Bull

Glossary

AIX

ORDER REFERENCE
86 A2 29EF 01

Bull

Glossary

AIX

Software

October 2002

BULL CEDOC
357 AVENUE PATTON
B.P.20845
49008 ANGERS CEDEX 01
FRANCE

ORDER REFERENCE
86 A2 29EF 01

Printed in France

Suggestions and criticisms concerning the form, content, and presentation of this book are invited. A form is provided at the end of this book for this purpose.

To order additional copies of this book or other Bull Technical Publications, you are invited to use the Ordering Form also provided at the end of this book.

Trademarks and Acknowledgements

We acknowledge the right of proprietors of trademarks mentioned in this book.

AIX® is a registered trademark of International Business Machines Corporation, and is being used under licence.

UNIX is a registered trademark in the United States of America and other countries licensed exclusively through the Open Group.

About This Book

This glossary provides definitions of specialized terms used in the documentation library for the AIX operating system. Terms that are defined in nontechnical dictionaries and that have no special meaning in information processing are not defined in this glossary.

This glossary includes terms and definitions from the following publications:

The *Information Technology Vocabulary*, developed by Subcommittee 1 of the International Organization for Standardization and the International Electrotechnical Commission (ISO/IEC JTC1/SC1).

The *American National Standard Dictionary for Information Systems*, copyright 1990 by the American National Standards Institute (ANSI).

The *IBM Dictionary of Computing*, copyright 1999 by International Business Machines Corporation.

ISO 9000

ISO 9000 registered quality systems were used in the development and manufacturing of this product.

Glossary: Contents

Chapter 1. Special Characters

2–1/2 dimension
A drafting term that describes limited associativity between orthogonal views; two dimensions with a perspective view that cannot be modified once created and that cannot be rotated. Allows for semi–automatic creation of isometric views.

3270 Device Emulation
Support that allows a local or remote device on one system to appear as a 3270 device to another system.

3270 Host Connection Program (HCON)
A program that allows a system unit to emulate a workstation or printer attached to a System/370 host. See also *session profile* on page 2-216 and *HCONuser* on page 2-106.

$HOME An environment variable, set by the system, that designates a user's home directory. Many programs use this variable to designate a directory where they store temporary work files.

/usr file system
Contains files and programs necessary for operating the machine.

/tmp file system
A shared storage location for files.

/var file system
Contains files that are variable on a per–client basis, such as spool and mail files.

/ file system The root file system; contains files that contain machine–specific configuration data.

Chapter 2. Alphabetical List

A

A See *ampere* on page 2-9

abbreviation A character string representing a longer character string.

abend (1) The abnormal end of a task.
(2) The ending of a task before its completion because of an error condition that recovery facilities cannot resolve while the task is running.

abort This term refers to transactions that do not complete. Any changes made by a transaction that is aborted, for whatever reason, must be undone. Once a transaction is undone (**rolled back**), no evidence that the transaction was ever attempted remains outside of records in the transaction processing system's log. See also *rolled back* on page 2-206.

absolute address
An address that, without the need for further evaluation, identifies a storage location or a device.

absolute device
A locating device, such as a tablet, that reports its position to the operating system as a set of numbers on a coordinate system.

absolute time A point on a time scale.

absolute value The numeric value of a real number regardless of its algebraic sign (positive or negative).

Abstract Syntax Notation One (ASN.1)
A notation that enables both complicated types to be defined and values of these types to be specified.

abstraction (data)
A data type with a private representation and a public set of operations.

accelerator In AIXwindows, a keyboard alternative to a mouse button action. For example, holding the <Shift> and <M> keys on the keyboard can be made to post a menu in the same way that a mouse button action does. Accelerators typically provide increased input speed and greater convenience.

access To obtain data from or to put data in storage.

access control list (ACL)
(1) A file attribute that contains the basic and extended permissions that control access to the file.
(2) A list of hosts (maintained by Enhanced X-Windows) that have access to client programs. By default, only programs on the local host and those in this list, also known as "access list," can use the display. The list can be changed by clients on the local host; some server implementations can also modify the list. The authorization protocol name and data received by the server at connection setup may affect the list as well. See also *discretionary access control* on page 2-71.

access control list entry
Data in an access control list that specifies a set of permissions. In the case of a principal or group entry, the permission set is that which may be granted to a principal having the privilege attribute specified in the entry; in the case of a mask entry, the permission set is that which masks the permission set in a principal or group entry.

access declaration
Used to restore access to members of a base class.

access level In computer security, the level of authority a user has while accessing a secured file or library.

access permission
A group of designations that determine who can access a particular file and how the user can access the file. See also *permission code* on page 2-173.

access procedure
The procedure or protocol used for gaining access to a shared resource. In a local area network, the shared resource is the transmission medium. The medium access procedures specified by the IEEE 802 standard are CSMA/CD token bus and token ring.

access resolution
The process by which the accessibility of a particular class member is determined.

access right See *permission* on page 2-173.

accessible Said of an object for which the client possesses a valid designator or handle.

account The log in directory and other information that give a user access to the system.

accounting system
A utility that monitors various aspects of system operations; it collects detailed data on each transaction and provides tools for processing the data to produce different kinds of reports.

