interface between the operator and the terminal.

Examples of the second type of end user are a FORTRAN program which calculates payroll information based on employee records stored in a remote data base, and the remote data base management program that provides the employee records. Such programs typically interact with a data communications "front end," which may be implemented either in the same (host) computer or in a separate hardware device. The host computer's operating system is viewed as part of the end-to-end data communication system in most cases, since user programs typically can only access network resources via operating system support. The user/system interface then corresponds to the functional interface between the application program and the operating system.

Either a human operator or an application program may use data recording media in transferring information to or receiving information from a system. Typical media used by terminal operators are punched cards, magnetic stripe cards, punched paper tape, and typewritten or printed pages. Typical media used by application programs are magnetic type and magnetic disks. In all cases where such data media are employed, they are associated with the user rather than with the system. The user/system interface is defined to include the medium/terminal interface in such cases. four basic types of user/system interfaces are thus defined: (1) basic operator interface, (2) operator interface (with associated data medium), (3) basic application program interface, and (4) application program interface (with associated data medium). These are illustrated in Figure 6.

The definitions just described place the end user interfaces well outside the traditional DTE/DCE (Data Terminal Equipment/Data Circuit-terminating Equipment) or "computer/communications" boundaries. This viewpoint is



a) Basic Operator Interface

USER/SYSTEM INTERFACE
DTE/DCE INTERFACE
PUNCHED CARD INPUT/ OUTPUT TERMINAL
DCE
(TRANSMISSION/ SWITCHING ELEMENTS)
(PUNCHED CARD DECK)
END USER
DATA COMMUNICATION SYSTEM

b) Operator Interface (with Associated Data Medium)

c) Basic Application Program Interface

d) Application Program Interface (with Associated Data Medium)

Figure 6. End user/data communication system interfaces.

essential in a user-oriented standard, since modern terminals and high-level protocols perform communication functions (such as error control, flow control, and virtual circuit establishment) that have a profound effect on end-to-end performance. One modern data communication network whose end user interfaces are defined in this way is IBM's Systems Network Architecture (McFadyen, 1976).

Information can be transmitted across the user/system interface in a variety of ways. Typical interactions at the operator/terminal interface are manual keystrokes on a terminal keyboard and the printing or displaying of received characters. Typical interactions at the application program/operating system interface are operating system calls and the exchange of application program data. Typical interactions at a medium/terminal interface are the reading and writing of information on punched cards, magnetic disks, and magnetic tapes. When the user/system interface is within a computer, information transfers can occur either by physical movement of the information (e.g., between buffers) or by transfer of right of access to the information (i.e., buffer "ownership").

All of the user/system interactions just described are examples of what the standard calls "interface events." An interface event is any discrete transfer of user or overhead information across a user/system interface. User information includes all information intended to cross both user/system interfaces. All other information (e.g., ENQ, ACK, and SYN characters, off-hook and on-hook signals) is overhead information.

In any description of performance, certain key interface events are identified as events to be counted, timed, or compared in calculating performance parameter values. As noted earlier, most existing standards and specifications identify such key events by reference to particular system-specific signals. The ANS X3.102 model departs from this approach by defining the performance parameters in terms of more general, system-independent reference events. Each ANS X3.102 reference event is a "generic event" which subsumes many system-specific interface events having a common performance significance; and each is defined in such a way that it can always be identified, if it occurs, in any particular data communication session. The reference events collectively specify all information needed to describe performance in a comprehensive, user-oriented way.

An example will help to clarify the relationship between system-specific interface events and the associated reference events. A user's action in lifting a telephone handset off-hook transfers one bit of overhead information (the new hookswitch position) from the user to the system. This system-specific interface event corresponds to the X3.102 reference event "access request." The same reference event might be generated by a completely different interface event in another system: an example is issuance of a "connect" system call in the ARPA network.

A second example of a system-independent reference event defined in X3.102 is "start of block transfer." Such an event must obviously be identified to define performance parameters such as Block Transfer Time. In order to define that event, one must clearly identify (1) what is meant by a "block," and (2) when the transfer of a block between users should be regarded as "started." Standard X3.102 defines a <u>user information block</u> as a contiguous group of user information bits delimited at a source user/system interface for transfer to a destination user as a unit. The transfer of a block is said to have started when two conditions have been met:

1. The user information contained in the block is physically present within the system facility.

2. The system has been authorized to deliver the block to the destination user.

The latter criterion (authorization) is the most natural way to establish the block boundaries as well; i.e., authorizing delivery of a given unit of information identifies that unit as an ANS X3.102 block. Authorization may either be an explicit user action (e.g., typing Carriage Return at a buffered CRT terminal) or an implicit part of entering the user information itself (e.g., typing a single character at an asynchronous terminal).

Given the above definitions, the nature of the information unit called a "block" and the physical events associated with block transfer will differ from one system to another. Nevertheless, in every system <u>some</u> specific information unit can be identified as an ANS X3.102 block, and the start of transfer of that unit can be determined, using the above criteria. Hence, the reference event start of block transfer can always be identified. A similar approach is used in defining the other user information transfer reference events. A complete list of the ANS X3.102 reference events is provided in Appendix C of the standard.

In describing the objectives of the ANS X3.102 data communication process model, we raised two questions that have not been explicitly answered as yet:

1. How can the process of telecommunication be described in a way that is meaningful and familiar to the end user, and yet not restricted to a particular type of interface or interaction sequence?

2. How should the performance parameter definitions be related to such a description?

We are now in a position to answer these questions. Standard X3.102 represents the data communication process as a chronological sequence of reference events, each specifying the time of occurrence and performance significance of an associated user or overhead information transfer. The performance parameter definitions are based on the reference events. This makes the parameters system-independent, and therefore universally applicable.

Although the ANS X3.102 performance model was developed primarily to represent end-to-end services, it is not restricted to such applications--any digital telecommunication process can be represented as a chronological sequence of reference events. In order to apply the model (and therefore the standard) to a digital subsystem, it is only necessary to (1) define the interfaces of interest, (2) identify the specific events occurring at these interfaces, and (3) associate each specific interface event with a corresponding model reference event. The ANS X3.102 parameters can then be applied directly to the subsystem, since the parameter definitions are all based on the reference events.

Figure 7 illustrates a possible subsystem application. In this application, the subsystem interface is placed at the DTE/DCE physical interface; and the operator and terminal are regarded as an "aggregate user" of the information transfer channel. Such applications can be useful in allocating end-to-end performance objectives to system components and services and, conversely, in determining the impact of subsystem choices on end-to-end performance.

3.2.2 Function Definition

Performance has little meaning as an isolated concept. To be useful, a description of performance must be clearly related to some particular function. The second step in developing the ANS X3.102 parameters was to define a set of specific data communication functions to be used as the focus of the performance description effort.

The three primary data communication functions addressed in the standard are defined in terms of general reference events as follows.

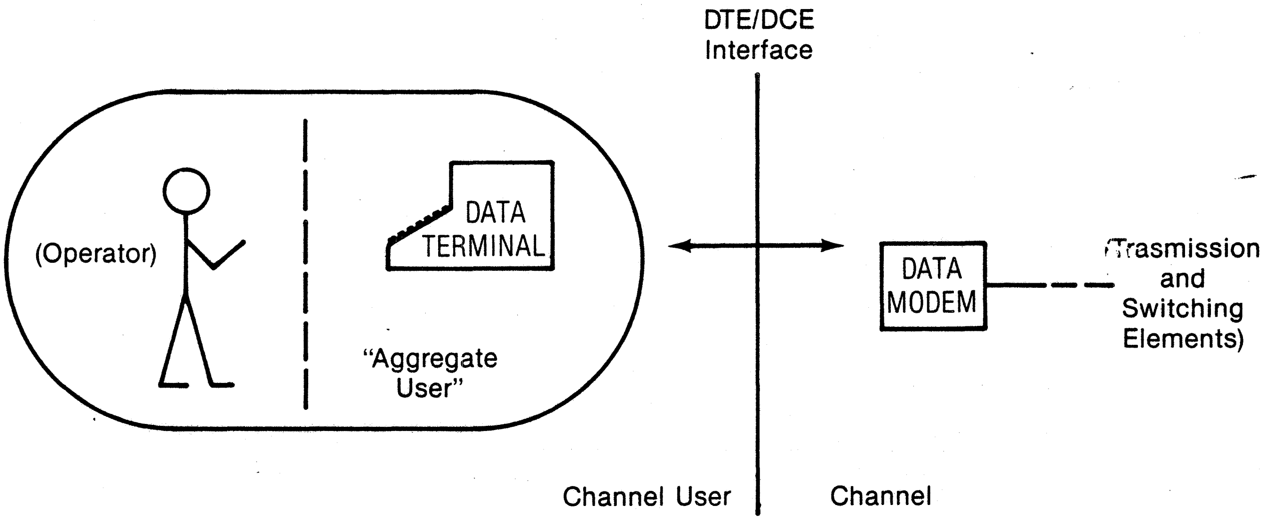


Figure 7. Typical subsystem application.

> The access function begins upon issuance of an "access request" signal or its implied equivalent at the interface between a user and the data communication system. It ends when the first bit of source user information is input to the system (after connection establishment in connection-oriented services). It includes all activities traditionally associated with physical circuit establishment (e.g., dialing, switching, and ringing) as well as any activities performed at higher protocol levels (e.g., X.25 virtual circuit establishment).

Making the end of access coincident with the start of input of user information to the system reflects the user view that no data communication service has actually been provided until user information begins to flow. Note, however, that the end of access is defined to occur when the first user information unit is input to the system, even though actual transmission of that unit may not begin until a subsequent "transmission authorization" is issued by the source user. Data input and transmission authorization are coincident in many systems, but may be separated by a substantial time interval in systems that provide input buffering (e.g., the CCITT X.28 editing buffer).

> The user information transfer function begins when the access function ends. The user information transfer function ends when the last "disengagement request" in a particular data communication session is issued. It includes all formatting, transmission, storage, error control, and media conversion activities performed between start of transfer and completion of delivery, including any needed retransmissions within the system.

Two more specific user information transfer functions are defined to provide a more comprehensive description of performance: the bit transfer function and the block transfer function. Each function begins when the specified user information unit (bit or block) has been input to the system and the system has been authorized to deliver it. Each function ends when the specified user information unit is actually delivered to the destination user (with appropriate notification of that user where required). The ANS X3.102 meaning of the term "block" has been described earlier (Section 3.2.1).

> There is a disengagement function associated with each participant in a data communication session. Each disengagement function begins on issuance of a "disengagement request". The disengagement function ends, for each user, when (1) disengagement has been requested for that user; and (2) that user is able to initiate a new access attempt. Disengagement includes both physical circuit disconnection (where required) and higher-level protocol termination activities such as X.25 virtual circuit clearing.

Depending on system characteristics, the disengagement request for an end user may be issued by that end user, by the other user participating in the data communication session, or by the system. Most data communication systems notify the end user that a new access can be initiated by issuing an explicit "disengagement confirmation" signal. In cases where no such notification is provided in the interface protocol, the user may issue a new access request to confirm successful disengagement.

The terms "access request," disengagement request," and "disengagement confirmation" are general descriptors of purpose rather than particular interface signals. An access request is any interface signal issued for the purpose of initiating a data communication session. Examples are the "off hook" signal in the public telephone network, the "open request" in the ARPANET, and "open destination" (OPNDST) VTAM macro in IBM's Systems Network Architecture (SNA).

A disengagement request is any interface signal issued for the purpose of terminating an entity's participation in a data communication session. The disengagement request signals corresponding to the three access request signals just cited are the "on hook" signal in the public telephone network, the "close request" signal in the ARPANET, and the "close destination" (CLSDST) macro in SNA.

A disengagement confirmation signal is issued for the purpose of confirming termination of a data communication session. In the case of the ARPANET and SNA, disengagement confirmation is indicated by an explicit interface signal (a "close complete" response). In the public switched telephone network, disengagement confirmation is an implied event that must be verified by a subsequent user access request (i.e., going "off-hook" and checking for the dial tone).

The bit transfer and block transfer functions defined above each serve a distinct purpose in the description of user information transfer performance. The bit transfer function fulfills the need for a common basis for comparing services having different characteristic block lengths. The block transfer function describes performance in terms of the information unit that is normally most relevant to the user.

An important characteristic of the primary communication functions defined in ANS X3.102 is that they are user dependent; i.e., their successful completion depends, in general, on events that must be produced by a user. As noted earlier, there is a problem in using parameters based on such functions to describe required system performance: the supplier has no control over user performance, and hence cannot ensure that user dependent parameter values will be met. Standard X3.102 overcomes this problem by explicitly describing the influence of user delay on the primary parameter values by means of separate "ancillary" parameters. The definition and use of these parameters is described in Section 3.2.5.

3.2.3 Outcome Definition

In defining performance parameters for a function, there is a clear need to identify the possible outcomes (or end results) that might occur on any given performance of the function. The third step in developing the ANS X3.102 parameters was to define such a set of possible outcomes for each of the primary data communication functions. These possible outcomes can be grouped into three general categories:

1. Successful Performance. The function is completed within a specified maximum performance time, and the result or output is exactly what was intended. A familiar example is successful connection to the correct called party in a voice telephone call.

2. Incorrect Performance. The function is completed within the specified maximum performance time, but the result or output is somehow different from what was intended. A familiar example is connection to a "wrong number" (as a result of a system switching error) in a voice telephone call.

3. Nonperformance. The function is not completed within a specified maximum performance time. A familiar example is the blocking of a voice telephone call attempt by the system (as indicated by a "circuit busy" signal).

These three outcome categories are significant because they correspond very closely with the three basic performance concerns most frequently expressed by data communication users. Successful performance is associated with the user's concern with speed (delay or rate). Incorrect performance is associated with a user concern with accuracy. Nonperformance is associated with a user concern with reliability. Standard X3.102 uses these three general performance criteria as an overall framework for organizing the primary parameters.

The X3.102 standard divides the incorrect performance and nonperformance outcome categories into more detailed outcomes to enable the definition of more specific performance measures. In general, system outputs produced during the performance of a function can be incorrect in three ways: they may be incorrect in content; they may occur at an incorrect location; or they may include duplicate or other unrequested (extra) information. Failure to produce the expected output of a function can be the result of either system or user nonperformance. Thus, ANS X3.102 distinguishes six possible outcomes of an individual trial performance of a typical primary function:

1. Successful Performance. The expected output occurs and is correct in both location and content.

2. Content Error. The expected output occurs at the correct location, but is incorrect in content.

3. Location Error. The expected output occurs at an incorrect location.

4. Extra Event. An extra (unwanted) output occurs in addition to that expected.

5. System Nonperformance. The expected output does not occur within the maximum performance time. This may occur either as a result of the system issuing a blocking (busy) signal or due to excessive delay by the system.

6. User Nonperformance. The expected output does not occur within the maximum performance time. This may occur either as a result of the user issuing a blocking (busy) signal or due to excessive delay by the user.

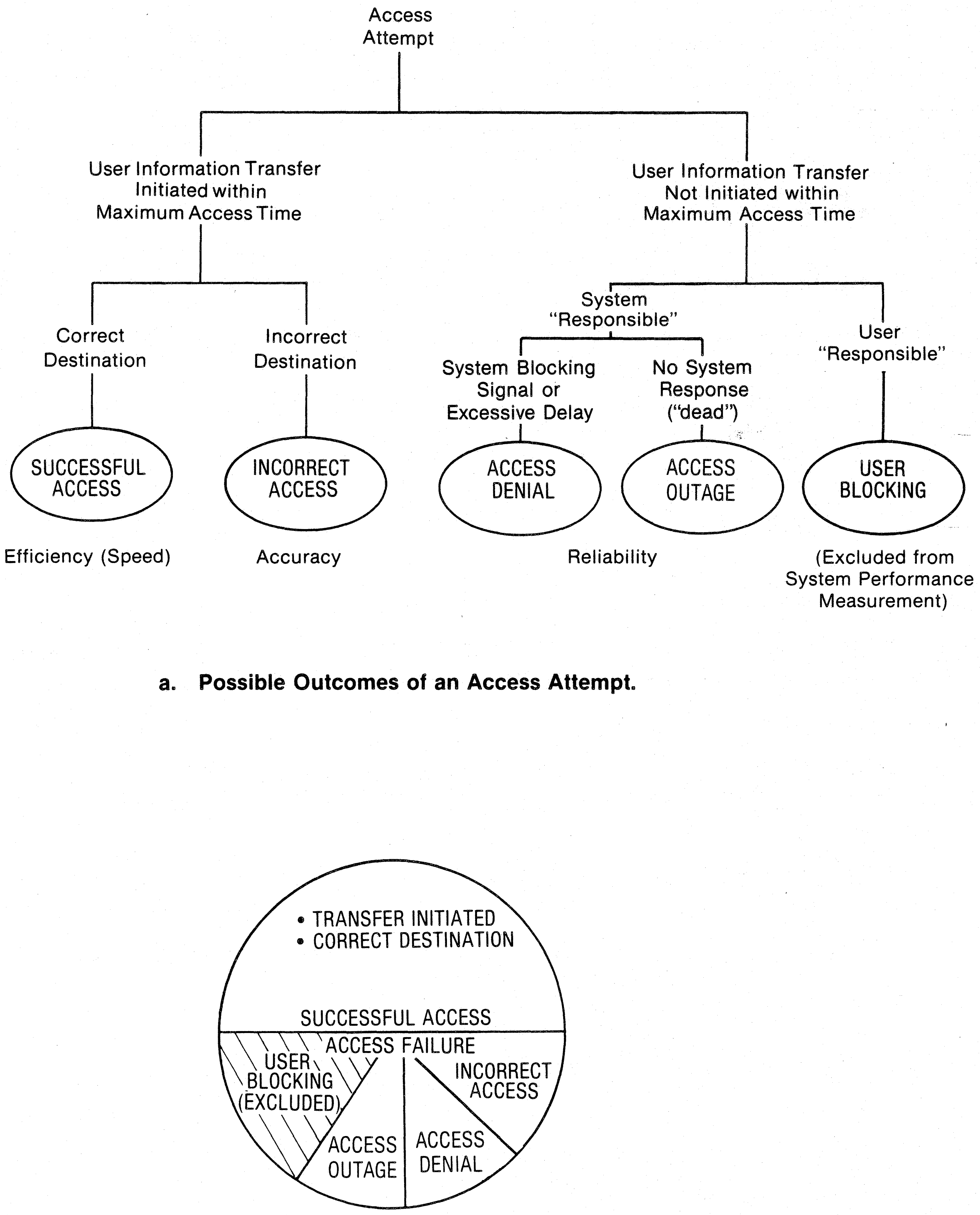
Outcome "sample spaces" for the primary functions were defined by selecting pertinent outcomes from the above list and specializing their meaning for each function. Figure 8 shows how this was done in the case of the access function. The standard defines five possible access outcomes: Successful Access, Incorrect Access, Access Denial, Access Outage, and User Blocking.

Successful Access is the case where user information transfer is initiated as intended within the specified maximum access time.

Incorrect Access is the case where transfer is initiated within the maximum time, but the information transfer is to a user other than the one intended by the originator.

Access Denial is the case where an access attempt fails as a result of either the issuance of a blocking signal or excessive delay by the system.

Access Outage corresponds to the case of a "dead" system. It is defined to occur when the system fails to issue any active interface signal during the access attempt.



a. **Possible Outcomes of an Access Attempt.**

b. **Sample Space Representation.**

Figure 8. Access outcome definition.

> User Blocking is the case where an access attempt fails as a result of either the issuance of a blocking signal or excessive delay by a user. User blocking outcomes are excluded in defining the access parameters.

Note that the Content Error and Extra Event outcomes are not applicable in the case of the access function, since such errors will result in either Incorrect Access or Access Denial. Note also that two separate system nonperformance outcomes were distinguished in defining the access parameters: Access Denial and Access Outage. The reason for distinguishing these outcomes was that the appropriate user actions in the two cases differ. In the case of Access Denial, a user can often get service by simply reattempting access. In the case of Access Outage, maintenance action is often required.

The public switched telephone network provides familiar examples of these access failure outcomes. Incorrect Access is the case of a "wrong number" caused by a system switching error. Access Denial is indicated by the high repetition "circuit busy" tone. User Blocking is indicated by the low repetition "user busy" tone. Access Outage is indicated by the absence of dial tone for an extended period.

Figure 9 shows the possible outcomes the standard defines for the block transfer function.