ACK See *acknowledgment character* on page 2-4.

ACK0 A transmission control character for even positive acknowledgment; indicates that text was received without transmission errors.

ACK1 A transmission control character for odd positive acknowledgment; indicates that text was received without transmission errors.

acknowledge (1) To answer. To respond to a poll, address, or message.
(2) In the X.25 API, to confirm that a data packet with the D–bit set has arrived.

acknowledge timeout
The number of seconds that a station should wait for an acknowledgment from a remote station after sending data.

acknowledgment character (ACK)
In binary synchronous communications, a transmission control character sent as an affirmative response to a data transmission.

ACL See *access control list* on page 2-3.

ACM Association for Computing Machinery.

ACSE Association Control Service Element.

action (1) A defined task that an application performs. An action modifies the properties of an object or manipulates the object in some way. A processing step or operation.
(2) In the **awk** command, the **lex** command, and the **yacc** command, a C language program fragment that defines what the program does when it recognizes input.
(3) In CDE, a desktop construct that provides a method for running applications, executing commands, and other activities such as printing, removing files, and changing directories. Actions are defined in a database of configuration files.

action icon In CDE, an icon that represents an action in a File Manager or Application Manager window, or in the workspace. An action icon is created by creating an empty executable file with the same name as the action it represents.

action label In CDE, the name displayed under the icon of an action. See also *action name* on page 2-5.

action name In CDE, the name associated with an action, which by default is displayed under the icon for the action. See also *action label* on page 2-5.

action statement
C language program fragments that define how the generated lexical analyzer reacts to regular expressions that it recognizes.

action table In Xtoolkit, a table that specifies the mapping of externally available procedure strings to the corresponding procedure implemented by the widget class. All widget class records contain an action table.

activate To point with the mouse pointer and double–click, successfully causing something to happen.

active (1) The windowpane in which the text cursor is currently positioned is said to be ”active.”
(2) One of the states in the lifetime of a transaction. This is the state during which it is accessing or modifying data.

active gateway A gateway that is treated like a network interface in that it is expected to exchange routing information. If it does not do so for a period of time, the route associated with the gateway is deleted. Contrast with *passive gateway* on page 2-171. See also *gateway* on page 2-99.

active grab In Enhanced X–Windows, a grab actually owned by the grabbing client. Contrast with *passive grab* on page 2-171. See also *button grabbing* on page 2-17 and *grab* on page 2-102.

actual parameter
The actual value passed to a routine. Contrast with *formal parameter* on page 2-94.

ACU See *automatic calling unit* on page 2-15 or *auto–call unit* on page 2-15.

adapter (1) A mechanism for connecting two unlike parts or machines.
(2) A printed circuit card that modifies the system unit to allow it to operate in a particular way. See also *communications adapter* on page 2-43 and *card* on page 2-31.

adapter code In X.25 communications, the X.25 Interface Co-Processor/2 Protocol Code, which controls the frame–level and packet–level communication processing.

add mode In addition and subtraction operations, a mode in which the decimal marker is placed at a predetermined location with respect to the last digit entered. In selection actions, a mode that allows the selection to be unaffected by keyboard navigation.

address (1) The telephone number that remote systems use to call the system.
(2) To refer to a device or an item of data by its address.
(3) In word processing, the location, identified by an address code, of a specific section of the recording medium or storage.
(4) In data communication, the unique code assigned to each device or workstation connected to a network. See also *network user address* on page 2-153.
(5) A numbering system used in network communications to identify a specific network or host with which to communicate. Addresses are often denoted in dotted decimal form. See also *presentation address* on page 2-182.

address extension
In X.25 communications, the called and calling address extensions are optional CCITT–specified facilities, available on networks that conform to the 1984 version of X.25. Synonymous with *extended address*.

address family See *protocol family* on page 2-189 .

address field The part of a packet containing addressing information. See also *packet* on page 2-166.

address list The list used by the **xtalk** command to associate users' names with network user addresses and other information, for the purpose of making outgoing X.25 calls without the caller having to know the addresses. There is one address list for the system and one for each user. See also *system address list* on page 2-234 and *user address list* on page 2-253.