> Successful Block Transfer is the case where a transmitted block is delivered to the intended destination within a specified maximum block transfer time, and the delivered block is completely correct in content.
>
> Extra Block is the case where the system delivers to a destination a block that was not output by the source. Extra blocks may be duplicates of blocks previously sent by the source, or may be blocks generated by the system. Misdelivered Blocks will be counted as Extra Blocks unless a separate misdelivery detection procedure is invoked.
>
> Incorrect Block is the case where a transmitted block is delivered to the intended destination, but the delivered block contains one or more bit errors, additions or deletions.
>
> Misdelivered Block is the case where the transmitted block is delivered to a destination other than the one intended by the source. The block may be either correct or incorrect in content.
>
> Lost Block is the case where a transmitted block is not delivered to the intended destination within the maximum block transfer time, and the failure is attributable to the system.
>
> Refused Block is the case where a transmitted block is not delivered to the intended destination within the maximum block transfer time, and the

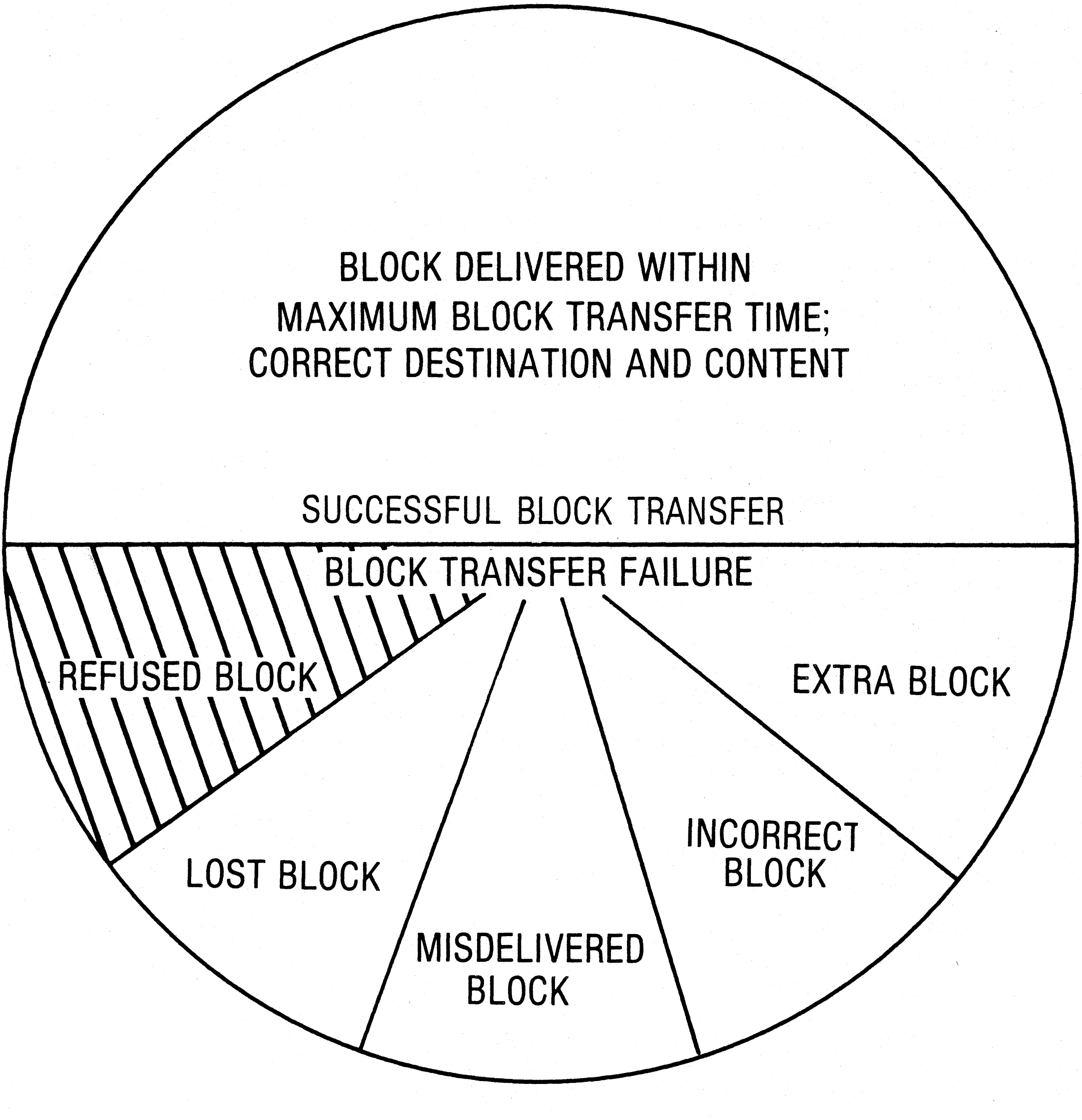


Figure 9. Block transfer outcomes.

failure is attributable to a user.[4] Refused Blocks are excluded in defining the user information transfer parameters.

The standard defines the bit transfer outcomes in a similar manner.

Figure 10 shows the possible outcomes ANS X3.102 defines for the disengagement function.

> Successful Disengagement is defined to occur when the disengaging user is freed to initiate a new data communication session within the specified maximum disengagement time. As noted earlier, this outcome is often indicated by an explicit disengagement confirmation signal issue by the system. If the system does not provide a disengagement confirmation signal, the user may confirm Successful Disengagement by making a subsequent access request.
>
> Disengagement Denial is the case where the disengaging user is not freed to initiate a new session within the specified maximum disengagement time, and the failure is attributable to the system.
>
> User Disengagement Blocking is the case where the disengaging user is not freed to initiate a new session within the specified maximum disengagement time, and the failure is attributable to a user. User Disengagement Blocking outcomes are excluded in defining the disengagement parameters.

Note that the content error, location error, and extra event outcomes are not applicable to the disengagement function.

Figure 11 summarizes the possible outcomes ANS X3.102 defines for each of the primary functions. Specific examples of each outcome are presented in Section 4.

3.2.4 Parameter Selection

The final step in developing the parameters was to select and define a minimum set of parameters to describe performance relative to each function and outcome. Figure 12 shows how this was done in the case of the access function. Access performance was described in terms of four specific measures:

> Access Time is the average value of elapse time between the start of an access attempt and Successful Access. Elapsed time values are calculated only on access attempts that result in Successful Access.
>
> Incorrect Access Probability is the ratio of total access attempts that result in Incorrect Access to total access attempts in an access performance sample (excluding User Blocking outcomes).

[4] A destination user might "refuse" a block, for example, by exercising flow control.

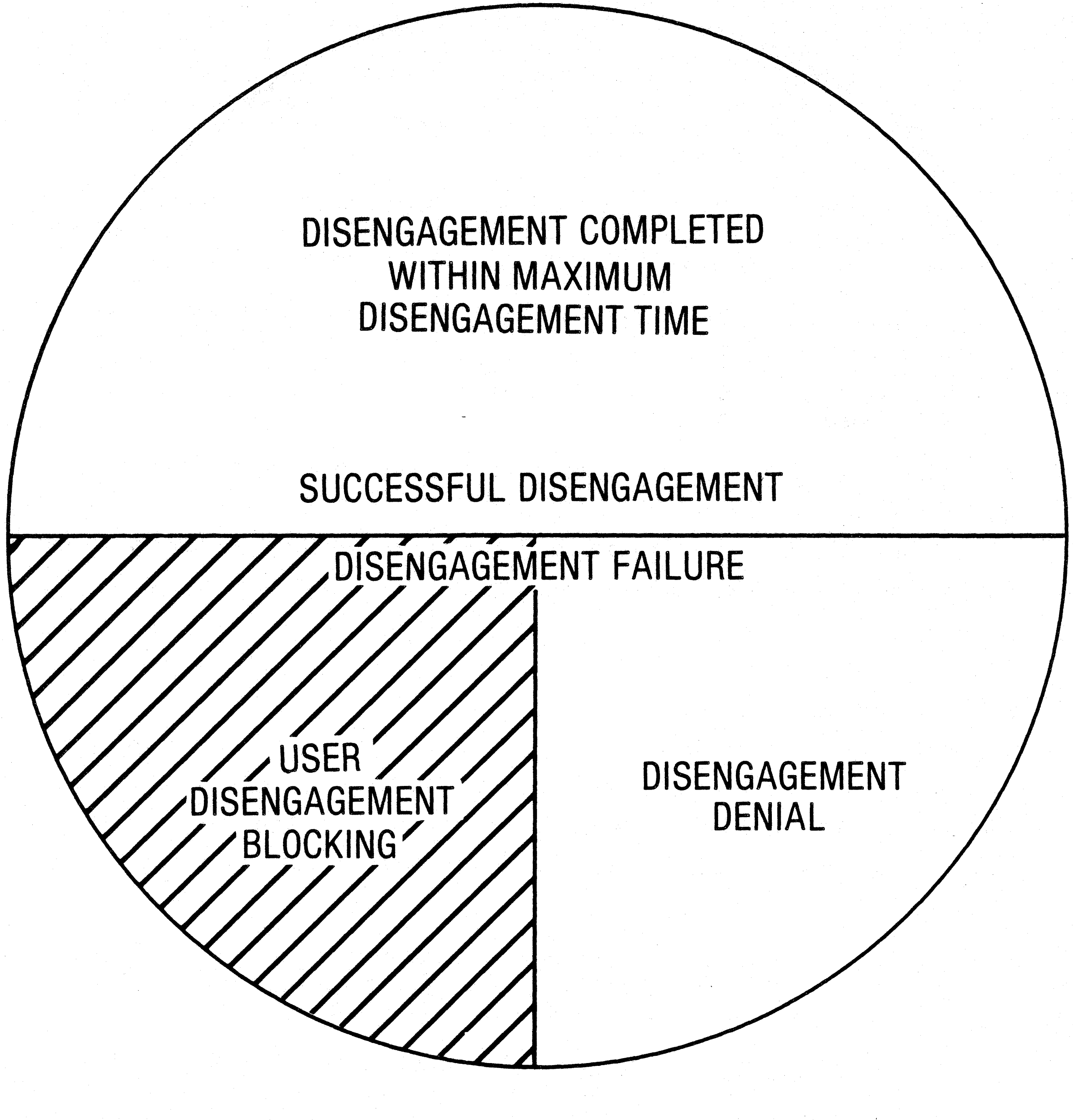

DISENGAGEMENT COMPLETED WITHIN MAXIMUM DISENGAGEMENT TIME
SUCCESSFUL DISENGAGEMENT
DISENGAGEMENT FAILURE
USER DISENGAGEMENT BLOCKING
DISENGAGEMENT DENIAL

Figure 10. Disengagement outcomes.

PRIMARY FUNCTIONS	OUTCOMES INCLUDED IN SAMPLE SPACE						
	SUCCESSFUL PERFORMANCE	CONTENT ERROR	LOCATION ERROR	SYSTEM NON-PERFORMANCE		USER NON-PERFORMANCE	EXTRA EVENT
ACCESS	✓	▨	✓	(DENIAL)	(OUTAGE)	✓	▨
BIT TRANSFER	✓	✓	✓	✓		✓	✓
BLOCK TRANSFER	✓	✓	✓	✓		✓	✓
DISENGAGEMENT	✓	▨	▨	✓		✓	▨

Figure 11. Outcome summary.

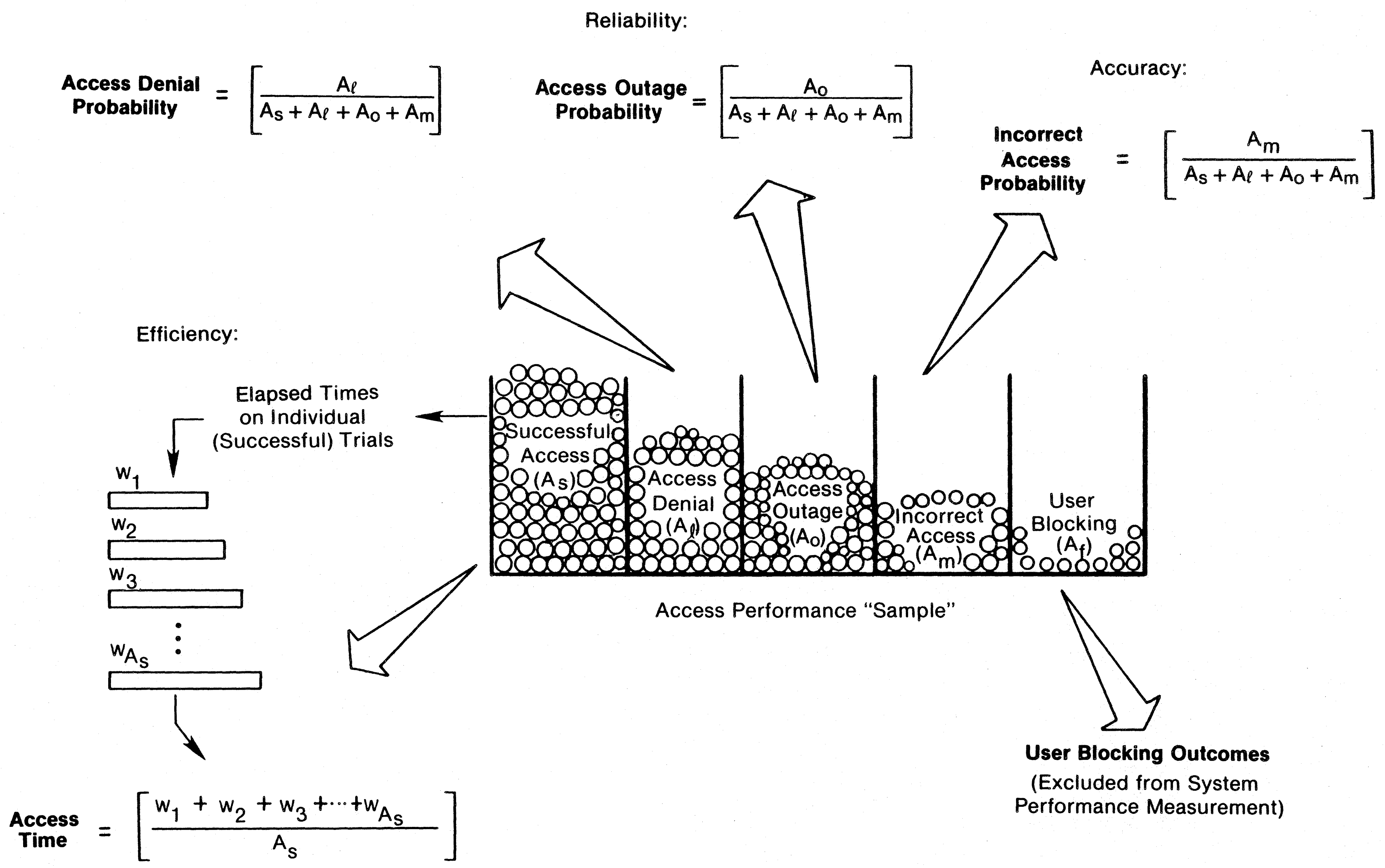

Reliability:
Access Denial Probability = $\left[\frac{A_\ell}{A_s + A_\ell + A_o + A_m}\right]$
Access Outage Probability = $\left[\frac{A_o}{A_s + A_\ell + A_o + A_m}\right]$
Accuracy:
Incorrect Access Probability = $\left[\frac{A_m}{A_s + A_\ell + A_o + A_m}\right]$
Efficiency:
Elapsed Times on Individual (Successful) Trials
w_1
w_2
w_3
w_{A_s}
Access Time = $\left[\frac{w_1 + w_2 + w_3 + \cdots + w_{A_s}}{A_s}\right]$
Successful Access (A_s)
Access Denial (A_ℓ)
Access Outage (A_o)
Incorrect Access (A_m)
User Blocking (A_f)
Access Performance "Sample"
User Blocking Outcomes
(Excluded from System Performance Measurement)

Figure 12. Parameter definition example--access parameters.

Access Denial Probability is the ratio of total access attempts that result in Access Denial to total access attempts in an access performance sample (excluding User Blocking outcomes).

Access Outage Probability is the ratio of total access attempts that result in Access Outage to total access attempts in an access performance sample (excluding User Blocking outcomes).

A key aspect of the ANS X3.102 parameter definitions is their expression in mathematical form. As noted earlier, this approach eliminates the ambiguity often associated with purely narrative parameter definitions, and also provides a standard procedure for calculating the parameter values. The mathematical parameter definitions are based, in each case, on the concept of an access performance sample--that is, a large number of successive access attempts distributed in appropriate outcome categories or "bins".

Values for the access parameters are calculated from the data recorded during a series of access attempts as shown in Figure 12. The value of the speed parameter Access Time is calculated by adding the individual elapsed times (W_i) for all A_s Successful Access outcomes and dividing by A_s. The value of the accuracy parameter Incorrect Access Probability is calculated by dividing the total number of Incorrect Access outcomes (A_m) by the total number of access outcomes in the reduced sample, excluding User Blocking outcomes--i.e., dividing A_m by ($A_s + A_\ell + A_o + A_m$). Similarly, the values of the reliability parameters Access Denial Probability and Access Outage Probability are calculated by dividing the appropriate failure outcome (A_ℓ or A_o) by the total number of access trials in the reduced sample ($A_s + A_\ell + A_o + A_m$). User Blocking outcomes are excluded in calculating the access failure probabilities to ensure the comparability of values measured under different usage conditions.

The preceding descriptions have referred to a "maximum access time" beyond which an access attempt is declared a failure for performance assessment purposes. To ensure comparability, ANS X3.102 defines a fixed value for this timeout point--three times the Access Time specified for the service. Thus, a timeout is declared whenever the observed access delay exceeds three times the value the user expects on a typical access attempt. Note that this timeout constant has significance only in the assessment of performance--access attempts that extend beyond the timeout point need not be abandoned in actual usage. Note also that additional characteristics of the Access Time distribution (e.g., the variance or the 95-percent points) may also be of interest in some situations.

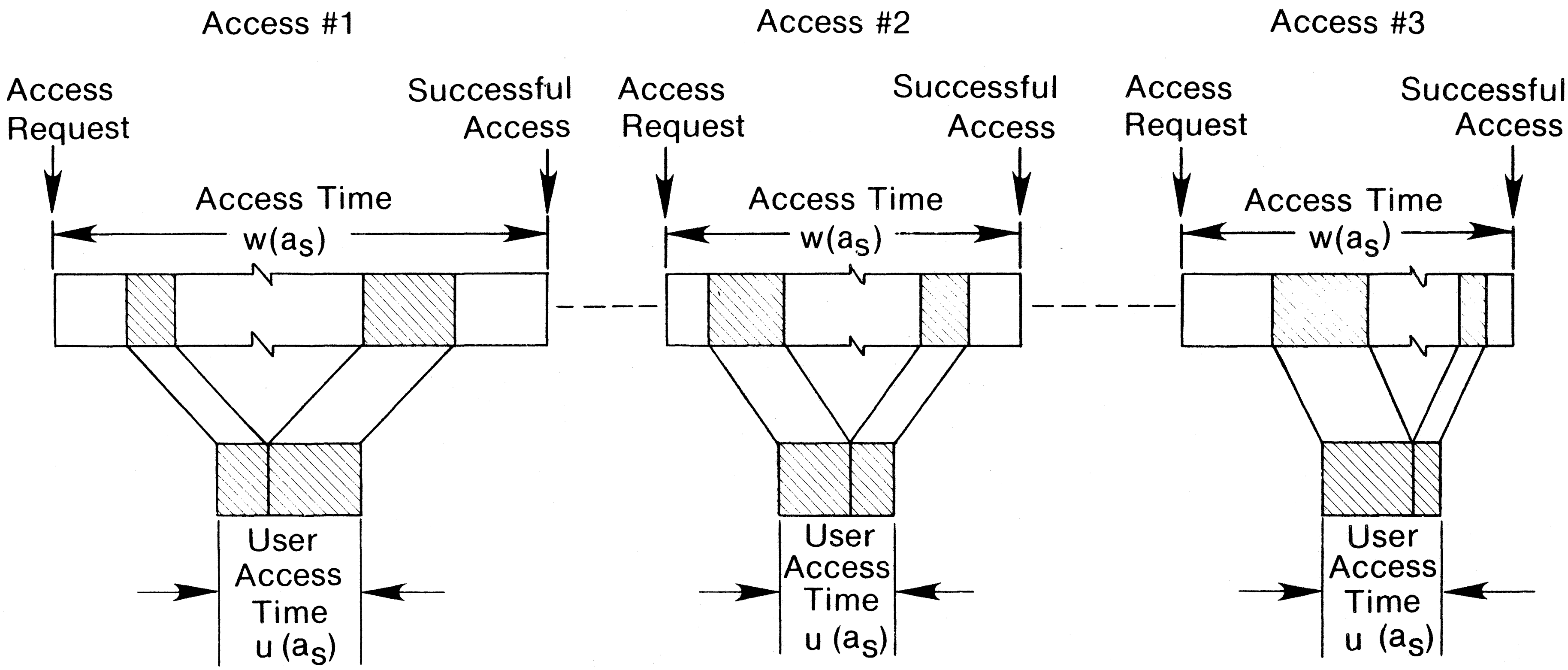
The same general approach used in the access case was followed in selecting and defining performance parameters for the user information transfer and disengagement functions. A separate probability parameter was defined to express the likelihood of each possible failure outcome. An "average elapsed time" parameter was defined, in each case, to express the delay associated with successful performance. Two additional parameters were defined to express user information transfer performance with respect to "throughput" and "availability." The selected primary parameters are summarized in Section 3.2.6.

3.2.5 Ancillary Parameters

As discussed earlier, the primary ANS X3.102 parameters are user dependent. To make these parameters useful in describing unilateral system performance, it is necessary to develop a method of expressing the user contribution to observed delays. Standard X3.102 defines a small set of "ancillary" performance parameters that fulfill this need.

The ancillary parameters are developed by dividing the total performance time for an associated primary function into alternating periods of system and user responsibility, and then calculating the average proportion of total performance time for which the users are responsible. As a simple illustration, consider the voice telephone access example discussed earlier (Figure 4). The total performance time for the access function is the time between the calling user's "off-hook" action and the called party's answer. This total performance time can be divided into alternate periods of system and user responsibility by noting, at any time, which entity must produce the next interface event. During the period between "off-hook" and dial tone, the system is responsible (for producing dial tone); during the period between dial tone and dialing the first digit, the user is responsible; and so on. The ancillary parameter User Fraction of Access Time expresses the average proportion of total Access Time that is attributable to the user activities.

Figure 13 illustrates the approach used in defining the ancillary parameters in more detail, using the primary function of access as an example. The figure depicts a series of successful access attempts, each having a total access time $w(a_s)$ and a total user access time $u(a_s)$. The latter quantity represents the total access time attributable to user responsibility on each particular trial. The ancillary parameter User Fraction of Access Time is calculated by adding the user access time values over a suitable number of



$$\left\{ \text{User Fraction of Access Time} \right\} = \left\{ \frac{\text{Total User Access Time}}{\text{Total Access Time}} \right\} = \left\{ \frac{\overset{A_s}{\sum} \; u(a_s)}{\overset{A_s}{\sum} \; w(a_s)} \right\}$$

Figure 13. Ancillary parameter definition--access example.

successful access attempts, and then dividing by the corresponding sum of the total access times. Only Successful Access outcomes are considered in estimating User Fraction of Access Time in order to avoid biasing the average with unrepresentative values.

A similiar approach was used in defining the ancillary parameters for the user information transfer and disengagement functions. The standard defines a total of four ancillary parameters: User Fraction of Access Time, User Fraction of Block Transfer Time, User Fraction of Input/Output Time, and User Fraction of Disengagement Time. User Fraction of Input/Output Time describes the users' influence on the primary parameter User Information Bit Transfer Rate, as discussed in Section 4.

In addition to permitting the specification of user-independent parameter values, the ancillary parameters provide a basis for identifying the entity (user or system) responsible for timeout performance failures. This application of the ancillary parameters is also discussed in Section 4.

3.2.6 Problem Solutions - Summary

It was noted earlier that the development of ANS X3.102 required the solution of three performance description problems. The technical approach adopted in the standard provides a solution to each of these problems, as summarized below.