Address Resolution Protocol (ARP)
One of the protocols provided by TCP/IP that dynamically maps between Internet addresses, Baseband Adapter addresses, X.25 addresses, and Token–Ring Adapter addresses on a local area network.

address space The code, stack, and data that is accessible by a process.

Address Translation Register (ATR)
A mechanism that translates real addresses to virtual addresses.

addressing (1) In data communications, the way that the sending or controlling station selects the station to which it is sending data.
(2) A means of identifying storage locations.
(3) Specifying an address or location within a file.
(4) The assignment of addresses to the instructions of a program. See also *selection* on page 2-214.

adjust The process of moving text to fit between the left and right margins.

ADK (Application Developer's Kit)
The component of the iFOR/LS system that is used by software developers to define and create licenses for software products.

administration environment
The part of the Monitor that facilitates system configuration and management.

ADU See *automatic dialing unit* on page 2-15.

Advanced Program–to–Program Communication (APPC)
A communications architecture that allows transaction programs to exchange information on a peer–to–peer basis. SNA LU 6.2 allows APPC architecture to operate on an SNA network.

advisory lock A type of lock that a process holds on a region of a file preventing any other process from locking the region or an overlapping region. See also *enforced lock* on page 2-80.

aggregate (1) An array, structure, or union.
(2) In programming languages, a structured collection of data objects for a data type.
(3) A transmitted carrier signal that consists of the 12 single–side bands being sent over the transmission circuit.

Common Desktop Environment
A visual user interface for the operating system consisting of icons and menus within an AIXwindows window. An iconic view of the file system providing users with windows, icons, and menus to perform program and file management tasks.

AIXwindows Environment
A software graphical user interface environment based on OSF/MOTIF consisting of the AIXwindows toolkit, graphics libraries, window manager, and desktop running on top of a compatible operating system.

AIXwindows Resource Manager (MRM)
A database management system that allows a user to effectively retrieve and assign information, such as specific values and other attributes, especially regarding applications that allow a user to select color, font, and other resource preferences. The MRM system consists of library subroutines that access the AIXwindows User Interface Language (UIL) at run time and create user interfaces. See also *AIXwindows User Interface Language (UIL)* on page 2-7.

AIXwindows Toolkit
An object–oriented collection of C language data structures and subroutines that supplement the Enhanced X-Windows toolkit and simplify the creation of interactive client application interfaces.

AIXwindows User Interface Definition (UID)
A file containing the definitions for all objects included in a particular user application. The UID is a subset of the AIXwindows User Interface Language (UIL) and functions similarly to the UIL for the particular application to which it belongs. See also *AIXwindows User Interface Language (UIL)* on page 2-7.

AIXwindows User Interface Language (UIL)
A compiled–specific language for describing the initial state of a user interface. UIL specifies the widgets, gadgets, and compound objects that make up the interface; it also identifies the subroutines to be called whenever the interface changes state as a result of user interaction. See also *AIXwindows User Interface Definition (UID)* on page 2-7.

alarm An audible signal at a workstation or printer that is used to get the operator's attention.

alert In SNA, an error message sent to the system services control point (SSCP) at the host system.

algorithm A finite set of well–defined rules for the solution of a problem in a finite number of steps; for example, a complete specification of a sequence of arithmetic operations for evaluating sine x to a given precision.

alias (1) An alternate name for a node or a file that can be used in place of the real name of the node or file.
(2) An alternate label for a data element or point in a computer program.
(3) An alternate name for a member of a partitioned data set. (4.) Unofficial name used for the network. Synonymous with *nickname*.
(4) An assumed or actual association between two data entities, or between a data entity and a pointer.

aliasing A compilation process that attempts to determine what aliases exist, so that optimization does not result in incorrect program results.

alignment The position in main storage of a fixed–length field, such as halfword or doubleword, on an integral boundary for that unit of information. For example, a word boundary is a storage address evenly divisible by four.