1. System dependence. The standard solves this problem by defining the performance parameters in terms of general, system-independent reference events.

2. Detailed Parameter Definition. The standard solves this problem by using sample spaces and mathematical equations as the major parameter definition tools. Sample spaces encourage the analyst to consider and carefully define all relevant outcomes of a performance trial. Equation definitions eliminate the ambiguity often associated with purely narrative definitions.

3. User Delay. The standard solves this problem through the use of the ancillary performance parameters. These enable providers to remove user delays from the primary speed parameter values when a description of unilateral system performance is required.

4. UNDERSTANDING THE PARAMETERS

Suppose you were asked, with no prior explanation, to use the parameters listed in Table 1 in specifying a data communication service requirement. What questions would you ask about each parameter before beginning the specification? For most potential users, the key questions about each parameter would include the following:

- o What is the meaning of this parameter in simple, straightforward, user-oriented terms?
- o How is this parameter related to other widely used performance parameters?
- o Why is the value of this parameter significant to the data communication user?
- o What are the best and worst possible values for this parameter, and what are the implications of these values?
- o What typical values might be specified for this parameter in characterizing: (1) performance requirements for familiar user applications and (2) performance capabilities of existing data communication systems and services?
- o How do the values for this parameter influence key decisions in data communication system design?
- o Conversely, how are the values for this parameter influenced by key decisions in data communication system design?

This section answers these questions by means of tutorial essay descriptions of the ANS X3.102 parameters. The individual parameter descriptions are presented in order by function (access, user information transfer, disengagement), with the four ancillary parameters described last. A separate description is provided for each primary parameter, with the exception that corresponding bit- and block-oriented transfer parameters (e.g., Bit Error Probability and Block Error Probability) are described together. The ancillary parameters are also described together to emphasize interdependencies and definitional similarities. Readers are referred to ANS X3.102 and to other reports for more rigorous parameter definitions and for application details.

4.1 Access Parameters

Requesting access to a data communication service is a little like going to the post office to mail a letter. Your objective is to get your letter (message) on its way to the intended destination as soon as possible. Any delay (e.g., standing in line behind other customers) is an undesirable waste of time.

An ideal data communication service (or an ideal postal service) would accept your message, and start it on its way to the intended destination immediately, every time you requested service. Such a service would tend to be prohibitively expensive, so you tolerate some delay. But you still judge the service on the basis of how closely it approaches that ideal.

The fundamental user concerns about service performance are also similar in the two cases:

- Speed. How long will I have to wait to get my message started on its way (assuming that I am successful in doing this)?

- Accuracy. What is the likelihood that the service will process my service request incorrectly, thus establishing an information path that directs the message to the wrong destination?

- Reliability. What is the likelihood that I will be denied service as a result of congestion or a system outage?

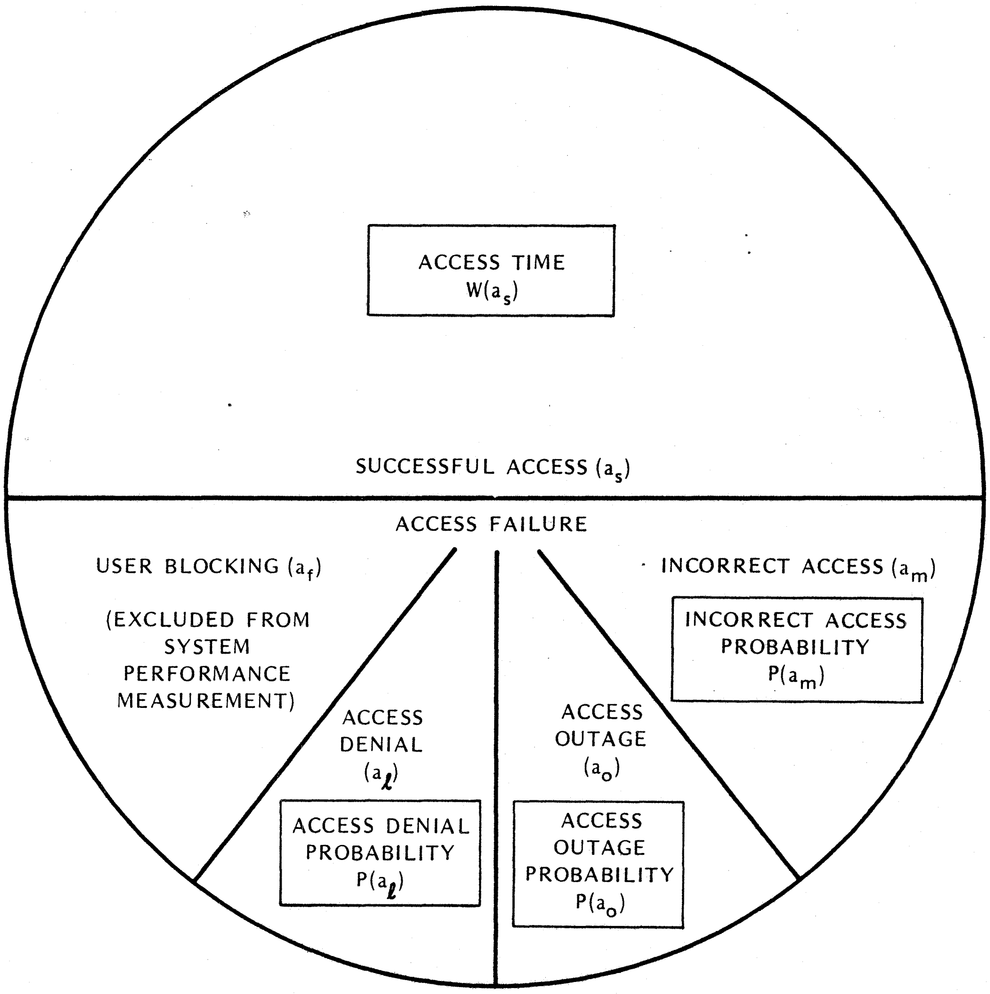
The standard defines four primary performance parameters that directly express these user concerns: Access Time, Incorrect Access Probability, Access Denial Probability, and Access Outage Probability. See Figure 14.

4.1.1 Access Time

Access Time is the average time the user must wait after requesting data communication service for the system to begin accepting user information for transmission. Access Time begins on issuance of an Access Request or its implied equivalent (e.g., system polling) at the originating user/system interface. It ends when the first bit of source user information is input to the system (after connection establishment in connection-oriented services). Access Time values are calculated only on access attempts that result in Successful Access.

The Access Request event will have different forms in different systems. Three examples of explicit Access Request signals have been cited in Section 3.2.2. An Access Request can also be implicit, as in the case where a user asks the system to poll him for possible messages at some specific future time.

Successful Access is defined to occur when at least one bit of user information is transferred from a source user to the system within the specified maximum access time. In the case of connection-oriented systems, there is the additional requirement that the intended nonoriginating user must have been contacted and committed to the session prior to the start of user information transfer. The latter requirement distinguishes Successful Access outcomes from Incorrect Access outcomes.


ACCESS TIME
$W(a_s)$
SUCCESSFUL ACCESS (a_s)
ACCESS FAILURE
USER BLOCKING (a_f)
(EXCLUDED FROM SYSTEM PERFORMANCE MEASUREMENT)
ACCESS DENIAL (a_ℓ)
ACCESS DENIAL PROBABILITY $P(a_\ell)$
ACCESS OUTAGE (a_o)
ACCESS OUTAGE PROBABILITY $P(a_o)$
INCORRECT ACCESS (a_m)
INCORRECT ACCESS PROBABILITY $P(a_m)$

ACCESS PARAMETERS

1. Access Time = $W(a_s) = \dfrac{1}{A_s}\displaystyle\sum_{a_s=1}^{A_s} w(a_s)$
2. Incorrect Access Probability = $P(a_m) = A_m/A'$
3. Access Denial Probability = $P(a_\ell) = A_\ell/A'$
4. Access Outage Probability = $P(a_o) = A_o/A'$

DEFINITIONS

A' = Total number of access attempts counted during an access parameter measurement: $A_s + A_m + A_\ell + A_o$

A_s = Total number of Successful Access outcomes counted during an access parameter measurement.

A_ℓ = Total number of Access Denials counted during an access parameter measurement.

A_o = Total number of Access Outage outcomes counted during an access parameter measurement.

A_m = Total number of Incorrect Access outcomes counted during an access parameter measurement.

$t(a)$ = Time a particular access attempt starts

$t(a_s)$ = Time Successful Access is attained on a particular access attempt.

$w(a_s)$ = Value of access time measured on a particular successful access attempt: $t(a_s) - t(a)$

Figure 14. Access parameters.

The relationship between Access Time and the traditional telephone switching parameters Dial Tone Delay and Time to Receipt of Audible Ringing has been discussed earlier. To review briefly, Access Time describes the total time between "off hook" and called party answer, while the latter two parameters describe specific intervals of system performance within that time.

Access Time is closely related to another commonly used switching parameter, Connection Establishment Time. The latter parameter is defined in ANS X3.79 as follows:

> "Connection Establishment Time represents the time interval required to establish an information transfer channel to the desired destination...Connection Establishment Time begins when network service is requested by going off-hook or activating the call request (CRQ) function at the DTE-DCE interface. It ends when clear to send (CB) or equivalent function is activated at the DTE-DCE interface at either the calling or called station, whichever transmits first."

Connection Establishment Time differs from Access Time in two major respects:

1. The starting and ending events are defined to occur at the DTE-DCE interface rather than at the end user/system interface.

2. The ending event is a system-generated "clear to send" signal rather than the actual start of user information transfer.

The events used in defining Access Time are more appropriate in a user-oriented standard because they are observable at the end user interfaces and are system independent. The difference in timing between the two event pairs can be substantial, particularly in layered architecture systems.

Access Time also has a close kinship with the "average waiting time" parameter defined in queueing theory (Kleinrock, 1976). The latter parameter describes the average time a customer must spend waiting in queue on any given arrival before receiving some desired service. In the case of data communications, the transfer of user information is the desired service; issuing an Access Request denotes queue entry; and the start of user information transfer denotes the end of waiting and the beginning of service.

Access Time differs from "average waiting time" in one important respect: Access Time is the average of a truncated distribution. Figure 15 illustrates the meaning of this difference. If we measure a large number of individual delay values and plot the relative frequency of occurrence of each possible value, the result is a histogram or distribution of delay values. In general,

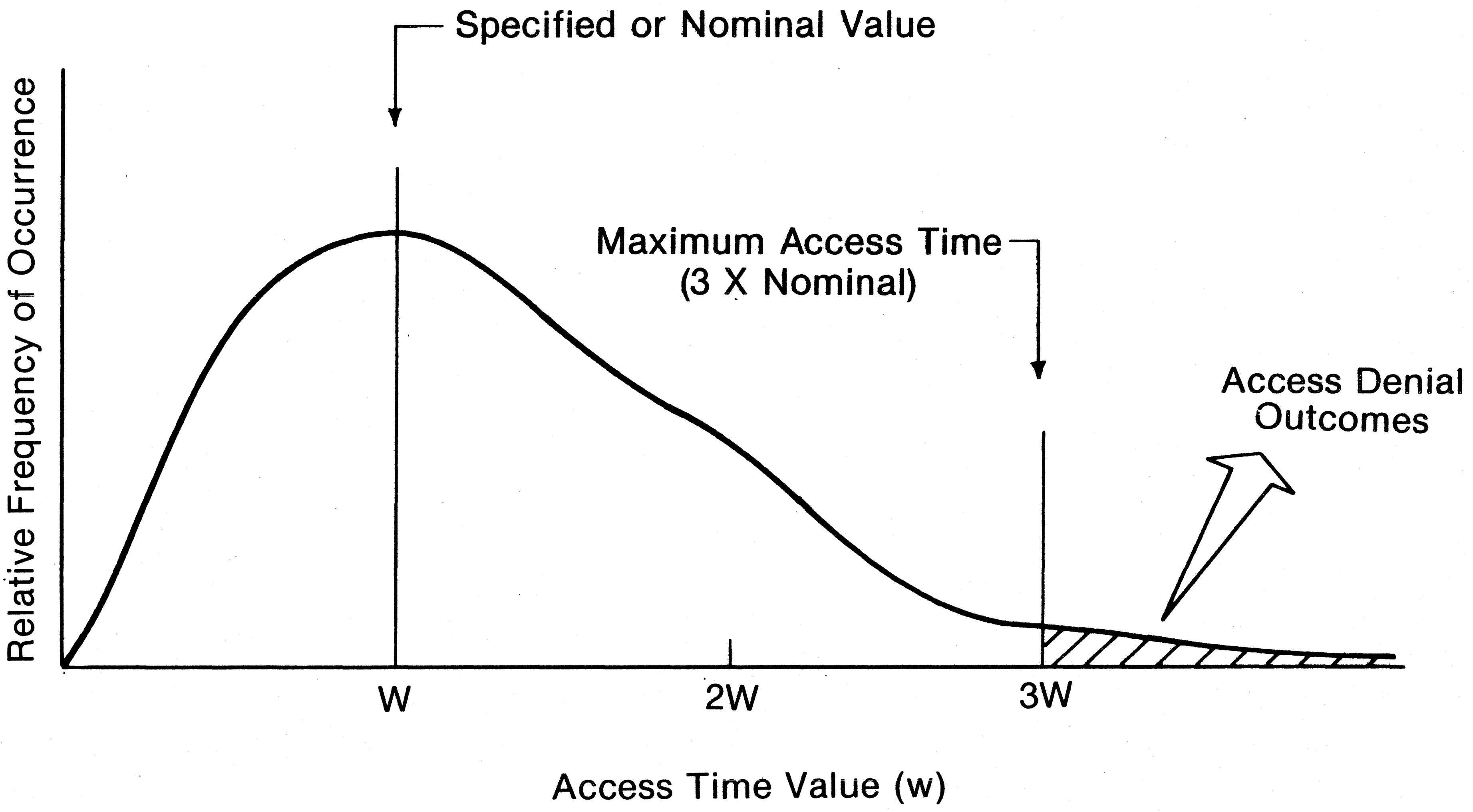


Figure 15. Truncation of the access time distribution.

such a distribution will be unbounded on the right, since extremely long delays will occasionally occur. It is desirable to exclude such extreme values in calculating an average for two reasons:

1. Their observation requires, in the limit, infinite patience on the part of the observer.

2. They can unduly influence an average because of their large magnitude.

The standard excludes abnormally long delay values in calculating Access Time by truncating (cutting off) the Access Time distribution at a value three times the nominal value specified for the service (the mean before truncation). The standard counts access attempts which last longer than this maximum access time as failures for performance assessment purposes. In cases where the system is responsible for the excessive delay, the failure is classified as an Access Denial. In cases where the user is responsible for the excessive delay, the failure is excluded from consideration as if it had never occurred. The same approach is used in defining all other time averages in the standard.

The timeout constant three was chosen, somewhat arbitrarily, as a value that would be generally consistent with user expectations about service performance and would include most of the area under a typical delay time distribution. For further discussion of this factor, see Crow (1979).

Why is Access Time significant to data communication users? There are two reasons:

o Time is Money. Users value their time, and quite rightly view time spent establishing data communication as unproductive time. When a user makes many accesses per day (e.g., an airline reservations clerk), the monetary cost of a lengthy Access Time can be significant, especially if frustration affects the user's work attitude.

o Information Ages. Virtually all information becomes less relevant and, therefore, less valuable with the passage of time. As an example, information on the position of a fast-moving aircraft (or the price of fast-moving stock) may become useless very quickly.

In essence, Access Time is the price we pay for the economic benefit of sharing a communication resource with others. A user who cannot tolerate the typical 30 to 40 second Access Time of the public telephone network can reduce the time by about three-fourths by having a leased telephone service installed, since only the nonoriginating user's delay in answering is then involved. Of course, having a leased line connection to each frequently called party would be prohibitively expensive. There is no theoretical upper limit on an Access Time specification, although extremely long access delays imply a service that is of little value to most users.

Appropriate user requirements for Access Time vary widely as a function of user application. At the lower extreme are critical, real-time applications. At the upper extreme are routine record communications and electronic mail. A recent ITS study recommends 0.9 percentile Access Times in the range of 0.15 to 4.0 seconds for packet-switching network users (Nesenbergs, et al., 1980).[5] An average Access Time of 15 seconds was specified in a Request for Quotations (RFQ) issued by the Environmental Protection Agency for the data communications portion of a nationwide timesharing network (EPA, 1980).

Access Time values are strongly influenced by system design. Access delay is caused primarily by resource sharing (that is, switching), so it is not surprising that dedicated, nonswitched services provide the shortest Access Times. One might expect Access Time to be zero with such services, but this is generally not the case. Even when data communication facilities are not shared, they are not necessarily used continuously. If they are not, a short access delay will often be encountered at the beginning of each usage period, while the system synchronizes equipment at the two user locations (Gray, 1972; Kimmett and Seitz, 1978). Such delays will normally be in the millisecond range if no user dependence is involved. Often user delay will greatly increase the total delay, as in the case of seeking the attention of a terminal operator.

The standard distinguishes two general categories of resource-sharing communication services: connection-oriented and connectionless. Connection-oriented services require that the intended nonoriginating user (called party) be contacted and committed to a session prior to the start of user information transfer. Traditional circuit-switched and modern virtual circuit services fall into this category. Connectionless services allow user information transfer to begin without such a commitment. Traditional message-switched and modern datagram services fall into this category.

Access Times are normally longer in connection-oriented services than in connectionless services because the process of obtaining nonoriginating user

[5] The 0.9-percentile values are exceeded 10% of the time. The corresponding averages may be considerably lower.

commitment takes time. Access Times in connectionless services are typically less than 5 seconds. Access Times in connection-oriented services typically range from a few seconds to 30 seconds or more, depending on the connection protocol and whether operator responses are required. An Access Time of 7 seconds has been measured for ARPANET operator-to-program communications via the Telnet protocol (Payne, 1978). More recently, a value of 1.8 seconds has been measured for ARPANET program-to-program communications (Wortendyke et al., 1982).

The decision to defer destination user commitment until the user information transfer phase decreases Access Time in connectionless services, but often increases user information transfer delays by a corresponding amount. This effect is discussed in Section 4.2.1. The effect of user dependence on Access Time is discussed in Section 4.4.

4.1.2 Incorrect Access Probability

Incorrect Access Probability expresses the likelihood that user information will be transmitted on an improper path as a result of a system error during the access process. As noted earlier, it is the ratio of total Incorrect Access outcomes to total access attempts in a performance sample, excluding access attempts ending in User Blocking.

Incorrect Access is essentially the case of a "wrong number." It occurs when the system establishes a physical or logical connection to a user other than the one intended by the data communication session originator, and then does not correct the error before the start of user information transfer. Incorrect Access can occur only in connection-oriented services, since no physical or logical connection is established between users in connectionless services. Incorrect Access is distinguished from Successful Access by the fact that the intended nonoriginating user is not contacted and committed to the data communication session during the access performance period.

What kinds of system errors cause Incorrect Access? Perhaps the simplest cause is a transmission error in communicating the signaling information from the originating user to the first switch, or between switches in the system. A second possible cause is a switch error in translating the signaling information into (for example) a physical crosspoint connection. The latter type of error is normally infrequent and random, since it is likely to be due to a marginal switch component. However, such errors can also be systematic, as in the case of an improper numbering change within the system.

Incorrect Access is closely associated with what common carriers and switch manufacturers refer to as "mishandled" or "misprocessed" calls (Kobylar and Malec, 1973; Malec, 1975). However, Incorrect Access Probability differs from the "misprocessed call" probability in two respects:

1. Misprocessed calls typically include calls that are not completed (i.e., the switch does not respond) as well as calls that are misconnected.

2. Misprocessed call probability describes the performance of a switch rather than that of an end-to-end service. This difference can be very significant in systems that provide automatic answerback or other connection verification features, as discussed later.

Incorrect Access also has an obvious association with the concept of misdelivery. The following definition of misdelivery is typical of those encountered in the literature (DCA, 1975):

> "Misdelivery is defined as the delivery of a segment (i.e., message) in violation of the originally specified addressing information."

Incorrect Access and misdelivery are often related as cause and effect in connection-oriented systems. If a system establishes a circuit connection to an incorrect (but compatible) destination during access, and does not detect the error prior to the start of user information transfer, it is likely that at least some user information will be misdelivered to that destination.

However, Incorrect Access does not invariably result in misdelivery. This is because the test for Incorrect Access is a negative test. That is, Incorrect Access is declared (in a connection-oriented service) when the intended nonoriginating user is not contacted and committed to the session prior to the start of user information transfer. This test does not distinguish between the case where some other user is contacted and the case where the commitment step in access is somehow "short-circuited" by the system, with no other user contacted. Misdelivery will normally occur in the former case, but data loss will be the end effect in the latter.

These two possible consequences of Incorrect Access, misdelivery and loss, make Incorrect Access Probability very significant to data communication users. In the case of misdelivery, the risk to the source user is twofold. First, the data may be delivered to a destination user who has the desire and the capability to exploit it to the source user's disadvantage. An example would be the transmission of information on prospective financial transactions over public data communication facilities without the use of encryption.

Second, the source user may be led to believe that the information has been delivered to the intended destination when, in fact, it has not. The source user may then base subsequent actions on a false assumption. An example would be a failure to update important cost data in a remote management information system. Only the latter risk is applicable in the case of data loss.

Like all probabilities, Incorrect Access Probability has possible values between zero and one. A value of zero would indicate that Incorrect Access is impossible, a situation that can only be achieved in systems that perform no switching during the access phase. Such systems include connectionless systems (which perform their switching during' the user information transfer phase) and nonswitched, connection-oriented systems (e.g., a system using dedicated leased lines). An Incorrect Access Probability value of one would indicate that Incorrect Access is certain. Such a value would suggest that a systematic switching error had occurred.

Quantitative data on user requirements for Incorrect Access Probability is scarce. Nesenbergs et al. (1980) suggest a range of values of about 10^{-10} for interactive packet-switching users, while a somewhat more stringent value (10^{-11}) is specified in DCA (1975). Neither estimate appears to be based on a quantitative user impact assessment. Incorrect Access may be no more than a nuisance in a benign communication environment. In such situations, users may well specify an acceptable, but rather arbitrary value that is easily attained by most existing systems (for example, 10^{-4}). A value of 10^{-5} is specified in the EPA RFQ cited earlier (EPA, 1980).

System performance data on Incorrect Access Probability is also relatively scarce. As noted earlier, Incorrect Access can be caused by system errors either in transmitting or in interpreting the signaling information. Errors in transmitting signaling information are much more frequent in older systems using direct current, multi-frequency, or single frequency switching than in modern systems using common channel signaling (e.g., 10^{-4} vs. 10^{-8}). A value of 10^{-4} appears to be a typical telephone company objective for "mishandled call" probability in a single switch (Kobylar and Malec, 1973). In estimating Incorrect Access Probability, this number would be reduced by the fact that not all "mishandled" calls are misconnected, and increased by the fact that a typical telephone circuit normally involves several switches in tandem (series). A value of 10^{-5} is probably a reasonable estimate of the likelihood of misconnection as a result of switching error in a typical circuit-switched system.

Of course, errors in signaling and switching may or may not cause Incorrect Access. The likelihood of Incorrect Access given such an error depends on two factors:

1. Whether the error results in contact with a terminal (or terminal function) compatible with that of the intended called party.

2. Whether the system provides circuit verification techniques, such as automatic answerback (AT&T, 1968).

The likelihood of connection with a compatible terminal depends on the mix of terminals in the network. Answerback schemes can reduce Incorrect Access Probability by a factor of hundreds or thousands.

In general, virtual circuit switching systems have lower Incorrect Access Probabilities than conventional "physical circuit" switching systems due to their more effective use of end-to-end error control. As an example, military packet-switching networks have been designed with a 32-bit Cyclic Redundancy Check (CRC) on all transmitted data. Nesenbergs et al. (1980) estimate that such CRC checks should provide an undetected error rate for circuit establishment messages better than 2^{-32} (10^{-10}).

One practical limitation of the parameter Incorrect Access Probability should be noted in conclusion: it does not include situations where an unintended destination is contacted as a result of a user error in inputting the addressing information (e.g., misdialing). Such errors should be considered in establishing user requirements for system performance.

4.1.3 Access Denial Probability

Access Denial Probability expresses the likelihood of system blocking during access. Access Denial Probability is distinguished from Access Outage Probability, which is described in the next section. It is the ratio of total access attempts that result in Access Denial to total access attempts in a performance sample, excluding access attempts ending in User Blocking.

Access Denial can occur in two basic ways:

1. The system issues a blocking signal to the originating user during the access period, thereby terminating the access attempt.

2. The system makes some response but delays excessively in performing required actions during the access period, with the result that user information transfer is not initiated within the maximum access time.

What is a "system blocking" signal? In essence, it is the system's way of telling the user that it cannot provide data communication service on a particular access request because some required system facility is currently unavailable. The required facility (e.g., a particular trunk circuit) may be unavailable because it is serving another user (that is, it is busy) or because it is not operational (e.g., due to a component failure). In the latter case, the cause of blocking is an outage within the system but its effect is a system blocking signal at the user/system interface--Access Denial. If the outage prevented any system response at the user/system interface, the failure would be classified as an Access Outage.

A system blocking signal constitutes a definite denial, rather than just a delay or deferral, of an access attempt. A familiar example of a system blocking signal is the two cycle-per-second "circuit busy" signal in the public switched telephone network. Such a signal tells the user that the current access attempt will not succeed no matter how long he hangs on. His only way to achieve Successful Access is to hang up and try again. System blocking signals should be distinguished from signals that merely delay Successful Access, as in the case of the familiar "all reservation clerks are busy--please do not hang up" recording.

Systems experiencing congestion or outage may not respond or may delay excessively in responding to a user's access request. Virtually everyone has experienced such a situation at one time or another in making a long distance telephone call. The system gives a few promising clicks after dialing and then seems to go dead. Often, the optimistic user who waits is eventually disconnected by the system. This situation is defined as an Access Denial if the delay persists longer than three times the Access Time specified for the service. Access Denials are distinguished from User Blocking outcomes by comparison of the ancillary parameter values, as described in Section 4.4.

Access Denial Probability is closely associated with what is known as the "Grade of Service" or "Blocking Probability" in circuit-switched systems. In general, such systems cannot economically provide access to all users during the worst case loading period (known as the "busy hour"). Instead, they are designed to serve all but a certain (usually very small) fraction of calls attempted during that period. The fraction, P, of call attempts not served by a connection-oriented system during the busy hour is its Grade of Service or Blocking Probability. The symbol P.01 indicates that one call in a hundred will be blocked; P.04 indicates that four calls in a hundred will be blocked;

and so on. Customers accept a small Blocking Probability in exchange for the economic benefits of resource sharing. But if the Blocking Probability is too high, they may abandon the service in favor of more acceptable alternatives.

Access Denial Probability is most significant to the user when alternative means of data communication are not available. In such cases, the user has no choice but to continue attempting to access the denying system. The negative consequences are similar to those cited earlier for long Access Times; that is, loss of productive time and data aging. In general, a series of Access Denials is more detrimental to the user than a single access delay of equivalent duration, because each Access Denial nullifies previously completed access steps (e.g., dialing). There is a definite buildup of dissatisfaction with repeated Access Denials in the case of human users.

Access Denial Probability values range between zero and one. A value of zero implies that the user is never denied access--that is, the system is completely nonblocking. At the other extreme, a value of one implies a service that never actually serves the user.

In considering user requirements for Access Denial Probability, it is important to distinguish between what the user actually needs and what the user will accept if nothing else is available. There are switched data communication services with blocking probabilities of 0.4 and even higher (e.g., AUTOVON). There is also evidence of user dissatisfaction with such services (GAO, 1977). Access Denial Probabilities in the range of 1 to 5 percent are normally satisfactory in applications where data aging is slow. An example might be remote data processing for computer program development. Values of 10^{-3} or lower may be needed in critical real-time situations. An Access Denial Probability of 10^{-2} is specified in EPA (1980).

The system design feature that most strongly influences Access Denial Probability is the resource sharing or switching technique used. Many smaller data communication systems attempt no resource sharing and are, therefore, nonblocking. A familiar example is the use of a dedicated line interconnecting two users. However, such services often have relatively higher Access Outage Probabilities (see Section 4.1.4).

Large multi-user networks almost invariably use some type of resource sharing. The overall strategy of resource sharing is to take advantage of intermittent user demand by deliberately designing certain costly system elements (e.g., switches and trunks) with less capacity than would be needed to serve all users simultaneously. Under normal usage the design capacity will be adequate and the economic benefits realized without significant user inconvenience. But under heavier usage the design capacity will not be adequate, and the transmission of some traffic must be deferred.

The choice of resource sharing technique has an obvious impact on Access Denial Probability. Circuit-switched systems defer excess traffic by denying access and, therefore, have relatively high Access Denial Probabilities. Typical values for system blocking probability in well designed circuit-switched systems are in the range of 1 to 4 percent (AT&T, 1961; Duffy and Mercer, 1978).

Message-switched systems defer traffic by prolonging system storage rather than by denying access. Thus, they exhibit relatively low Access Denial Probabilities. In most message-switched systems, Successful Access can occur even when there is, at that time, no physical path possible between the source user and the intended destination. In such a situation, a switching node connected to the source user simply holds the message until the necessary communication links to the intended destination user are restored to service.

4.1.4 Access Outage Probability

Access Outage Probability expresses the likelihood that a system will be in an outage state which prevents it from responding to the originating user on any given access attempt. It is the ratio of total access attempts that result in Access Outage to total access attempts in a performance sample, excluding access attempts ending in User Blocking.

Access Outage is essentially the case of a dead system. A simple example is a telephone system that fails to provide dial tone. Access Outage may be due to either a system-wide failure (e.g., due to an extended loss of electrical power), or one or more specific component failures that impact the originating user (e.g, a failure in the local loop or local switching computer).

Access Outage Probability is closely associated with the concept of "availability." A typical definition of communication availability is that of ANSI (1974):

> "The portion of a selected time interval during which the information path is capable of performing its assigned data communications function. Availability is expressed as a percentage."

The significance of Access Outage Probability to the user depends on whether alternative means of data communication are available. If no alternative system is available, the user has no choice but to notify the service provider of the outage and await repairs. The principal consequence then is data aging, since most users will not repeatedly reattempt access when an outage is known to exist. If an alternative system is available, Access Outage Probability expresses the likelihood that it will be needed. Access Outage Probability is thus useful in assessing the need for, and potential use of, backup services.

Access Outage Probability values range between zero and one. A value of zero implies a system that never has an outage. At the other extreme, a value of one implies a service that is always "out" and, hence, is of no value to the user.

A rough upper bound on user requirements for Access Outage Probability can be derived from specified values for availability (A). If a user attempts access at regular (or random) intervals during a time period of interest, and the system is "down" (1-A) percent of the time during that period, it follows that the user will encounter Access Outage on at most (1-A)% of the trials. Observed Access Outage Probabilities will typically be somewhat lower than (1-A), since not all outages will prevent the system from responding to a user's access request.

Data on user requirements for availability are relatively plentiful. Typical values in current requirement specifications are in the range of 90% to 99.9%, with values above 98% much more common than those below. Assuming uniform sampling, the corresponding worst-case Access Outage Probability values would be in the 10^{-1} to 10^{-3} range.

The system design feature that most strongly influences Access Outage Probability is the inherent reliability of the system facilities. Replication of system components and the provision of alternate paths are the most common methods for improving Access Outage Probability. For this reason, large multi-user networks tend to provide a better (lower) Access Outage Probability than smaller systems with dedicated facilities.

Availability specifications for existing data communication systems are mostly in the range of 98% to 99.9%, suggesting conservative Access Outage Probability values in the neighborhood of 10^{-2} to 10^{-3}. As noted earlier, an availability value of 98% is probably typical of dedicated communication links (including the modems but not the terminals). Specified availability values for switched services are often slightly higher than those for dedicated services--typically, in the neighborhood of 99%. The 1.64% ARPANET IMP "down rate" cited in Kleinrock (1976) would imply an Access Outage Probability in the neighborhood of 10^{-2}. A measured ARPANET Access Outage Probability of 2.6×10^{-3} is reported in Wortendyke et al. (1982).

4.2 User Information Transfer Parameters

User performance concerns during the user information transfer phase may also be grouped in three general categories:

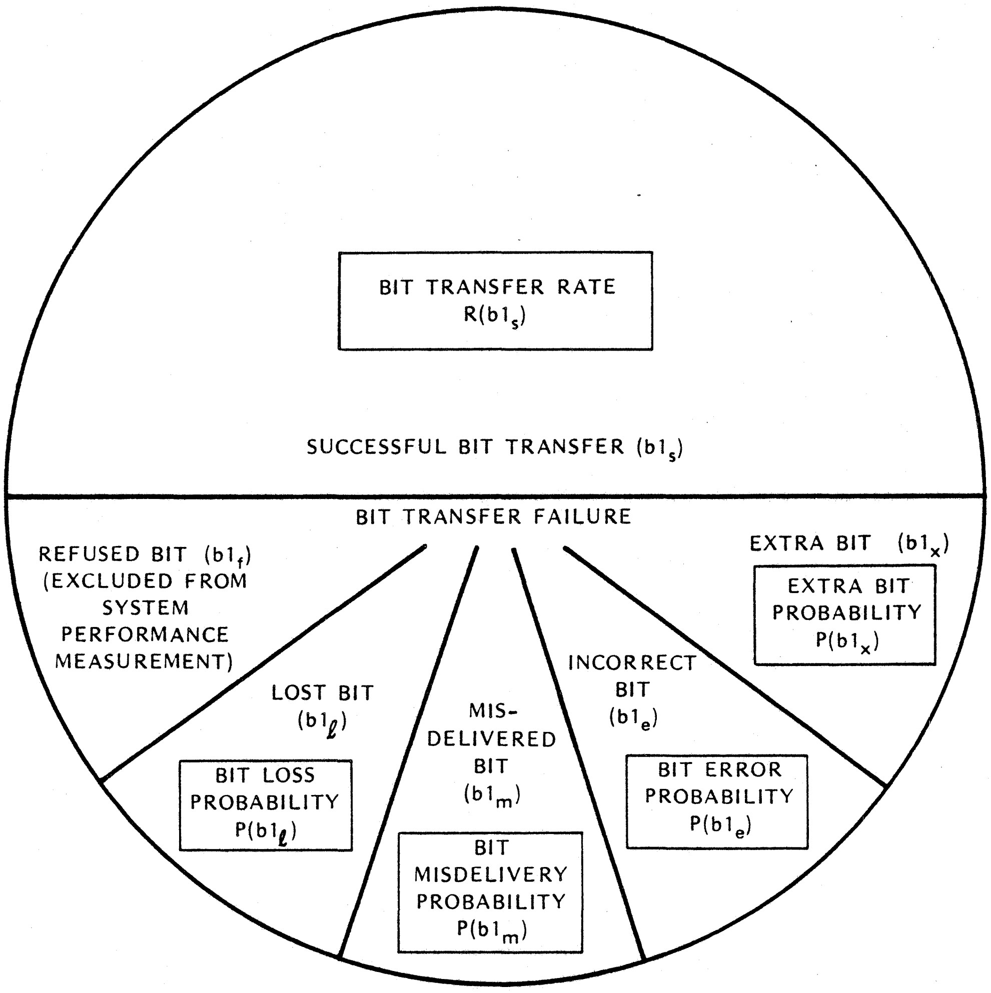
> Speed. What delay will my information experience in traversing the data communication system? What rate of information flow will the system allow?
>
> Accuracy. What is the likelihood that the system will alter or misdeliver the information? What is the likelihood that the system will deliver duplicate messages or create extra information not sent by the source user?
>
> Reliability. What is the likelihood that the system will lose information? What is the likelihood that the performance of the system will drop so low that the service will not be usable?

In general, the end user does not care how the information is physically transported so long as the end-to-end performance objectives are met.

The following sections describe the 11 primary user information transfer performance parameters specified in ANS X3.102. (See Figures 16 and 17.) These parameters are Block Transfer Time, Bit/Block Error Probability, Bit/Block Misdelivery Probability, Bit/Block Loss Probability, Extra Bit/Block Probability, User Information Bit Transfer Rate, and Transfer Denial Probability. Corresponding bit- and block-oriented parameters are described together, with differences in definition and impact highlighted. The block-oriented parameters are probably the most generally useful, although the bit-oriented parameters are often useful in comparing systems with different block sizes.

4.2.1 Block Transfer Time

Block Transfer Time expresses the total delay a user information unit experiences in transit between users. It is the average value of elapsed time between the start of a block transfer attempt at the source user/system interface and Successful Block Transfer at the system/destination user


BIT TRANSFER RATE
$R(b1_s)$
SUCCESSFUL BIT TRANSFER $(b1_s)$
BIT TRANSFER FAILURE
REFUSED BIT $(b1_f)$ (EXCLUDED FROM SYSTEM PERFORMANCE MEASUREMENT)
LOST BIT $(b1_\ell)$
BIT LOSS PROBABILITY $P(b1_\ell)$
MIS-DELIVERED BIT $(b1_m)$
BIT MISDELIVERY PROBABILITY $P(b1_m)$
INCORRECT BIT $(b1_e)$
BIT ERROR PROBABILITY $P(b1_e)$
EXTRA BIT $(b1_x)$
EXTRA BIT PROBABILITY $P(b1_x)$

BIT TRANSFER PARAMETERS

1. Bit Loss Probability = $P(b1_\ell) = B1_\ell/(B1' - B1_x)$
2. Bit Misdelivery Probability = $P(b1_m) = B1_m/(B1' - B1_\ell - B1_x)$
3. Bit Error Probability = $P(b1_e) = B1_e/(B1_s + B1_e)$
4. Extra Bit Probability = $P(b1_x) = B1_x/(B1' - B1_\ell)$
5. User Information Bit Transfer Rate = $R(b1_s) = \dfrac{B1_s}{w(b3_s)}$

DEFINITIONS

$B1'$ = Total number of Bit Transfer outcomes to be included in an individual UIT performance measurement (all Bit Transfer outcomes except $b1_f$).

$B1_s$ = Total number of Successful Bit Transfer outcomes counted during a UIT performance measurement.

$B1_f$ = Total number of Refused Bit outcomes counted during a UIT performance measurement.

$B1_\ell$ = Total number of Lost Bit outcomes counted during a UIT performance measurement.

$B1_m$ = Total number of Misdelivered Bit outcomes counted during a UIT performance measurement.

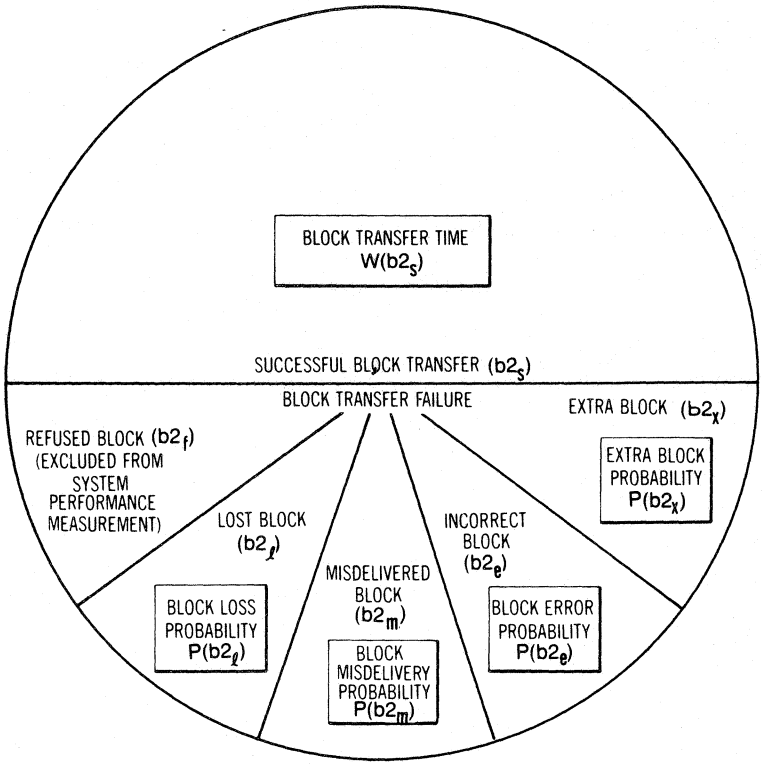
$B1_e$ = Total number of Incorrect Bit outcomes counted during a UIT performance measurement.

$B1_x$ = Total number of Extra Bit outcomes counted during a UIT performance measurement.

$w(b3_s)$ = Greater of input time $w(b3_I)$ or output time $w(b3_o)$ required to transfer sample to/from the system.

UIT = User Information Transfer.

Figure 16. User information bit transfer parameters.


BLOCK TRANSFER TIME
W(b2$_s$)
SUCCESSFUL BLØCK TRANSFER (b2$_s$)
BLOCK TRANSFER FAILURE
EXTRA BLOCK (b2$_x$)
EXTRA BLOCK PROBABILITY P(b2$_x$)
REFUSED BLOCK (b2$_f$) (EXCLUDED FROM SYSTEM PERFORMANCE MEASUREMENT)
LOST BLOCK (b2$_\ell$)
BLOCK LOSS PROBABILITY P(b2$_\ell$)
MISDELIVERED BLOCK (b2$_m$)
BLOCK MISDELIVERY PROBABILITY P(b2$_m$)
INCORRECT BLOCK (b2$_e$)
BLOCK ERROR PROBABILITY P(b2$_e$)

BLOCK TRANSFER PARAMETERS

1. Block Transfer Time = $W(b2_s) = \frac{1}{B2_s} \sum_{b2_s=1}^{B2_s} w(b2_s)$
2. Block Loss Probability = $P(b2_\ell) = B2_\ell/(B2' - B2_x)$
3. Block Misdelivery Probability = $P(b2_m) = B2_m/(B2' - B2_\ell - B2_x)$
4. Block Error Probability = $P(b2_e) = B2_e/(B2_s + B2_e)$
5. Extra Block Probability = $P(b2_x) = B2_x/(B2' - B2_\ell)$

DEFINITIONS

$B2'$ = Total number of block transfer outcomes to be included in an individual UIT performance measurement (All block transfer outcomes except $b2_f$).

$B2_s$ = Total number of Successful Block Transfer outcomes counted during a UIT performance measurement.

$B2_f$ = Total number of Refused Block outcomes counted during a UIT performance measurement.

$B2_\ell$ = Total number of Lost Block outcomes counted during a UIT performance measurement.

$B2_m$ = Total number of Misdelivered Block outcomes counted during a UIT performance measurement.

$B2_e$ = Total number of Incorrect Block outcomes counted during a UIT performance measurement.

$B2_x$ = Total number of Extra Block outcomes counted during a UIT performance measurement.

$t(b2)$ = Time a particular block transfer attempt starts.

$t(b2_s)$ = Time Successful Block Transfer is attained on a particular block transfer attempt.

$w(b2_s)$ = Value of block transfer time measured on a particular successful block transfer attempt: $t(b2_s) - t(b2)$

UIT = User Information Transfer.

Figure 17. User information block transfer parameters.

interface. Block Transfer Time values are calculated only on successful block transfer attempts.

As noted earlier, a user information block is defined as a contiguous group of user information bits delimited at the source user interface for transfer to a destination user as a unit. Thus, for instance, a block may be a single ASCII character, a card image, a computer word, or the information field of a frame, depending on the equipment and protocol characteristics of the user/system interface.

A block transfer attempt begins when the block has been input to the system and the system has been authorized to deliver it to the destination. As noted earlier, authorization may either be an implicit part of entering the data itself (e.g., typing a single ASCII character at an asynchronous data terminal), or may be an explicit user action (e.g., typing a data forwarding signal such as Carriage Return at a buffered terminal). In the former case, block transfer begins on input of the data itself. In the latter case, block transfer begins on input of the authorization signal.

Successful Block Transfer occurs when the block has been transferred from the system to the destination user, with appropriate notification to that user when required. As in the case of authorization, delivery notification may be implicit or explicit. If the block is transferred across the destination user interface in a series of increments (e.g., the information field of a frame containing, perhaps, several hundred bits) followed by a delivery notification signal (e.g., "read complete"), the transfer may take considerable time. That time is counted in calculating Block Transfer Time.