All Points Addressable (APA) displa
A display that allows each picture element (pel) to be individually addressed and displayed. An APA display permits the display of images that are not predefined in character boxes. Contrast with *character display* on page 2-34. See also *bitmapped display* on page 2-19.

allocate (1) To assign a resource, such as a disk file or a diskette file, to perform a specific task. Contrast with *deallocate* on page 2-63.
(2) A request to allocate a session between the local LU and a remote LU.
(3.) In NCS, to create a Remote Procedure Call (RPC) handle that identifies an object.

alphabetic character
A letter or other symbol, excluding digits, used in a language. Usually the uppercase and lowercase letters A through Z plus other special symbols (such as $ and _) allowed by a particular language. See also *alphanumeric character* on page 2-8.

alphanumeric character
Consisting of letters, numbers, and often other symbols, such as punctuation marks and mathematical symbols. See also *alphabetic character* on page 2-8.

alphanumeric set
Character set composed of uppercase and lowercase letters and numbers, but no symbols. See also *alphanumeric character* on page 2-8.

alternate cell (isolated cell)
An NCS cell that restricts access to a partial group of nodes in the network. This cell type is usually used to confine iFOR/LS license transactions to a particular set of nodes in the network.

alternate character set
A set of characters that includes some special characters, such as mathematical characters and Greek characters, and that is defined for some printers.

ALU Arithmetic and logical unit.

ambient light In three–dimensional graphics, light that reflects off one or more surfaces in the scene before arriving at the target surface. Ambient light is assumed to be nondirectional, and is reflected uniformly in all directions by the reflecting surface. In GL, ambient light is mocked up by use of ambient terms in the lighting equation, rather than actually computing the reflections.

ambiguous derivation
Derivation is ambiguous if a C++ class is derived from two or more base classes that have the same name.

American National Standard Code for Information Interchange (ASCII)
The code developed by ANSI for information interchange among data processing systems, data communications systems, and associated equipment. The ASCII character set consists of 7–bit control characters and symbolic characters.

American National Standards Institute (ANSI)
An organization sponsored by the Computer and Business Equipment Manufacturers Association through which accredited organizations create and maintain voluntary industry standards.

amp See *ampere* on page 2-9.

ampere (A or amp)
A unit of measurement for electric current that is equivalent to a flow of 1 coulomb per second, or to the current produced by 1 volt applied across a resistance of 1 ohm.

amplitude The size or magnitude of a voltage or current wave form.

ancestor In Enhanced X–Windows, a widget that has inferior widgets. In other words, the superior or predecessor of an inferior widget. If *W* is an inferior of *A*, then *A* is an ancestor of *W*.

annotation See *license annotation* on page 2-128.

anonymous union
A union in C++ without a class name. It must not be followed by a declarator.

ANSI See *American National Standards Institute* on page 2-9.

antialiasing Techniques used to smooth the ”jaggies” otherwise found on lines and polygon edges caused by scan conversion. Common techniques include adjusting pixel positions or setting pixel intensities according to the percent of pixel area coverage at each point.

a.out (1) An output file produced by default for certain commands. By default, this file is executable and contains information for the symbolic debug program.
(2) The object file format created by the **cc** command and expected by the **exec** subroutine.

APA See *All Points Addressable Display* on page 2-8.

APAR Authorized program analysis report. A report of a problem caused by a suspected defect in a current unaltered release of a program.

API See *application program interface* on page 2-10.

APL A programming language. A general–purpose language for diverse applications such as commercial data processing, system design, mathematical and scientific computation, database applications, and the teaching of mathematics and other subjects.

AppBuilder In CDE, a software application used for constructing a graphical user interface.

APPC See *Advanced Program–To–Program Communications* on page 2-6.

append (1) The action that causes data to be added to the end of existing data.
(2) In word processing, to attach a file to the end of another file.

applet A program, intended for delivery over the Internet, which can be included in an HTML page, just as an image can be included.

application (1) A program or group of programs that apply to a particular business area, such as Inventory Control or the Accounts Receivable application.
(2) Software coded by or for end users that performs a service or accomplishes work–related tasks.
(3) In AIXwindows and Enhanced X–Windows, the environment is modeled on a traditional client–server relationship in which compatible applications are considered clients of the Enhanced X-Windows server.

Application Developer's Kit
See *ADK* on page 2-6.

application development environment
The Monitor functions used to construct transactional applications.

application group
In CDE, an Application Manager folder that holds a specific software application or set of software applications.

application icon
See *action icon* on page 2-5.

application identifier (ID)
A unique identifier used to identify an application in the RPCs sent in a distributed environment.