Block Transfer Time is closely related to the parameter Message Transfer Time (MTT) in the ANSI performance standard for bit-oriented protocols (ANSI, 1980). Message Transfer Time is defined in that standard as follows:

> "MTT is the time in seconds that is required for a message to be transferred from a source frame buffer and accepted at the designated sink frame buffer. Where more than one link is involved in the transfer, it includes all of the time required for enroute storage and forwarding."

The same standard defines the term "message" as follows:

> "A message is an arbitrary amount of information whose beginning and end are defined. The information may be contained in one or more frames which must all be accepted (for the message to be accepted) in order to stop the MTT measurement."

This definition of "message" includes the ANS X3.102 definition of "block" as well as many other possible information units.

Message Transfer Time differs from Block Transfer Time in two respects. First, the Message Transfer Time definition establishes the start of an MTT measurement as follows:

> "MTT measurements start when both of the following have occurred: (a) transmission has been requested, and (b) the information field for the first frame has been entered in the source frame buffer. Transmission service requests may be evidenced by: the issuance of a call request; the transition to off-hook; an operator initiated action; or other equivalent action."

Thus defined, MTT includes Access Time as well as the time spent in retries if access is denied. Block Transfer Time excludes these access phase delays.

The second major difference between MTT and Block Transfer Time is in the choice of measurement interfaces. American National Standard X3.79 defines the end of an MTT measurement as follows:

> "MTT measurement is stopped upon acceptance of the final frame of the message at the destination frame buffer."

In a typical layered protocol hierarchy, frame acceptance occurs at the data link layer. As defined in ANS X3.102, the end of block transfer occurs when the block crosses the interface between application layer and the end user (e.g., terminal operator or user application program). The time difference between acceptance at the data link layer and delivery to the end user will often be small. However, it can be quite substantial if extensive processing, retransmissions, or time-sharing delays are involved.

Block Transfer Time also resembles a general transfer time parameter originally defined in the CCITT Green Book (CCITT, 1973):

> "Transfer Time - The time that elapses between the initial offering of a unit of the user's data to a network by a transmitting data terminal equipment and the complete delivery of that unit to a receiving data terminal equipment. . .A unit of data may be a bit, byte, packet, message, etc."

The significant difference here is the interface at which the starting and ending events take place.

In describing the access performance parameters, two distinct disadvantages of data communication delay were identified: loss of productive time and data aging. Both disadvantages apply in the case of block transfer, but with a slightly different emphasis. A user attempting to access a system must typically devote time to that effort for as long as it takes to succeed.

In contrast, a user who has transferred a block of information into the system is unoccupied as far as those data are concerned, and may well use the transfer time for other productive purposes. This is particularly true of "electronic mail" systems, where the preparation, input, and output time for a message may be negligible compared to its transfer time (e.g., minutes vs. hours).

Data aging is often more significant in the case of Block Transfer Time than in the case of Access Time. When a user is denied access to a data communication system, the user at least knows that the message is not on its way and can try again or take some alternative action. In contrast, the user may have no way of knowing when the transfer of the information is being delayed. Thus, the consequences may be more severe. Some modern networks resolve this situation by providing an explicit "delivery confirmation" response to the source user.

The possible values for Block Transfer Time range between zero and a practical upper bound defined by the "three times nominal" timeout defined earlier. A value of zero implies an infinite speed of transfer between source and destination. Extremely long values imply substantial opportunity for data aging.

As in the case of Access Time, user requirements for Block Transfer Time vary over a wide range. At the lower extreme are real-time process control and teleprocessing applications, where average one-way transfer times much less than a second are specified (Martin, 1976; DCA, 1975; Kelley, 1977; EPA, 1980). The upper extreme probably occurs in electronic message services, where next day delivery is usually quite acceptable.

Block Transfer Time can be roughly divided into three components: modulation time, propagation time, and storage time. Modulation time is the minimum time a signal element must be maintained at the input to a circuit in order to ensure its detection at the output of that circuit. It corresponds to the so-called "baud time" of a data modem and is inversely proportional to the signaling bandwidth.

Modulation time may actually determine the minimum Block Transfer Time on short, low-speed channels. As an example, a 20-mile cable circuit operating at 150 bits per second has a modulation time per bit over six times as long as the propagation time (Kimmett and Seitz, 1978).

Propagation time is the total time a signal takes to travel the physical distance between two ends of a transmission circuit. The shortest propagation times are provided by terrestrial (ground based) radiating systems (e.g., microwave), which combine high propagation velocity (about 186,000 miles per second) with relatively direct signal paths. Cable systems also provide relatively direct signal paths, but their transmission velocities are much lower (e.g., 20,000 miles per second or 50 microseconds per mile). Synchronous satellites exhibit much longer propagation times because of the much longer path distances involved (e.g., 250 milliseconds for a 45,000 mile, single-hop path).

Storage time includes all time during which a unit of user information is not physically moving across the physical media towards its destination. In all but the simplest systems, storage time is the dominant factor in determining Block Transfer Time. There are two principal reasons for temporarily storing user information within a system during its transfer between end users:

> Data Aggregation. Some systems collect many serially transmitted blocks together at each end of a transmission link to facilitate error detection and other control functions. Data may also be aggregated in a destination terminal to deliver it to the user in meaningful groups of blocks.
>
> Resource Sharing. Message and packet switched systems store user information at various internal switching nodes to increase utilization of the associated transmission links. Systems may also store user information to facilitate the sharing of user resources, as in the case of "mail box" and "call hold" features (AT&T, 1978).

Simple circuit-switched systems with unbuffered terminals have among the lowest Block Transfer Times available--in the range of 30 to 100 milliseconds for typical transmission path lengths. Connection-oriented systems with buffered terminals have somewhat longer Block Transfer Times, because the blocks are longer (e.g., 80 characters). Typical values for such systems are in the range of 100 to 300 milliseconds. The ARPANET, a prototype packet-switching network with virtual-circuit protocol, was designed to provide end-to-end delays less than one-half second for typical messages of a few thousand bits (Roberts and Wessler, 1970). Measured results indicate that actual transfer times in the ARPANET are in fact lower (Kleinrock, 1976; Wortendyke et al., 1982).

Block Transfer Times are substantially higher in traditional message switching systems because the messages are stored in their entirety at each switching node. The end-to-end message transfer times for DCA's AUTODIN I are probably typical (Armed Services Investigating Subcommittee, 1971):

Message Precedence	Transfer Time
Flash	< 10 minutes
Immediate	< 30 minutes
Priority	< 3 hours
Routine	< 6 hours

As noted earlier, the upper extreme on Block Transfer Time occurs in electronic mail systems. This is particularly true in the case where the destination user must take some action to read the mail. Delays on the order of a day or more are common. Note the need for the ancillary parameters to factor out the user component of such delays.

4.2.2 Bit/Block Error Probability

Bit Error Probability and Block Error Probability express the likelihood that a unit of information transferred from a source user to the intended destination user will be delivered with incorrect binary content. The numerator of each probability ratio is the total number of information units (bits or blocks) delivered to the intended destination user with content errors. The denominator is the total number of information units transferred between the source and destination of interest--that is, the sum of the correct and the incorrect blocks or bits.

In the case of Bit Error Probability, content error normally means that bits were inverted between the source and destination--that is, a transmitted one becomes a zero or vice versa. The X3.102 standard also considers several more complex cases, and offers the following data reduction guidelines.

1. In the case of code conversion, error comparisons should be based on the intended and actual bit patterns at the destination user interface.

2. In the case where information crosses the user/system interfaces in the form of nonbinary symbols (e.g., graphic symbols), the input or output symbols should be translated into bits on the basis of the binary representation physically closest to the user.

In the case of Block Error Probability, content error in a delivered block is defined to exist whenever:

1. One or more bits (or even all of the bits) in the block are incorrect; or

2. Some, but not all, of the bits in the block are lost or extra. (If all of the bits are lost or all of the bits are extra, then the block is a Lost Block or an Extra Block.)

In general, the Block Error Probability for an n-bit block will be between one and n times the sum of the Bit Error, Bit Loss, and Extra Bit Probabilities, depending on how many failure outcomes occur in each block.

Bit and Block Error Probability are among the most widely used data communication performance parameters, so there is little need to relate them to other parameters for the purpose of familiarization. However, it must be emphasized that both parameters apply to end-to-end services as defined earlier. Their values thus reflect the error-producing or error-removing effects of data terminals and higher level protocols.

Bit Error Probability is similar to the Residual Error Rate (RER) parameter defined in ANS X3.79 in that both parameters measure errors that remain after error control. However, the latter parameter includes lost, extra, and misdelivered bits in the numerator, and uses the total number of transmitted (source user) bits as the denominator. This makes it theoretically possible for RER to exceed one.

The significance of Bit and Block Error Probability to end users is also relatively obvious, but a brief discussion may be helpful. Two general categories of error effects can be distinguished, depending on whether the end user does or does not detect the error prior to using the delivered information. User detection of errors in the delivered information is most likely in the case where the user is a human terminal operator.

If the error is isolated and occurs in text that contains much redundancy (e.g., misprinting a single character in English text), the operator can usually infer the meaning and the error may be no more than a minor nuisance. If the error is more extensive or occurs in nonredundant text (e.g., garbling of an entire line of text or an error in numerical data), then the impact on the user will be much more significant. In the latter case, the destination user must typically re-contact the source user, request a retransmission, and defer any action based on that information until the retransmission is received. In essence, the users are performing the system function of error control in a costly and inefficient manner.

The effects of delivered errors are generally more serious when they are not detected by the destination user prior to actual use of the delivered information. This will almost always be the case where no human operator is involved. The many possible effects of undetected errors can be summarized by saying that they cause the destination user to make decisions and take actions based on erroneous information. In the case of applications such as electronic funds transfer, such mistakes can be very costly.

Bit and Block Error Probability values vary between zero and one, with a practical upper limit of 0.5 on the former. In each case, a value of zero implies that incorrect information is never delivered to the end users. A Bit Error Probability of 0.5 means that any delivered bit is just as likely to be wrong as right; therefore, no useful information can be communicated. A Block Error Probability value of one indicates that every delivered block contains at least one incorrect, lost, or extra bit.

User requirements for Bit and Block Error Probability depend, as one would expect, on the consequences of errors. Narrative message applications are among the least stringent, because their high inherent redundancy makes user correction possible. It has been estimated, for example, that normal English text is 50 percent redundant compared to a random character sequence (Shannon, 1948). Very high Bit Error Probability values may be tolerated at the output of digital subsystems used in transmitting voice. It has been shown, for example, that Continuous Variable Slope Delta systems can produce acceptable speech with channel Bit Error Probabilities approaching 1 in 10 (McRae et al., 1976).

As suggested earlier, user requirements for Bit and Block Error Probability are most stringent in applications where the cost of errors is high. A Bit Error Probability of 10^{-12} has been specified for military packet switching users having error controlled access circuits (DCA, 1975). A more recent study suggested a less stringent, and probably more realistic, value of 10^{-10} for a similar application (Nesenbergs et al., 1980). Bit Error Probability requirements for normal teleprocessing applications are in the range of 10^{-5} to 10^{-8}. A value of 8×10^{-6} is specified in EPA (1980).

Some feeling for the significance of these numbers can be obtained by relating them to output rate. A 10^{-5} Bit Error Probability corresponds to approximately one bit error every 17 minutes at 100 bits per second; every two minutes at 1000 bits per second; and every 10 seconds at 10 kilobits per second. A 10^{-10} Bit Error Probability corresponds to one bit error every 32 years, every 3 years, or every 4 months, respectively, at the same output rates.

In describing the Bit and Block Error Probabilities of existing systems, it is important to distinguish between values observed at the transmission channel interfaces and at end user interfaces. So-called "raw channel" Bit

Error Probabilities vary from 10^{-3} (for high frequency radio systems) to better than 10^{-7} (for all digital, nonradiating, local area networks). A value of 10^{-5} is probably typical for the public switched telephone network (AT&T, 1971). For any given transmission speed, the raw channel error probability is primarily determined by two factors:

1. The signal-to-noise ratio at the receiver input.
2. The effective transmission bandwidth.

These two factors can be effectively traded off in many cases (Utlaut, 1978).

The raw channel error performance of a data communication system can be vastly improved through the use of érror control techniques (Hamming, 1950; Kuhn, 1963). The most commonly used technique today is simple error detection and retransmission (often called ARQ). Well-designed ARQ systems can produce output channel Bit Error Probabilities in the range of 10^{-8} to 10^{-10} with negligible coding redundancy, almost regardless of the raw channel error probability. No bit errors were observed in over 3 x 10^6 transmitted bits in the ARPANET experiment (Wortendyke et al., 1982).

While ARQ systems may reduce error probability dramatically, they may also severely restrict throughput as the channel error probability approaches the reciprocal of the block size. That is, when the error rate is such that each block is almost certain to contain at least one error, most of the transmitted blocks are retransmissions. This disadvantage can be reduced in many cases by hybrid ARQ and forward error correction systems (Nesenbergs, 1975).

4.2.3 Bit/Block Misdelivery Probability

Bit and Block Misdelivery Probability specify the proportion of bits and blocks that were transferred from source user A to destination user B, but were actually intended for some destination user other than B. The numerator of each probability ratio is the total number of misdelivered information units (bits or blocks); and the denominator is the total number of information units transferred between the specified source and destination users. Expressing misdelivery probability on a bit basis is not intended to imply that individual bits will be misdelivered. Such outcomes will normally occur in groups of one or more blocks.

How can misdelivery occur? One obvious cause in connection-oriented systems is Incorrect Access--that is, a source user is connected to the wrong destination user during the access phase. Misdelivery can also occur in connectionless systems, as a result of routing errors within the system. Misrouting of a message can either be a random event (e.g., caused by an undetected error in a message address field), or a systematic occurrence (e.g., caused by an incorrect address table in a message switching center). Errors of the latter type may be a result of software errors, hardware failures, operator errors, or even deliberate tampering.

The significance of misdelivery to the source user has been discussed earlier in connection with Incorrect Access Probability. Briefly, the three chief risks are:

1. Exploitation of the misdelivered information by a dishonest recipient;

2. Inappropriate actions by the source user based on the false assumption that the information has been successfully delivered; and

3. Inappropriate actions by the unintended destination user based on the false assumption that the information was intended for that user.

Possible Bit and Block Misdelivery Probability values range between zero and one. A value of zero implies that misdelivery does not occur. A value of one implies an addressing error in which all transmitted traffic is systematically misrouted to an unintended destination user.

One published requirements specification (DCA, 1975) calls for a "segment" misdelivery probability of 10^{-11}. This number applies directly to both Bit and Block Misdelivery Probability, since misdelivery outcomes normally occur in block or "segment" groups. More recently, Nesenbergs et al. (1980) suggest the same target value. For comparison, such a value is sufficient to enable a user pair to exchange 10 million packets per day for 27 years before the first misdelivery occurs.

Misdelivery Probabilities of this magnitude are impossible to measure due to their infrequency. Thus they are, in a sense, academic. However, they can have value as an influence on system design. They may also be included in the specification for a service, but the value of this is questionable in the absence of a means of measuring them.

Misdelivery probability can be reduced to negligible proportions by at least one brute force approach: that of providing a protected, dedicated, hard-wired line between the source user and the destination user. Such an approach would only be justified in situations where the consequences of misdelivery are very grave.

Another approach is the use of encryption techniques. While encryption techniques cannot prevent misrouting, a simple message routing indicator scheme may be combined with encryption of the routing indicator to detect misrouting. Encryption can also be used to prevent the unintended destination user from being able to use the information (FIPS, 1981; Feistel, 1973; Popek, 1974).

The single design feature that most strongly influences Bit and Block Misdelivery Probability in switched systems is the error control technique. Depending on the number of compatible terminals in a network, systems without error control on the addressing information could experience misdelivery probabilities in excess of 10^{-5} (Kimmett and Seitz, 1978). Error control provisions such as those employed in common channel signaling systems and in the ARPA network will reduce these probabilities substantially, perhaps to the range of 10^{-9} in a benign environment (without deliberate tampering). CCITT Study Group VII suggests an "illustrative figure" for datagram misdelivery probability of 10^{-6} (CCITT, 1978). As one would expect, these values are greatly increased by the presence of deliberate tampering. Encryption techniques are routinely used to foil such attempts.

The cause and effect relationship between Incorrect Access and Bit or Block Misdelivery has been discussed earlier. As a general rule, it can be estimated that the values for these two parameter types will seldom differ by more than an order of magnitude in connection-oriented systems. The latter values may be somewhat lower as a result of user detection of Incorrect Access events prior to the completion of user information transfer.

4.2.4 Bit/Block Loss Probability

Bit Loss Probability and Block Loss Probability express the likelihood that a system will fail to deliver a unit of information output by a source user to the intended destination user within the specified maximum transfer time. The numerator of each probability ratio is the total number of information units (bits or blocks) lost as a result of system performance failures. The denominator is the total number of information units output by the source, excluding any not delivered as a result of user nonperformance. The timeout period on both bit and block transfer attempts is three times the nominal (specified) value of the parameter Block Transfer Time. Block Loss is distinguished from Block Refusal (that is, nondelivery for which the user is responsible) by comparison of ancillary parameter values, as described in Section 4.4.

How can a system lose a user's information? There are at least seven distinct ways. The first is through signal interruption. In systems that do not have internal block storage and retransmission capability, a "fade" or other attenuation may interrupt the signal representing a transmitted sequence of bits. The result is that these bits are simply lost from the delivered data stream. This is a common phenomenon in asynchronous systems. As an example, the character loss rate often exceeds the character error rate by more than an order of magnitude on short, low speed data links using the public switched telephone network (AT&T, 1971).

The second possible cause of data loss is timing errors. Whenever two communicating elements in a digital system are driven by different clocks, there is the possibility that one or more bits will not be sampled by the receiver while they are being presented by the sender. Unless the "slip" is later corrected by error detection and retransmission, the unsampled bits will be lost by the system.

A third possible cause of data loss is ARQ protocol failures. Simple ARQ protocols control retransmission of blocks containing errors by means of positive acknowledgement (ACK) and negative acknowledgement (NAK) signals. An ACK or NAK signal is returned from the receiver to the sender after each block transmission. A NAK will cause the sender to retransmit the block to the receiver. Block loss will occur whenever a NAK is changed (due to transmission errors) to an ACK, since the sender will assume that the block was delivered when, in fact, it was lost. The probability of such errors can be made extremely small by appropriate error coding, but not all systems use such methods.

A fourth possible cause of data loss is hardware or software "crashes" in system switching computers. A simple illustration of a hardware crash is the case where the power supplying a semiconductor memory in a message switching computer is unexpectedly interrupted. The system will then lose all messages stored in that memory unless backup provisions have been made. Software crashes are even more serious, since the affected switch may continue to operate in an unpredictable manner for some time before the failure is detected. Sunshine (1975) proves that it is impossible to prevent data loss or duplication in all cases when one side of a protocol fails with memory loss.

A possible fifth cause of data loss is a flow control failure known as a lockup. Kleinrock (1976) describes one of many lockup conditions actually observed in the ARPANET:

> "Reassembly lockup, the most famous of the ARPANET deadlock conditions, was due to a logical flaw in the original flow control procedure. It occurred when partially reassembled messages could not be completely reassembled since the congested network prevented their remaining packets from reaching the destination; that is, each of the destination's neighbors had given all of their store-and-forward buffers to additional messages heading for that same destination (at which no unassigned reassembly buffers were available). Thus the urgently needed remaining packets could not pass through the barrier of, blocked IMPs surrounding the destination."

While noting that this and several other lockup conditions have been eliminated by subsequent changes in flow control procedures, Kleinrock points out that "indeed, whenever one introduces conditions on the message flow, there exists the danger that these conditions cannot be met and then the message flow will cease. Reassembly and sequencing are examples of such conditions."

A sixth possible cause of data loss is the deliberate discarding of packets to eliminate network congestion. This strategy is the rule rather than the exception in modern packet switching networks. CCITT (1978) suggests an "illustrative figure" for the probability of such a discard (with notification to the sender) of 10^{-3}. The suggested value for discard without notification is 10^{-5}. Kleinrock et al. (1976) present measured results indicating that, on the average, one message in a hundred that enters the ARPANET does not reach its destination. The stated reason for this undesirable behavior is that many destination host computers are tardy in accepting messages. Note that the host computer front end is part of the system from the end user point of view. Wortendyke et al. (1982) report a measured ARPANET end-to-end Block Loss Probability (excluding timeouts) of 2×10^{-3}. The corresponding value including timeouts was 6×10^{-3}.

A seventh possible cause of data loss is internal misrouting. Data that are misrouted in a data communication system may or may not be delivered to an incorrect destination user. But, in either case, the data are lost as far as the source user/destination user pair are concerned.

The impact of data loss on users has been discussed earlier, since data loss and excessive delay have similar effects. The loss of a few early bits in a block may cause the user to misinterpret the meaning of succeeding bits in some applications. The effect on the user may then be the same as if the system's Bit Error Probability had suddenly jumped to 0.5.

Bit and Block Loss Probability values range between zero and one. A value of zero means no loss whatever and a value of one suggests a system that is an open circuit delivering no information to the destination user. Perceived user requirements for Bit Loss Probability range from values as high as 10^{-3} for redundant message text to values as low as 10^{-11} in highly critical applications. In character-asynchronous systems, the former value corresponds to about two errors on a printed page of text. Teleprocessing user requirements are typically intermediate between these extremes, in the range of 10^{-5} to 10^{-8}. A Block (character) Loss Probability of 8×10^{-6} is specified in EPA (1980).

As in the case of the error probabilities, the key design impact on the loss probabilities is the choice of error control technique within the system. Systems that do not provide data storage and retransmission are always vulnerable to signal interruptions and slips. There are special situations, such as space-to-earth communications, in which retransmission protocols are not possible. Character loss probabilities for unbuffered, asynchronous systems in the public switched network are in the range of 10^{-3} to 10^{-5} (AT&T, 1971).

In most modern systems with well designed retransmission protocols, the predominant cause of data loss will be switch crashes and network congestion. Given Kleinrock's measurements of a 1 to 2 percent down rate for the ARPANET IMPs, it seems questionable that Bit Loss Probability requirements like 10^{-11} are attainable.

4.2.5 Extra Bit/Block Probability

Extra Bit Probability and Extra Block Probability express the likelihood that the information delivered to a destination user will contain duplicate bits or blocks, or other extra information not output by the source user. The numerator of each probability ratio is the total number of extra information units (bits or blocks) received by a particular destination user. The denominator is the total number of information units received by that user. Unless Misdelivered Bits are explicitly identified, they will be counted as Extra Bits (Seitz and McManamon, 1978).

How can a system include extra information in a sequence of bits delivered to a destination user? The most frequent cause is the inadvertent duplication of previously delivered data. Three of the seven phenomena discussed in the previous section on data loss can also cause data duplication: timing errors, ARQ protocol failures, and hardware or software crashes.

Timing errors between subsystems cause duplication rather than loss whenever the clock in the sending subsystem is slower than that in the receiving subsystem. In such a situation, input data will occasionally be sampled twice. If the error is not corrected later by error detection and retransmission, the duplicate data will be delivered to the destination user.

ARQ protocol failures cause duplication in essentially the complement of the way that they cause data loss: whenever an ACK is changed to a NAK as a result of transmission errors, the data sender in that part of the system will unnecessarily retransmit the block. Thus, two copies of the block will then exist at the receiver. Both copies may be delivered to the destination user if the protocol used does not assign a unique identification to each packet. The same events may occur when an ACK is lost in positive acknowledgement, retransmission on timeout (PAR) protocols.

Probably the dominant cause of data duplication in modern data communication systems is hardware or software crashes. When a switch crashes, its memory about the current status of information in transit may well be lost. Most switches are programmed to retransmit dubious blocks in such a circumstance. Some of these blocks may have previously been delivered.

Some data communication networks deliberately transmit duplicate copies of user information to improve transfer reliability or speed, and then eliminate the duplicates by filtering at the receiver. One modern network that uses this technique is the National Weather Service's Automation of Field Operations and Services (AFOS) network. In the case of AFOS, a node on the ring sends a block in each direction as a means of ensuring transmission even if the ring should be broken in one direction. At the destination, the duplicate is destroyed. However, an error in the header information may cause the receiving node to assume that two unique blocks are involved, thereby resulting in duplicate blocks.

The impact of data duplication on the user is substantially different from data loss. Extra information has no impact at all on the source user, since his entire output is delivered as intended. The impact of extra information on the destination user depends on the type of user and on how clearly the duplicated information is delimited from other, nonduplicate information. A clearly defined, complete duplicate of a previously delivered message is normally no more than a minor nuisance to a human end user. He will simply throw it out. At the other extreme, duplication of even a few bits of numerical data may cause a computer application program to completely misinterpret an input file, thereby producing a meaningless or misleading output.

Extra Bit and Extra Block Probability values theoretically range from between zero and one. However, 0.5 is probably a more realistic upper bound. A value of 0.5 suggests that every block output by the source user is delivered to the destination user twice.

Data on user requirements for Extra Bit and Extra Block Probability is very scarce. Nesenbergs et al. (1980) suggest an Extra Bit Probability value of from 10^{-10} to 10^{-11} for interactive data communication services in the future DCS. This value is based on the premise that Extra Bits have essentially the same effect as Incorrect Bits. EPA (1980) specifies an Extra Character Probability of 8 x 10^{-6} for the teleprocessing application cited earlier, apparently based on the same premise.

The key design impact on Extra Bit and Extra Block Probability is the choice of error control technique. Character asynchronous circuit-switched systems with no retransmission or buffering will provide the lowest values (essentially zero), since such systems contain no storage in which duplicate information could be created. Traditional message-switched systems probably exhibit the highest values (e.g., as high 10^{-3}) because of their long-term storage of entire user messages in each switching node. Kimmett and Seitz (1978) estimate a value of about 10^{-6} for a star-connected message-switching network with modern outage recovery features.

4.2.6 User Information Bit Transfer Rate

User Information Bit Transfer Rate describes the flow of user information through a data communication system. It is the slower of two rates--the rate at which user information is transferred from a source user to the system, and the rate at which the same user information is transferred from the system to the destination user.

Stated more formally, User Information Bit Transfer Rate is the total number of Successful Bit Transfer outcomes in an individual transfer sample divided by the input/output time for that sample. The Successful Bit Transfer outcome occurs when a bit is transferred from the source user to the intended destination user without error within the maximum transfer time for the associated block. The input/output time for a transfer sample is the larger of the input time or the output time for that sample (Figure 18).

The sample input time begins when input of the sample begins, and ends when either: (1) all bits in the sample have been input to the system, and the system authorized to deliver them; or (2) sample input/output timeout occurs. The sample output time begins when the first user information bit in the sample is delivered by the system to the destination user. It ends when either (1) the last bit of user information in the sample is delivered to the destination user; or (2) sample input/output timeout occurs.

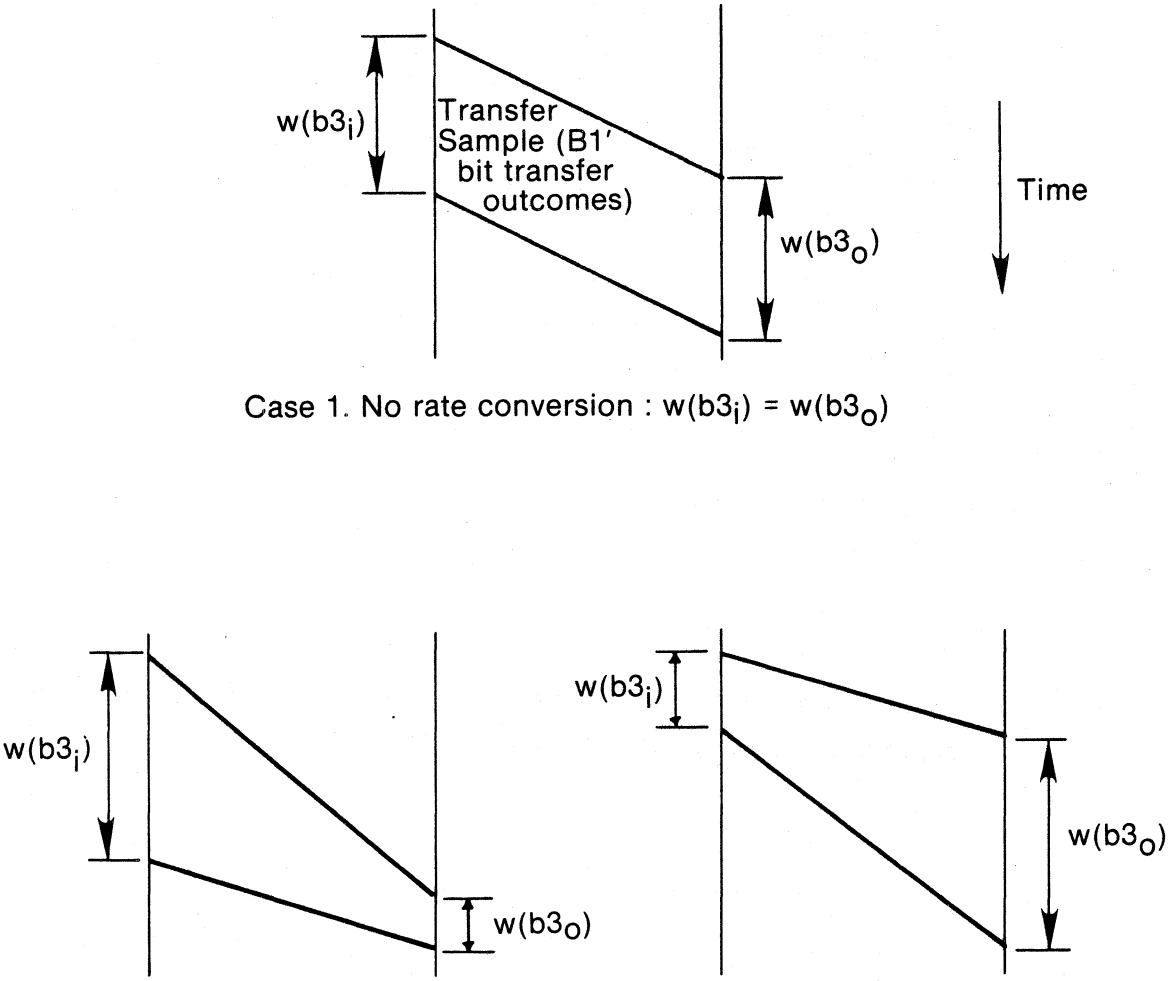
A sample input/output timeout occurs whenever the duration of an individual sample input or output period exceeds three times the specified average input/output time. Transfer samples that time out are not included in calculating User Information Bit Transfer Rate.

As implied in the definition of User Information Bit Transfer Rate, the flow of user information bits into and out of a system may be quite different. An extreme example is a computer application program supplying text to a remote low-speed printer. Even in communications between terminals, the input and output rates may differ due to differences in the terminal speed settings or other characteristics. Data input and/or output may also be "bursty," with instantaneous rates far higher than would be sustainable on a continuous basis.

Using the larger of the sample input time or sample output time in defining User Information Bit Transfer Rate ensures that specified transfer rates will be based on the slower interface (the bottleneck).

A brief survey of previously defined flow measures will further clarify the meaning of the X3.102 rate parameter. ANSI (1974, 1980) defines four earlier transfer rate parameters: Transfer Rate of Information Bits (TRIB), Link Transfer Rate of Information Bits (L-TRIB), Network Transfer Rate of Information Bits (N-TRIB), and User Message Data Rate (UMDR). Key excerpts from these parameter definitions are provided here:

> Transfer Rate of Information Bits. The TRIB criterion expresses the ratio of the number of Information Bits accepted by the receiving Terminal Configuration during a single Information Transfer Phase (Phase 3) to the duration of that Information Transfer Phase. TRIB is expressed in bits



Case 1. No rate conversion : $w(b3_i) = w(b3_o)$

Case 2. Rate increase: $w(b3_i) > w(b3_o)$

Case 3. Rate reduction : $w(b3_i) < w(b3_o)$

$$\text{User Information Bit Transfer Rate } R(b1_s) = \frac{B1_s}{\text{Max}\left[w(b3_i) \text{ or } w(b3_o)\right]}$$

$B1_s$ = Total Successful Bit Transfer outcomes in the transfer sample.

Figure 18. User information bit transfer rate.

per second. Information Bits are all bits contained in Information Characters except start-stop elements (if used) and parity bits. Information Bits are defined to be accepted by the receiving Terminal Configuration if a positive acknowledgement to a transmission block is received by the sending Terminal Configuration.

Link Transfer Rate of Information Bits. L-TRIB is the number of information bits transferred and accepted in a data communication link during a specified time interval divided by that time interval; it is expressed in bits per second. In a Primary to Secondary, point-to-point or multipoint configuration, the number of information bits used in determining L-TRIB is the sum of the information bits transmitted and received by the Primary. In a balanced configuration, it is the sum of the information bits transmitted and received by either station. Information bits in frames that are not accepted are excluded.

Network Transfer Rate of Information Bits. N-TRIB is a measure of the total information flow in a network. It is determined by dividing the sum of all accepted information bits leaving all exit ports of the network by a continuous time interval measurement.

User Message Data Rate. UMDR, a message-based measure of performance, is determined by dividing the number of user-defined data bits in a message by the Message Transfer Time. A description of Message Transfer Time has been provided in Section 4.2.1.

Of these four earlier rate parameters, TRIB is perhaps the most similar to User Information Bit Transfer Rate. TRIB differs from User Information Bit Transfer Rate in two ways. First, TRIB defines "Information Bits" and "Information Transfer Phase" on the basis of a particular, character-oriented communication control protocol, ANS X3.28 (ANSI, 1971). Second, TRIB includes network transit delay (e.g., Block Transfer Time) in its denominator. User Information Bit Transfer Rate deliberately excludes this delay to provide a more accurate characterization of store and forward systems.

The parameter L-TRIB differs from User Information Bit Transfer Rate in two ways. First, L-TRIB is measured at the frame buffer outputs (i.e., at Layer 2 in a typical layered reference model) rather than at the end user interfaces (above Layer 7). Second, L-TRIB is a bidirectional flow measure, since both transmitted and received bits are counted in the numerator. Bidirectional values for User Information Bit Transfer Rate may be obtained by combining or averaging the values characterizing each direction of flow.

The parameter N-TRIB differs from User Information Bit Transfer Rate (as well as the other ANS X3.79 parameters) in that it measures the total network throughput for all users rather than a particular user pair or group. Like L-TRIB, N-TRIB is measured at the frame buffer outputs (Layer 2) rather than at the end user interfaces.

The parameter UMDR differs from User Information Bit Transfer Rate in two respects. First, it is measured at the frame buffer (Layer 2). Second, it includes Access Time (and any time required to respond to Access Denials) in the denominator.

User Information Bit Transfer Rate is significant to users because it strongly affects the efficiency of user functions. A low system input or output rate wastes user time, and can be a substantial source of frustration to human users. An excessively high output rate may overload the destination user.

Values for User Information Bit Transfer Rate range between zero and a practical upper limit determined by the signaling rate of the service. Low values imply little useful flow. That is, either (1) the input rate is negligible, (2) the output rate is negligible, or (3) the information delivered is incorrect. Conversely, high values imply high input rates, high output rates, and good delivered accuracy.

In specifying user requirements for User Information Bit Transfer Rate, it should be remembered that the user is often the primary cause of flow restrictions in data communication systems. It makes little sense, for example, to specify a service with a User Information Bit Transfer Rate of 2000 bits per second if the input rate is limited by the source user's typing rate to 35 bits per second (the input rate of a reasonably proficient typist). Similar mismatches can occur, in general, on the output side. The important point is that advances in resource-sharing technology make it increasingly feasible and economical for users to specify transfer rate requirements on the basis of their actual capability to generate and absorb information. These capabilities are often far lower than the data transfer capabilities of traditional dedicated and switched services.

The User Information Bit Transfer Rate parameter encourages user-sensitive rate specifications by including user delays (e.g., think time and typing delays) in the parameter's definition. As noted earlier, the equivalent user-independent values can always be determined using the ancillary parameters (Section 4.4). In general, user requirements for User Information Bit Transfer Rate should be derived in the context of a data stream model like that described in Jackson and Stubbs (1969).

Typical user-to-user transfer rate requirements for interactive teleprocessing services may be inferred from Table 2 below (Grubb and Cotton, 1975). References 5, 6, and 7 in Table 2 are referenced in this report as

Schwartz et al., (1972), Jackson and Stubbs (1969), and Fuchs and Jackson (1979), respectively. Note that the term "system" in Table 2 refers to the teleprocessing computer, including the application program.

Table 2 Typical Transfer Rate Requirements for Interactive Teleprocessing

	Speed in Bits per Second			
	Signaling Speed	User	System	Average of Both
Tymnet (ref.5)	110.0	3.5	35.0	-
GE Information Services (ref. 5)	110.0	-	49.0	-
	300.0	-	147.0	-
Jackson, Stubbs and Fuchs of Bell Telephone Labs (ref. 6 & 7)				
moderately loaded scientific	110.0	3.4	61.6	25.2
heavily loaded scientific	110.0	1.9	14.7	10.7
moderately loaded business	150.0	5.6	58.1	28.0

The effects of user delay on throughput are evident in the above data. As Grubb and Cotton point out, the strong asymmetry of the operator-to-computer and computer-to-operator paths suggest that separate rate specifications for the two paths may be appropriate.

Transfer rate requirements for operator-to-operator applications are normally somewhat higher than the operator-to-computer values cited above, because a more relaxed communication format and less think time are involved. Values in the range of 10 to 20 bits per second are typical of operator-to-operator transactions when listening time is included. Corresponding continuous transmission values would be about twice as high.

Circuit switched and dedicated data communication services have traditionally been specified in terms of the signalling rate at the physical data terminal equipment/data circuit-terminating equipment (DTE/DCE) interface rather than in terms of user-to-user throughput. Commonly specified modem signalling rates are 300, 1200, 2400, 4800, and 9600 bits per second. These rates assume continuous input and output, include overhead bits not delivered to the end user, and make no provision for retransmissions. User Information Bit Transfer Rates observed at the end user interfaces are commonly 20 to 50% lower.

Martin (1976) lists some 20 widely available data communication services with signaling rates ranging from 45 bits per second (switched, sub-voice grade channels) to 500,000 bits per second (dedicated wideband channels). Among the switched services, the highest signaling rates commonly available are 56 and 64 kilobits per second. Typically, the dedicated wideband services are used for interconnecting switches, concentrators, and other system components rather than individual end users.

Since User Information Bit Transfer Rate describes the capacity needed by the users, its accurate specification assists providers in network planning and resource sharing. Packet-switching networks exploit the intermittency of user data input by time-multiplexing the communications of many user pairs on a single trunk. The sharing of transmission capacity improves trunk utilization and thus provides a more cost-effective service. The measurements reported in Wortendyke et al. (1982) indicate that the ARPANET can support about a 5 kbps throughput between host computer application programs via its 50 kbps trunks. Most public packet-switching networks are designed to support operator-to-program communications and offer substantially lower User Information Bit Transfer Rates.

4.2.7 Transfer Denial Probability

Transfer Denial Probability expresses the likelihood of an unacceptable degradation in the performance of a data communication service during user information transfer. This degradation may be in the form of unacceptably poor transmission quality or an unacceptably low throughput. Complete disconnection of communicating users reduces the throughput to zero and is thus included as a limiting case.

Stated more formally, Transfer Denial Probability is the ratio of total Transfer Denials to total transfer samples counted during a performance measurement. Transfer Denial is defined to occur whenever the performance observed during a transfer sample is worse than the threshold of acceptability for any one of four supported user information transfer parameters: Bit Error Probability, Bit Loss Probability, Extra Bit Probability, and User Information Bit Transfer Rate.

A transfer sample is a randomly selected observation of user information transfer performance between a specified source and destination user. It begins at the start of input of a selected user information block at the source user interface, and continues until a specified number of consecutive user information bit transfer outcomes has been determined.

The size of the transfer sample must be large enough to provide estimates of reasonable precision for the four supported parameters, without being so large that brief degradations are masked. To accomplish these conflicting goals, ANS X3.102 defines the size of a transfer sample indirectly, by stipulating that it must be sufficient to provide threshold estimates for the four supported parameters each having at least a relative precision of 50% at a 95% confidence level. In most cases, a transfer sample of a few thousand bits (distributed in several blocks) will suffice. Long-term parameter values are normally determined with much higher precision using a larger sample size. The subject of parameter estimation is discussed in Crow and Miles (1977), Crow (1978), and Crow (1979).

The Transfer Denial threshold for User Information Bit Transfer Rate is defined as one-third of the specified User Information Bit Transfer Rate. The Transfer Denial threshold for each of the three bit transfer failure probabilities is defined as the fourth root of the parameter's specified value.[6] Transfer samples are discarded if the source user intentionally disengages before the complete sample is input to the system, or if delivery of the sample to the destination user is not completed as a result of nonperformance on the part of either user.

Transfer Denial Probability is closely related to the widely used concept of reliability. Reliability is defined by the Advisory Group on Reliability of Electronic Equipment (AGREE, 1957) as follows:

> "Reliability is the probability of performing without failure a specified function under given conditions for a specified period of time."

Transfer Denial Probability is essentially a "specialized complement" of reliability as just defined. It is the probability that a specified function (the transfer of user information) will not be performed successfully (with all supported parameter values better than their thresholds) under given conditions (particular source and destination user interfaces and events) for a specified period of time (the sample transfer interval). It expresses the likelihood that a system will be unable to provide satisfactory performance in transferring a given number of user information bits.

[6] As an example, the Transfer Denial threshold corresponding to a specified Bit Error Probability of 10^{-8} is 10^{-2}.

Possible values for Transfer Denial Probability range between zero and one. A value of zero indicates that every transfer sample has an acceptable level of performance, and implies that the transfer performance experienced by the user will probably be continuously acceptable as well. A value of one indicates that no transfer sample has an acceptable level of performance, and implies that the performance experienced by the user will also be continuously unacceptable.

There is relatively little data on user requirements for Transfer Denial Probability. Based on inference from specified values for availability, it appears that user requirements for Transfer Denial Probability may range from 10^{-2} to about 10^{-5}. EPA (1980) specifies a "Probability of Outage" no greater than 5×10^{-5} for a typical remote access data processing application. The corresponding value for Transfer Denial Probability would be somewhat higher due to the fact that less stringent failure thresholds are used in its definition.

Very low values for Transfer Denial Probability can impose severe requirements on the system design. At the extreme, it may be necessary to provide duplicate communication lines to the user, duplexed node computers, and other very costly system design features to achieve unusually low values for Transfer Denial Probability. The IMP/TIP outage data reported in Kleinrock (1976) suggests that the local switches may be the "weak link" in many networks.

4.3 Disengagement Parameters

Most people have experienced the frustration of attempting to disengage from a system after receiving the requested service, only to find that the system delays disengagement excessively or "won't let go." This occasionally occurs in the public switched telephone network where a user is unable to immediately place a new call because he is still "connected" to the previous called party. It can also occur in data communication systems--for example, when the system loses a close request packet. In such cases, Successful Disengagement cannot occur until a second disengagement attempt is made.

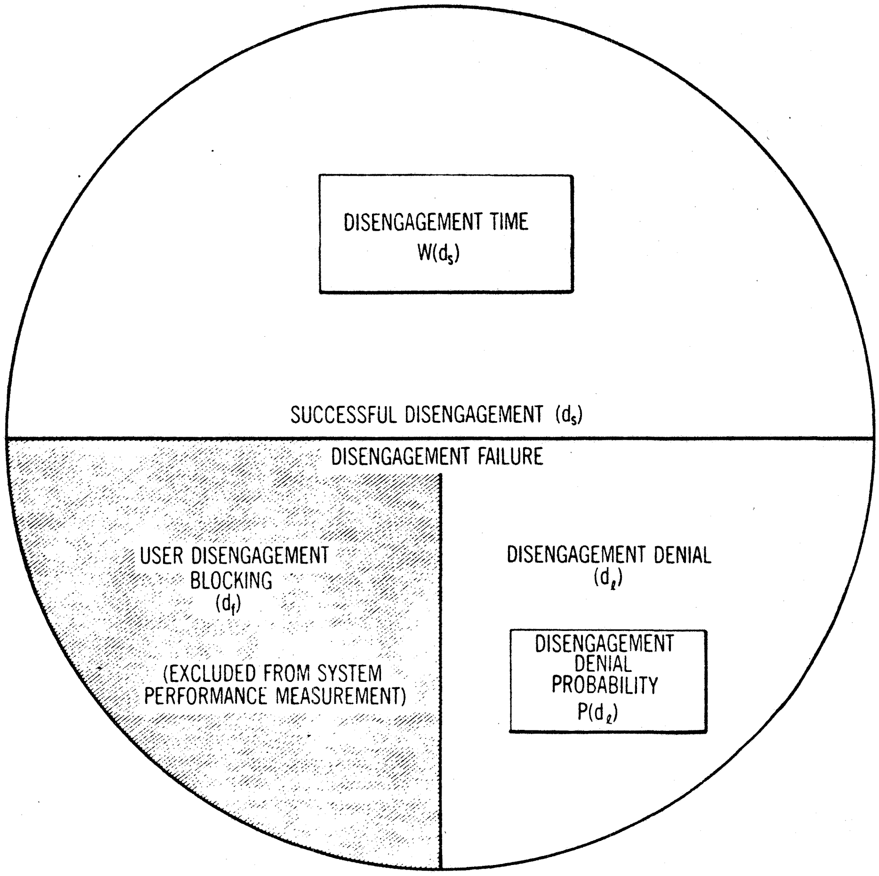
User concerns with disengagement performance tend to fall in two categories: (1) how long disengagement will take (speed); and (2) the likelihood of disengagement failure as a result of error or nonperformance on the part of the system (accuracy, reliability). The standard provides two disengagement parameters that express these user concerns: Disengagement Time and Disengagement Denial Probability. These are shown in Figure 19.

4.3.1 Disengagement Time

Disengagement Time is the average time a user must wait, after requesting disengagement from a data communication session, for the system to successfully accomplish the disengagement function. As noted in Section 3.2.2, a separate disengagement function is defined for each user participating in a data communication session. For each user, computation of Disengagement Time begins on issuance of a "disengagement request" signal and ends either (1) when the user receives the "disengagement confirmation" signal, in systems that provide such a signal; or (2) when the user is next able to initiate a new access attempt, in systems that do not. Depending on system characteristics, the disengagement request for an end user may be issued by that end user or by another participant in the session. Disengagement Times are normally measured separately for each user, but may be specified with a single (worse-case) value when variations between users are not considered significant.

Examples of specific disengagement request and disengagement confirmation signals have been cited earlier (Section 3.2.2). Identifying these signals in particular systems is normally straightforward, but two particular cases are of note. The first is the case of two-point connection-oriented sessions. In such sessions, disengaging one user necessarily implies disengaging the other, since a connection with one end has no meaning. Both disengagement functions therefore start with a single disengagement request.

The second case of note is that of preemption. In some systems (e.g., AUTOVON), ongoing communications may be terminated by the system in order to free resources needed to support higher priority users. Although such events could be treated as system-initiated disengagements, it is more consistent with the service concept (and the perception of the interrupted users) to treat them as Transfer Denials. This is particularly true in cases where the system does not notify low priority users of impending disconnection. As in the case of access requests, disengagement requests may be explicit or implicit. An example of an implicit disengagement request would be the case where the system initiates disengagement after a fixed number of blocks have been transferred.


DISENGAGEMENT TIME
$W(d_s)$
SUCCESSFUL DISENGAGEMENT (d_s)
DISENGAGEMENT FAILURE
USER DISENGAGEMENT BLOCKING
(d_f)
(EXCLUDED FROM SYSTEM PERFORMANCE MEASUREMENT)
DISENGAGEMENT DENIAL
(d_ℓ)
DISENGAGEMENT DENIAL PROBABILITY
$P(d_\ell)$

DISENGAGEMENT PARAMETERS

1. Disengagement Time $= W(d_S) = \frac{1}{D_S} \sum_{d_S=1}^{D_S} w(d_S)$
2. Disengagement Denial Probability $= P(d_\ell) = D_\ell / D'$

DEFINITIONS

D' = Total number of disengagement attempts counted during a disengagement parameter measurement: $D_s + D_\ell$.

D_S = Total number of Successful Disengagement outcomes counted during a disengagement parameter measurement.

D_ℓ = Total number of Disengagement Denials counted during a disengagement parameter measurement.

$t(d)$ = Time a particular disengagement attempt starts.

$t(d_S)$ = Time Successful Disengagement is attained on a particular disengagement attempt.

$w(d_S)$ = Value of disengagement time measured on a particular successful disengagement attempt: $t(d_S)$-$t(d)$

Figure 19. Disengagement parameters.

Disengagement Time appears to have no counterpart in previously defined data communication performance parameters. The ANSI (1974) parameter Total Overhead Time includes Disengagement Time as a component, but the contribution of the latter to the former will normally be small. There are some obvious similarities between Disengagement Time and Access Time, and in fact the disengagement and access functions are implemented by an identical "four-way handshake" in many packet switched systems (e.g., the ARPANET). The two functions do differ, of course, in the definition of their ending events.

User concern with Disengagement Time is based on the fact that Successful Disengagement is often a prerequisite to other user activities. The most obvious such activity is communication with another remote user. However, local communications may be affected as well. An example of the latter situation is an operator who uses a data terminal to communicate with both distant and local computer application programs. If his terminal remains logically connected to a distant program for a substantial period of time after he requests disengagement, he will be delayed in using the local program. Such delays are not at all unusual in some distributed computing systems, including the ARPANET (Payne, 1978).

Disengagement Time values range between zero and the "three times nominal" upper bound described earlier. A value of zero implies that all users involved in a session are free to initiate new sessions using the allocated facilities immediately upon issuance of a disengagement request. Large values for Disengagement Time suggest a system that not only wastes the user's time, but its own resources as well.

Appropriate user requirements for Disengagement Time depend on the particular application. Values less than a second may be appropriate in applications where a user continuously initiates sessions, as in "round robin" polling systems. Disengagement Time adds directly to the total round robin cycle time in such systems. The result is a general increase in system delays and corresponding data aging.

At the opposite performance extreme are applications where service usage periods are preceded by a long idle period. An example would be a retail inventory control system where accumulated receipts are transmitted to a central computer for processing once per day. In such cases, a Disengagement Time of many seconds or even minutes might be acceptable (as long as charging stops with the disengagement request).

User requirements for time sharing applications are typically intermediate between these extremes. An example is a Disengagement Time of 10 seconds specified in EPA (1980). It is usually appropriate and technically realistic to specify a Disengagement Time short enough to ensure that disengagement will not delay the next access attempt.

Data on Disengagement times for existing systems is sparse. One can infer minimum values of about 1 to 2 seconds for modern circuit-switched systems, since shorter depressions of a telephone hookswitch are often used to signal an operator or activate special functions. Linfield and Nesenbergs (1978) cite a typical "disconnection time" of 2 seconds in electronic switching systems. Payne (1978) reports measured values for operator Disengagement Time in the ARPANET in the range of 5.0 to 5.6 seconds. The latter values apply specifically to the Telnet protocol, and include 3.3 seconds of operator typing time for the CLOSE request. Wortendyke et al. (1982) demonstrate that Disengagement Times in packet switched networks may be substantially different for the user initiating disengagement (10-15 ms) and the other user (2.5 seconds). This finding justifies the ANS X3.102 decision to permit separate specifications for each user "where significant performance differences are expected."

The design features that·most strongly influence Disengagement Time are the type of resource sharing used and the degree of automation. Disengagement Time may be zero in simple datagram protocols. Services provided by dedicated lines typically offer very short Disengagement Times, because there are no shared system facilities which must be freed for use by other subscribers. The purpose of disengagement in such systems is simply to return the users to an established idle state after service usage.

Connectionless services such as electronic mail typically have somewhat longer Disengagement Times because there are local buffers in the system that must be freed for other users. Circuit-switched and virtual circuit systems typically have the longest Disengagement Times because there are shared resources (e.g., trunks) to be freed at both ends of the system. Kimmett and Seitz (1978) calculate Disengagement Time values of 0.5, 1.5, and 2.25 seconds for typical nonswitched, message-switched, and circuit-switched services, respectively.

4.3.2 Disengagement Denial Probability

Disengagement Denial Probability expresses the likelihood that a system will fail to detach a user from a session within a specified maximum time after issuance of a Disengagement Request. It is defined as the ratio of total disengagement attempts that result in Disengagement Denial to total disengagement attempts in a performance sample, excluding disengagement attempts that end in User Disengagement Blocking.

The Successful Disengagement outcome is indicated in one of two ways: (1) by the completion of a "disengagement confirmation" signal within the specified maximum disengagement time (in systems that provide such a signal); or (2) by the fact that the user is able to initiate a new access attempt within the specified maximum disengagement time (in systems that do not provide such a signal). The duration of the disengagement timeout period is three times the specified Disengagement Time. Disengagement Denial is distinguished from User Disengagement Blocking by comparison of the ancillary parameter values as discussed in Section 4.4.

Disengagement Denial Probability is significant to data communication users for two reasons. First, it provides information about the statistical nature of the Disengagement Time distribution. Like Access Time, Disengagement Time is the average of a truncated distribution. The probability that an individual disengagement attempt will exceed three times the specified value will be relatively high if the spread (variance) of the Disengagement Time distribution is large, and low if the spread is small. In other words, if the statistical distribution of Disengagement Times is relatively wide, with many long and many short times, the Disengagement Denial Probability will be higher than if the distribution is sharply peaked about the average. Disengagement Denial Probability values will be very low in systems with nearly constant Disengagement Times, since only a system malfunction (e.g., a software crash in a node computer) will cause disengagement timeout.

Disengagement Denial Probability is also significant to data communication users as a measure of system reliability. When a system completely fails to respond to disengagement requests, the effect on the user is often similar to that of a system outage (e.g., Access Outage or continuous Transfer Denials)--the service is unavailable until the problem is corrected.

Possible Disengagement Denial Probability values range between zero and one. A value of zero implies that the system never fails to disengage a user within the maximum timeout period. A value of one implies a system that never lets the user go within the maximum timeout period.

Appropriate user requirements for Disengagement Denial Probability depend on the service usage pattern. Low values are appropriate in polling and similar applications, where many separate data communication sessions are established in quick succession. Quite high values can be tolerated in applications where usage is normally preceded by a long idle period. Reliability requirements and the availability of backup facilities should also influence user requirement specifications. Nesenbergs et al. (1980) suggest a Disengagement Denial Probability requirement of 10^{-3} for interactive packet switching network users. A value of 10^{-5} is specified in EPA (1980).

Disengagement Denial Probability values are influenced by two general system design characteristics:

1. The relative complexity of the disengagement protocol employed.

2. The inherent accuracy and reliability of the facilities that implement that protocol.

In general, the lowest Disengagement Denial Probability values are found in connectionless systems. In such systems, the disengagement of each user is a simple, local function. Successful Disengagement does not require remote communication, and the disengagement process is thus not influenced by system transmission imperfections. Disengagement Denial Probability may even be zero in simple datagram protocols.

Disengagement is more complex, and therefore more subject to failure, in virtual-circuit systems. In such systems, disengagement typically involves a full four-way handshake between session participants. That is, a close message must be transferred from source to system, system to destination, destination to system, and system back to source to complete disengagement. Successful Disengagement of the source thus requires two successful passages of a close message through the system. If such a protocol is combined with a flow control mechanism that discards packets as a means of controlling system congestion or excessive delay, Disengagement Denial may be a rather frequent occurrence.

The ARPANET illustrates such a situation. Kleinrock et al. (1976) report a 10^{-2} loss probability for packets entering the network. Logically, one would expect the loss probability for one or both of two close requests to be about twice that high—a Disengagement Denial Probability of 2×10^{-2}. Payne (1978) reports a measured value for this parameter of 3×10^{-2}.

4.4 Ancillary Parameters

It is important to remember that data communication performance is user dependent. That fact is often disregarded in the design and operation of data communication systems, with the result that many systems are inefficient in meeting end user needs. This section describes a method of quantifying user dependence, via the ANS X3.102 ancillary parameters, to improve system performance specification and cost effectiveness.

The essential facts surrounding the user dependence problem are these:

1. Most data communication systems require user inputs at various points in a data communication session.

2. The user actions that generate those inputs inevitably take time. Often, the system has no alternative but to delay its own activities until the necessary user actions are accomplished.

3. The time required to complete the primary communication functions is therefore often dependent on user performance time.

The users and the system must normally be regarded as jointly responsible for determining overall data communication performance. The purpose of the ancillary parameters is to describe the relative contributions of the users and the system to observed communication delays.

In the voice telephone access example, Figure 4, we observed that overall access performance time in the public switched telephone network depends on both the system's speed in signaling and switching, and the user's speed in dialing and answering. User delays can also influence performance for the user information transfer and disengagement functions. For example, the User Information Bit Transfer Rate with an operator at a keyboard terminal is dependent on the user's think time and typing speed. Similarly, in systems that use a full four-way handshake for disengagement, the user not originating disengagement must respond to a close request from the system before the originating user can be successfully disengaged.

The four ancillary parameters express these user influences on communication performance in quantitative terms. Each parameter relates to a corresponding primary "speed" parameter, and expresses the average proportion of the performance time associated with that parameter that is attributable to user delays. Ancillary parameters are defined for Access Time, Block Transfer Time, and Disengagement Time, and also for the sample input/output time associated with User Information Bit Transfer Rate.

There are relatively few precedents for the ancillary parameters in data communications literature. Most published studies either disregard user dependence or make assumptions that eliminate or conceal its effect. Three exceptions are worthy of note:

1. The study of Duffy and Mercer (1978) on network performance and customer behavior during direct distance dial call attempts on the public switched telephone network. Among other findings, this study reports that "customer-determined components of the call setup time make up 71 percent of the total setup [access] time."

2. The study of Jackson and Stubbs (1969) on user/computer interactions in a typical remote-access timesharing system. A significant conclusion of this study is that "users themselves contribute substantially to the communication costs of their real-time computer access calls by introducing delays." Quantitative data from this study have been presented earlier.

3. The work of Kleinrock (1976) and others in applying queueing theory to computer networks. The concepts of customer "arrivals," interarrival times, and service times provide a natural framework for describing user dependence, although relatively few studies have actually applied them to that problem. One such application has been described earlier in this report (Kleinrock et al., 1976).

The ancillary performance parameters are significant for three reasons. First, each parameter can be used as a correction factor, to calculate user-independent values for the associated speed parameters. If W is the specified performance time for a function (e.g., access) and F is the associated ancillary parameter value (where F is between zero to one) then the user-independent performance time for that function is:

$$[1 - F][W]$$

The factor [1 - F] is the average system performance time fraction (the complement of F). Similarly, a user-independent value for User Information Bit Transfer Rate (R) can be estimated as:

$$R_{ind} = R/[1 - F] \quad .$$

In each case, the user-independent value expresses the performance that would be provided if all user delays were zero--that is, if all user activities were performed in zero time.

As an example, assume the Access Time value for the telephone service of Figure 4 is 25 seconds, and the specified User Fraction of Access Time is 0.6. Then the user-independent Access Time value (that is, the average system delay during access) is (1 - 0.6)(25) or 10 seconds. As another example, assume a

User Information Bit Transfer Rate of 600 bits per second and an associated User Fraction of Input/Output Time of 0.75. Then the user-independent User Information Bit Transfer Rate value (that is, the rate that would occur if there were no user-caused input or output delays) is 600/(1 - 0.75) or 2400 bits per second.

The ancillary parameters are also significant because they provide a basis for identifying the entity responsible for timeout failures--for example, whether the user or the system should be charged with the failure when an access attempt is not completed within the maximum access time. This decision is made by calculating a user performance time fraction for the particular (unsuccessful) trial in question, and then comparing the calculated value with the corresponding specified ancillary parameter value. If the user fraction for the particular trial exceeds the specified value, the failure is attributed to the user; otherwise, the failure is attributed to the system.

As an example of this process, assume again a service with a specified Access Time of 25 seconds and an associated user fraction of 0.6. The corresponding maximum Access Time would be 75 seconds (three times the specified value). Now, assume a particular access attempt times out, and it is determined that 50 seconds out of the 75-second total access performance period are attributable to user delays. The user fraction for the particular trial is then 50/75 or 0.67. Since 0.67 (the value for the particular trial) is larger than the specified fraction (0.6), the failure would be attributed to User Blocking and would be excluded in calculating values for the access performance parameters.

The ancillary performance parameters are also useful in assessing the efficiency and utilization of a service. The ancillary parameter values give communication managers and providers important information about the relative economy of a service for a particular user. High values indicate that overall performance is dominated by user delays. In such cases, a potential for more economical service through resource sharing may exist. Concentration is a familiar way of exploiting this potential. Low ancillary parameter values indicate that overall performance is dominated by system delays, suggesting that little resource sharing potential exists.

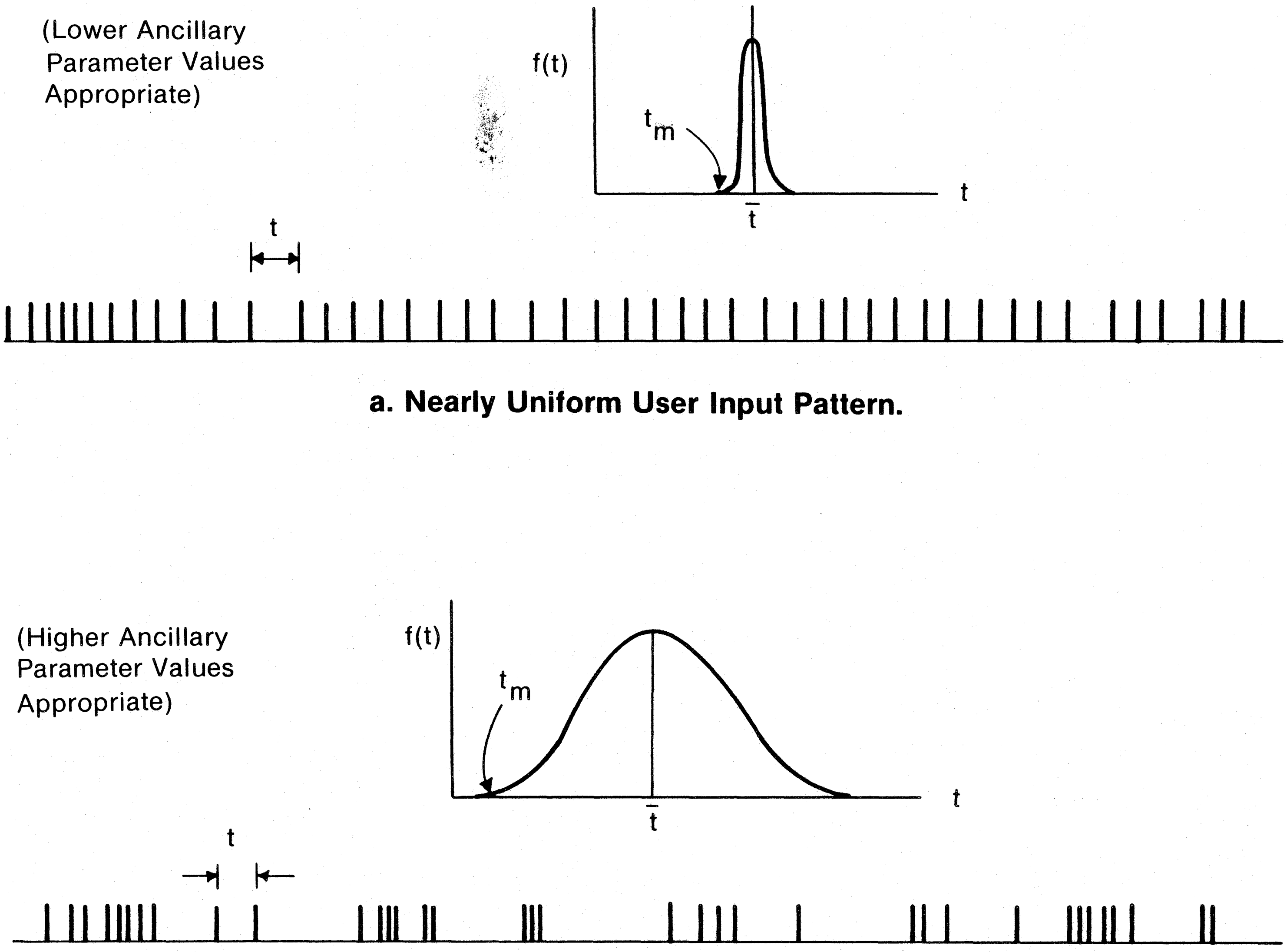
Communication users view the ancillary parameters from two perspectives, depending on the primary function in question. The key issue in the case of access and disengagement is ease of use. Low ancillary parameter values indicate a service that can be used with relatively little user investment in time and effort (e.g., an "off-hook" service). High values indicate a service that demands more user resources (e.g., a service with lengthy, elaborate circuit establishment procedures). The emphasis given "equal exchange access" in the AT&T divestiture ruling (Greene, 1982) indicates that this is a significant performance issue.

Users view the two ancillary parameters associated with user information transfer somewhat differently. Ease of use is still desirable, but its importance is much less significant than what might be called "reserve capacity"--the ability of the system to keep up with the user during momentary bursts of high-speed input. The more bursty the input, the more important such reserve capacity is. An all too familiar example of insufficient reserve capacity is a system that falls behind in echoing typed characters above a certain typing speed. High ancillary parameter values indicate that there is a substantial reserve capacity in the system, and conversely. A similar relationship holds in the case of user-controlled output.

Since ease of use is the key factor in the case of access and disengagement, ancillary parameter values for these functions should normally be specified on the basis of the value of the user's time. Low specified values (e.g., less than 0.1) are appropriate in applications where the user's time is extremely valuable. High values (e.g., greater than 0.9) may be tolerated in applications where the user has available time that cannot be used in other productive ways. An example of a user in the former category is a computer program controlling a critical real-time process. Users of a recreational game network like that proposed by Lucky (1979) might fall in the latter category.

Figure 20 illustrates the influence of the user input pattern (variation in input rate) on the User Fraction of Input/Output Time. If the user input pattern is uniform or nearly so, there is little need for reserve capacity and a relatively low ancillary parameter value is appropriate. However, if the user input pattern is very bursty, a substantial reserve capacity must be provided if the system is to keep up with the user during input bursts, and a higher ancillary parameter value is appropriate.

As a numerical example, assume a typist generates user information characters intermittently, with average and minimum intercharacter times of 500 and 100 milliseconds. These times correspond to typing speeds of 20 and 100 words per minute. Assume that system propagation and storage times are



a. Nearly Uniform User Input Pattern.

b. Very Bursty User Input Pattern.

Figure 2Ø. Impact of user input pattern on user fraction of input/output time.

negligible. The average character input rate and maximum character input rate differ by a factor of five (2 vs. 10 characters per second). A reserve capacity of 8 characters per second is needed to accommodate bursts of user input, so a relatively high User Fraction of Input/Output Time should be specified.

It may seem strange that a high user fraction is "bad" in the case of access and "good" in the case of transfer. The explanation is that in the case of access, the user is trying to obtain a service and a larger user fraction means that more user time must be devoted to that effort. In the case of transfer, the user has obtained the service and a higher user fraction indicates a faster, more responsive service.

The single design feature that most strongly influences ancillary parameter values is the user/system interface protocol. The lowest ancillary parameter values are observed in services where the system controls the transfer of information across both user/system interfaces. In such services, few or even no user actions may be required to complete a function. Relatively high values are observed in services where the users control or participate in controlling transfer across both interfaces. Each required user action adds to the total user delay.

In the case of access and disengagement, the ancillary parameter values are largely determined by the type of switching used. Connectionless services typically provide low ancillary parameter values. The functions of access and disengagement are simple or even null in such services, and few user/system interactions are involved. The opposite is true of connection-oriented services. Dedicated services (e.g., leased lines) provide relatively low values because the key user choices (e.g., the address of the desired destination) are hardwired. User Fraction of Access Time values of 0, 0.4, and 0.19 are calculated for particular message-switched, circuit-switched, and dedicated services, respectively, in Kimmett and Seitz (1978).

The impact of system design on the ancillary user information transfer parameter values is best communicated by examples. Consider first the case of simplex communication between two terminal operators via a circuit-switched telephone (or telex) connection. One operator enters the information on-line at a CRT, while the other simply reads the resulting output. In this situation, the User Fraction of Input/Output Time will normally be quite high (perhaps in excess of 0.9), because the user controls input and his input rate is much lower than the system's input capacity. The high ancillary parameter value indicates substantial reserve capacity and also suggests a potential for efficiency improvement through buffering or resource sharing.

Such improvement might be provided by a computer mail service. Assume the user's message is generated manually as described earlier, but off-line. Communication service then begins only when the user completes a message (and any associated editing) and requests transmission. The user Fraction of Input/Output Time will be low or even zero in such a service, because the system controls both input and output. In this case, the low ancillary parameter value indicates little reserve capacity and little potential for communication efficiency improvement. Similar relationships hold between the control of user information output and the User Fraction of Block Transfer Time.

These examples illustrate a strong dependency between ancillary parameter values and user/system interface protocols. To ensure cost effectiveness, the selection of ancillary parameter values should be considered carefully in developing performance requirements.

4.5 Summary

Table 3 summarizes the 21 user-oriented performance parameters defined in American National Standard X3.102. The parameters express performance relative to three primary communication functions: access, user information transfer, and disengagement. These functions correspond to connection establishment, data transfer, and disconnection in connection-oriented services, but are also applicable to connectionless services (e.g., electronic mail). They subdivide an overall data communication session in accordance with the user's perception of service, and provide a specific focus for performance description.

In defining the standard parameters, each function was considered with respect to three possible results, or outcomes, an individual performance trial might encounter: successful performance, incorrect performance, and nonperformance. These possible outcomes correspond closely with the three major performance concerns (or "performance criteria") most frequently expressed by data communications users: speed, accuracy, and reliability.

One or more "primary" parameters were defined to express performance relative to each function/criterion pair. As an example, four primary parameters were defined for the access function: one speed parameter (Access Time), one accuracy parameter (Incorrect Access Probability), and two

Table 3. Matrix Representation of the ANS X3.102 Parameters

FUNCTION	PERFORMANCE CRITERION			PERFORMANCE TIME ALLOCATION
	SPEED	ACCURACY	RELIABILITY	
ACCESS	ACCESS TIME	INCORRECT ACCESS PROBABILITY	ACCESS DENIAL PROBABILITY ACCESS OUTAGE PROBABILITY	USER FRACTION OF ACCESS TIME
USER INFORMATION TRANSFER	BLOCK TRANSFER TIME	BIT ERROR PROBABILITY BIT MISDELIVERY PROBABILITY EXTRA BIT PROBABILITY	BIT LOSS PROBABILITY	USER FRACTION OF BLOCK TRANSFER TIME
		BLOCK ERROR PROBABILITY BLOCK MISDELIVERY PROBABILITY EXTRA BLOCK PROBABILITY	BLOCK LOSS PROBABILITY	
	USER INFORMATION BIT TRANSFER RATE	TRANSFER DENIAL PROBABILITY		USER FRACTION OF INPUT/OUTPUT TIME
DISENGAGEMENT	DISENGAGEMENT TIME	DISENGAGEMENT DENIAL PROBABILITY		USER FRACTION OF DISENGAGEMENT TIME

Legend:

□	Primary Parameters
▨	Ancillary Parameters

reliability parameters (Access Denial Probability and Access Outage Probability). Failures attributable to user nonperformance (e.g., called user does not answer) were excluded in defining each primary parameter.

The ANS X3.102 parameters also include four "ancillary" parameters. Each ancillary parameter relates to a primary "speed" parameter, and expresses the average proportion of the performance time associated with that parameter that is attributable to user delays. As an example, the primary parameter Access Time normally includes delays attributable to the users (e.g., dialing time, answer time) as well as delays attributable to the system (e.g., switching time). The ancillary parameter User Fraction of Access Time expresses the average proportion of total Access Time that is attributable to the user delays.

The ancillary parameters have two important uses. First, they provide a method of factoring out user influence on the primary speed parameters, to produce user independent values characterizing the unilateral performance of the system. This is necessary in comparing performance values determined under different usage conditions. Second, the ancillary parameters provide a basis for determining the entity (user or system) responsible for nonperformance failures (e.g., access timeouts). This decision is made by calculating a user performance time fraction for the particular (unsuccessful) trial in question, and then comparing the calculated value with a corresponding (specified) ancillary parameter value.

The ANS X3.102 parameter definitions differ from those presented in earlier standards and specifications in two respects. First, the ANS X3.102 parameters are defined on the basis of general, system-independent reference events (e.g., access request) rather than on the basis of particular system-specific interface signals (e.g., issuance of an "off-hook" signal or a Call Request packet). System-specific interface signals are mapped into corresponding reference events on the basis of the user interface involved, the type of information transferred (e.g., user information or overhead) and the nature of the activity the transfer initiates. Defining the parameters on the basis of general reference events makes the parameters system-independent, and enables them to be used in comparing performance between systems that employ different user interface protocols (e.g., X.25 and X.21).

A second distinguishing characteristic of the ANS X3.102 parameter definitions is their expression in mathematical form. The parameter definitions are based, in each case, on the concept of an observed performance "sample"—i.e., a large number of successive performance trials distributed in appropriate outcome "bins." Individual parameters are defined as random variables on an associated probability sample space. The mathematical approach eliminates the ambiguity often associated with purely narrative parameter definitions, and also provides a standard procedure for calculating the performance parameter values.

It is anticipated that ANS X3.102 will be useful to communication users, communication providers, and communication managers in three distinct applications:

1. User Requirements Specification. In this application, the standard is used to specify the communication performance requirements of a particular user. The standard enables the analyst to assess the impact of communication performance on user processes without presupposing any particular system design, and provides a system-independent, functional framework for evolving user requirements.

2. Service Performance Characterization. In this application, the standard is used to characterize the end-to-end performance of a particular communication system or service. The standard provides suppliers with a single, uniform method of representing performance to potential users.

3. Service Selection. In this application, the standard is used to compare various alternative means of meeting a particular user requirement. The standard provides the communication manager with a practical method for evaluating service utility.

Functional specification of data communication services has been difficult in the past because of the lack of user-oriented, system-independent performance descriptors. American National Standard X3.102 provides such descriptors--in essence, a common language for relating the performance needs of end users with the capabilities of supplier systems. Its use should enable more precise definition of user requirements, facilitate provider competition, and lead to more cost effective data communication system designs.

5. ACKNOWLEDGEMENTS

The authors thank Joe Hull of ITS, Professor Earl Eyman of the University of Iowa, and Ray Moore and Jerry Linn of NBS/ICST for their thoughtful and constructive review of the draft manuscript. We also thank Cathy Edgar and Lenora Cahoon of ITS for their assistance in manuscript preparation and editorial review.

6. REFERENCES

AGREE (1957), Reliability of military electronic equipment, Advisory Group on the Reliability of Electronic Equipment, U.S. Government Printing Office, Washington, DC. Foundation work on reliability theory.

ANSI (1971), American National Standard, Procedures for the use of the communication control characters of American National Standard Code for Information Interchange in specified data communication links, X3.28-1971. Defines use of ASCII control characters in communications.

ANSI (1974), American National Standard, Determination of performance of data communication systems, X3.44-1974. Original ANSI standard on data communication performance assessment, focusing on character-oriented protocols (e.g., ANSI X3.28).

ANSI (1980), American National Standard, Determination of performance of data communication systems that use bit-oriented communication control procedures, X3.79-1980. Second ANSI standard on data communication performance assessment, focusing on bit-oriented protocols (e.g., ANSI X3.66). Expanded coverage of networks.

Armed Services Investigating Committee (1971), Review of Department of Defense worldwide communications, Phase I, Committee on Armed Services, U.S. House of Representatives, May 10. Tragicomic history of U.S. military communication failures in three international crisis situations--the Israeli attack on the U.S.S. *Liberty*, the North Korean seizure of the U.S.S. *Pueblo*, and the North Korean shoot-down of an unarmed EC-121 reconnaisance aircraft.

AT&T (1961), Switching Systems (American Telephone and Telegraph Company, 195 Broadway, New York, NY). Introduction to basic switching and traffic analysis.

AT&T (1968), Bell System Technical Reference, Model 37 teletypewriter stations for Dataphone^R service, September. Defines terminal features, options, and operating procedures.

AT&T (1971), Bell System Technical Reference Pub. 41007, 1969-70 telecommunications network connection survey, April. Comprehensive report on transmission quality in the public switched network.

AT&T (1978), Technical overview, Advanced Communications Service, American Telephone and Telegraph Company, Business Marketing Operations, Delivery System Strategy Group, November. Preliminary information on ACS implementation and service features.

Bodson, D. (1978), The national communications system, IEEE Communications Society magazine, March. Describes responsibilities and activities of the NCS organization.

CCITT (1973), Data Transmission, Green Book, Volume VIII, International Telecommunications Union, Geneva, Switzerland. Presents recommendations after the fifth Plenary. Presents standard definitions for several hundred key telecommunications terms.

CCITT (1978), Datagram service and interface, Study Group VII contribution No. 133, COM VII No. 133-E, January. Defines datagram service quality objectives.

Crow, E. L. (1974), Confidence limits for digital error rates, Office of Telecommunications Report 74-51, November. Derivation of confidence limits applicable to small probabilities. Independence of successive trials assumed.

Crow, E. L. (1978), Relations between bit and block error probabilities under Markov dependence, Office of Telecommunications Report 78-143, March. Derivation of confidence limits for block error probability. Approximation for the probability of m errors in a block of n bits.

Crow, E. L. (1979), Statistical methods for estimating time and rate parameters of digital communication systems, NTIA Report 79-21, June. Derivation of confidence limits and sampling procedures for the FED STD 1033 performance time, time rate, and rate efficiency parameters. Considers the effect of truncating a time distribution of three times its mean.

Crow, E. L., and M. J. Miles (1976a), A low-cost, accurate statistical method to measure bit error rates, Proceedings of the International Conference on Computer Communications, August. Practical procedure for bit error rate measurement.

Crow, E. L., and M. J. Miles (1976b), A minimum cost, accurate statistical method to measure bit error rates, Proceedings of the International Conference on Computer Communications, Toronto, Canada, August. Practical, understandable procedure for obtaining BER measurements of given precision.

Crow, E. L., and M. J. Miles (1977), Confidence limits for digital error rates from dependent transmissions, Office of Telecommunications Report 77-118, March. Extension of Crow's 1974 report to the case of dependent errors. Two-state (Markov) error model assumed.

DCA (1975), System Performance Specification of AUTODIN II Phase I, Defense Communication Agency, November. States performance requirements for a "second-generation" packet switching network.

Duffy, F. P., and R. A. Mercer (1978), A study of network performance and customer behavior during direct-distance-dialing call attempts in the U.S.A., Bell Sys. Tech. J. 57, January. Comprehensive measurements of DDD call attempts. Treats user delay and its influence on measured connection times.

EPA (1980), Request for quotations, telecommunications network service, Request No. WA-80-D289/ldm, May 28. Specifies requirements for a nationwide timesharing network to interconnect 300 and 120 bps terminals to host computers.

Feistel, H. (1973), Cryptography and computer privacy, Scientific American 228, No. 5, May. Lucid introductory paper with practical examples.

FIPS (1981), Guidelines for implementing and using the NBS Data Encryption Standard, FIP PUB 74, National Bureau of Standards, available from U. S. Department of Commerce, NTIS, Springfield, VA 22161.

Fuchs, E., and P. Jackson (1970), Estimating distributions of random variables for certain computer communications traffic modes, Communications of the ACM 13, No.12, December. Extends results of Jackson and Stubbs (1969).

GAO (1977), Better management of defense communications would reduce costs, Report to the Congress by the Comptroller General of the United States, LCD-77-106, December 14. Incisive examination of current inefficiencies in military communications procurement. Excessive use of dedicated services is reported.

Gray, J. P. (1972), Line control procedures, Proc. IEEE 60, November. Lucid presentation of protocol theory in terms of finite-state machines, with practical examples.

Greene (1982), Opinion Entered in AT&T Antitrust Case, District Court, District of Columbia, Civil Action Numbers 74-1698, 82-0192, and 82-0025 (PI). Summary of AT&T divestiture provisions.

Grubb, D. S., and I. W. Cotton (1975), Criteria for the performance evaluation of data communications services for computer networks, National Bureau of Standards Technical Note 882, September. Very readable survey of data communication performance parameters and issues. User perspective.

GSA (1978), Federal Property Management Regulations, Title 41, Subchapter F, ADP and Telecommunications, Amendment F-35, November. Defines procedures for Federal agency procurement of data communications equipment and services. Identifies agency coordination and reporting responsibilities.

GSA (1979), Interim Federal Standard 1033, Telecommunications: digital communication performance parameters, August 29. Published version of the original Interim Federal Standard, available from the Office of the Manager, National Communications System Technology and Standards, Washington, DC 20305. Specified parameters differ from those ultimately defined in ANS X3.102.

Hamming, R. W. (1950), Error detecting and error correcting codes, Bell Sys. Tech. J. 29, April. Highly readable foundation of error control.

Jackson, P., and C. Stubbs (1969), A Study of Multi-Access Computer Communications, Proceedings of the Spring Joint Computer Conference, May 14-16. Builds a "data stream model" from measurements of user/program interactions in a timesharing environment.

Kelley, K. G. (1977), An evaluation of data transfer requirements for the future DCS, DCEC Technical Note 24-77, November. Projects future DCS traffic volumes and "response time" requirements.

Kimmett, F. G., and N. B. Seitz (1978), Digital communication performance parameters for Federal Standard 1033, NTIA Report 78-4, Vol. II, application examples, May. Develops FED STD 1033 performance parameter values for three representative data communication services: nonswitched, circuit-switched, and message-switched.

Kleinrock, L. (1976), Queueing Systems, Volume II, Computer Applications (John Wiley & Sons, Inc., New York, NY). Outstanding, definitive text on design and analysis of computer communication networks. Much useful system performance information. Emphasis on packet switching.

Kleinrock, L., W. F. Naylor, and H. Opderbeck (1976), A study of line overhead in the ARPA network, Communications of the ACM 10, No. 1, January. Definitive analytical and experimental results on potential and actual ARPANET efficiency.

Kobylar, A. W., and H. A. Malec (1973), System effectiveness trade-off in a space-time-space network for a digital exchange, GTE Automatic Electric Tech. J., July. Defines reliability measures and presents simulation results for a typical PCM switch.

Kuhn, T. G. (1963), Retransmission error control, IEEE Trans. Commun. Sys. CS-11, No. 2, June. Practical study with emphasis on block size optimization. Simple block parity error detection schemes are illustrated.

Linfield, R. F., and M. Nesenbergs (1978), Access area switching and signaling concepts, issues, and alternatives, NTIA Report 78-2, May. PABX and signaling alternatives for future Army access area communications systems are discussed.

Lucky, R. W. (1979), Gamenet, IEEE Communications Magazine 17, No. 6, November. Fanciful, futuristic view of tomorrow's electronic games.

MacRae, D. D., F. A. Perkins, D. J. Risavy, and J. N. York (1976), 16 Kb/s Data modem techniques, RADC-TR-76-311, Harris Corp., October. Study of digital speech transmission quality.

Malec, H. A. (1975), Telephone switching system reliability - past, present, and future. Proceedings of the National Telecommunications Conference, December 1-3. Surveys reliability and effectiveness measures for commercial telephone switching systems.

Martin, J. (1976), Telecommunications and the Computer, Second Edition, (Prentice-Hall, Inc., Englewood Cliffs, NJ). Comprehensive, readable book covering basic technology, administration, and applications of telecommunications.

McFadyen, H. J. (1976), Systems network architecture: an overview, IBM Sys. J. 15, No. 1. Introduction to a special issue on SNA.

Nesenbergs, M. (1975), Study of error control coding for the U.S. Postal Service Electronic Message System, Office of Telecommunications Report, May. Surveys candidate error control techniques for a high data rate satellite network.

Nesenbergs, M., W. J. Hartman, and R. F. Linfield (1980), Performance parameters for digital and analog service modes, NTIA Report 81-57, January. Analysis of user and system requirements for digital and analog service of the future Defense Communications System.

Payne, J. A. (1978), ARPANET host-to-host access and disengagement measurements, NTIA Report 78-3, May. Describes access and disengagement performance measurements for ARPANET connections established via the Telnet protocol.

Popek, G. J. (1974), Protection structures, Computer magazine, June. Examines computer security concerns and protection strategies.

Roberts, L. F., and B. D. Wessler (1970), Computer network development to achieve resource sharing, Proceedings of the Spring Joint Computer Conference, 36, May 5-7. Early overview of the ARPA network.

Schwartz, M., R. Boorstyn, and R. Pickholtz (1972), Terminal-oriented computer communications network, Proc. IEEE 60, No. 11, November. Tutorial presentation of four computer communication networks.

Seitz, N. B., and P. M. McManamon (1978), Digital communication performance parameters for proposed Federal Standard 1033, NTIA Report 78-4, Volume 1, standard parameters, May. Comprehensive presentation of the standard's technical basis.

Seitz, N. B., K. P. Spies, and E. L. Crow (1981a), Telecommunications: digital communication performance measurement methods, proposed Federal Standard 1043, Version 5, May. Draft proposed Federal Standard available from the authors of this report.

Seitz, N. B., K. P. Spies, and E. L. Crow (1981b), Data communication performance measurement--a proposed Federal Standard, Proc. 1981 National Telecommunications Conference, New Orleans, Louisiana, November. Technical paper summarizing objectives and content of proposed Federal Standard 1043.

Shannon, C. E. (1948), A mathematical theory of communication, Bell Sys. Tech. J. 27, July. Profoundly significant, elegantly presented foundation of communication and information theory.

Sunshine, C. A. (1975), Interprocess communication protocols for computer networks, Digital Systems Laboratory, Department of Electrical Engineering, Stanford University, Technical Report No. 105, December. Comprehensive, readable thesis on communication protocol analysis and design. End-to-end point of view.

Utlaut, W. F. (1978), Spread spectrum: principles and possible application to spectrum utilization and allocation, IEEE Communications Society magazine 16, No. 5, September. Tutorial summary of spread-spectrum principles and applications.

Wortendyke, D. R., N. B. Seitz, K. P. Spies, E. L. Crow, and D. S. Grubb (1982), User-oriented performance measurements on the ARPANET: the testing of a proposed Federal standard, NTIA Report 82-112, November. Comprehensive report on a trial application of proposed Federal Standard 1043. Measured values for the ANS X3.102 parameters are presented.

FORM **NTIA-29**
(4-80)

U.S. DEPARTMENT OF COMMERCE
NAT'L. TELECOMMUNICATIONS AND INFORMATION ADMINISTRATION

BIBLIOGRAPHIC DATA SHEET

1. PUBLICATION NO.
NTIA Report 83-125

2. Gov't Accession No.

3. Recipient's Accession No.

4. TITLE AND SUBTITLE
American National Standard X3.102 User Reference Manual

5. Publication Date
October 1983

6. Performing Organization Code
NTIA/ITS.N3

7. AUTHOR(S)
N. B. Seitz, D. S. Grubb

9. Project/Task/Work Unit No.
9104120

8. PERFORMING ORGANIZATION NAME AND ADDRESS
National Telecommunications & Information Admin.
Institute for Telecommunication Sciences
325 Broadway
Boulder, CO 80303

10. Contract/Grant No.

11. Sponsoring Organization Name and Address
National Telecommunications & Information Admin.
Institute for Telecommunication Sciences
325 Broadway
Boulder, CO 80303

12. Type of Report and Period Covered

13.

14. SUPPLEMENTARY NOTES

15. ABSTRACT *(A 200-word or less factual summary of most significant information. If document includes a significant bibliography or literature survey, mention it here.)*

American National Standard X3.102 defines a set of 21 standard parameters that provide a uniform means of specifying the performance of data communication systems and services as seen by users. This report is basically an explanation and elaboration of that standard. The report first outlines the benefits of using the standard from the viewpoint of the end user, the communication provider, and the communication manager. The report then summarizes the standard's overall approach and content in informal, non-technical terms. Finally, the report examines the meaning and importance of each standard parameter in a series of tutorial parameter descriptions. Typical parameter values are presented and their design implications are discussed.

16. Key Words *(Alphabetical order, separated by semicolons)*

---

17. AVAILABILITY STATEMENT

☒ UNLIMITED.

□ FOR OFFICIAL DISTRIBUTION.

18. Security Class. *(This report)*
Unclassified

19. Security Class. *(This page)*
Unclassified

20. Number of pages
100

21. Price:

☆ U.S. Government Printing Office: 1980—678-495/529