Application Manager
In CDE, a window containing objects representing the system actions available to you.

application program
A program used to perform an application or part of an application.

application program interface (API)
(1) A set of run–time routines or system calls that allows an application program to use a particular service provided by either the operating system or another licensed program.
(2) The formally defined programming language interface that is between a system control program or a licensed program and the user of the program.

application programmer
A programmer who uses an API to produce an application.

application serve
In CDE, a host computer that provides access to a software application.

application shell
A subclass of top–level shell, this shell is used primarily for an application's top–level window.

application transaction program
(1) A program that performs an application or part of an application
(2) A program that connects and communicates with stations in a network, enabling users to perform application–oriented activities. See also *transaction program* on page 2-244 and *service transaction program* on page 2-216.

application window
A rectangular area that displays the graphics associated with a specific application. Application windows can be opened, closed, combined with other types of windows, moved, stacked, and otherwise manipulated through user interaction with a window manager.

apply (1) In journaling, to place after–images of records into a physical file member. The after–images are recorded as entries in a journal.
(2) When a service update is installed or *applied*, it enters the applied state and becomes the currently active version of the software. When an update is in the applied state, the previous version of the update is stored in a special save directory. This allows you to restore the previous version, if necessary, without having to reinstall it. Software that has been applied to the system can be either *committed* or *rejected*. The **installp –s** command can be used to get a list of applied products and updates that are available to be either committed or rejected. See also *commit* on page 2-43 and *reject* on page 2-198.

Apply In CDE, a choice that causes a selection (or group of selections) in a dialog box to take effect.

apply list file A file that contains an entry for each file to be restored during an installation or an update procedure.

Appointment Editor
In CDE, in Calendar, a window you use to schedule, change, or delete an appointment.

Appointment List
In CDE, a chronological list of appointments displayed by Calendar.

Arabic numerals
The 10 numerals used for depicting decimal numbers: the digits 0, 1, 2, 3, 4, 5, 6, 7, 8, and 9. No font is implied. See also *Roman numerals* on page 2-206.

archive (1) To store programs and data for safekeeping.
(2) A copy of one or more files or a copy of a database that is saved in case the original data is damaged or lost.

archive library A place where programs are stored for safekeeping.

arg pointer A pointer to a subroutine parameter argument.

argument (1) Numbers, letters, or words that expand or change the way a command works.
(2) A parameter passed between a calling routine and a called routine.
(3) An independent variable.

argument list A string of arguments.

argument string
An ordered list of parameters passed between programs or routines.

arithmetic constant
In a programming language, a constant of type integer, real, double precision, or complex. See also *arithmetic expression* on page 2-11 and *arithmetic operator* on page 2-11.

arithmetic expression
One or more arithmetic operators and arithmetic primaries, the evaluation of which produces a numeric value. An arithmetic expression can be an unsigned arithmetic constant, the name of an arithmetic constant, a reference to an arithmetic variable, array element, or function, or a combination of such primaries formed by using arithmetic operators and parentheses. See also *arithmetic constant* on page 2-11 and *arithmetic operator* on page 2-11.

arithmetic object
An integral object or objects having the float, double, or long double type. The C compiler also recognizes objects having the long double type as arithmetic objects.

arithmetic operator
A symbol that directs a compiler to perform an arithmetic operation. See also *arithmetic expression* on page 2-11 and *arithmetic constant* on page 2-11.

ARK (Administrator Runtime Kit)
In License Use Management, the run–time environment for licensed software products. It consists of the network license server daemon (**netlsd**) and its associated tools: **ls_admin**, **ls_rpt**, and **ls_stat**.

ARP See *Address Resolution Protocol* on page 2-6.

ARPA Advanced Research Projects Association.

ARPA RFC 822 Standard of ARPA internet text messages.

array (1) A structure that contains an ordered group of data elements. All elements in an array have the same data type.
(2) An arrangement of data in one or more dimensions, such as a list, table, or multidimensional arrangement of items.
(3) In programming languages, an aggregate that consists of data elements, with identical attributes, each of which may be uniquely referenced by subscripting.

array declarator
The part of a statement that describes an array used in a program unit. The description includes the name of the array, the number of dimensions, and the size of each dimension.

array element A single data item in an array.

array name The name of an ordered set of data items.

array padding In a multiple–array definition, the optimization of adding elements to each array, or adding small arrays between user–defined arrays, to improve the memory–access characteristics of the arrays.

arrow button A graphic control that simulates a push button with a directional arrow. People use the pointer and mouse to push the button and start some action that has an associated direction.

arrow keys In CDE, the four directional keys on a keyboard.

art tag In hypertext, a tag linking text to an artwork window.

artwork frame In Interleaf, the container for graphical data

ASCII See *American National Standard Code for Information Interchange* on page 2-8.

ASCII characters
The characters that make up the ASCII character set. The ASCII character set consists of 7–bit control characters and symbolic characters. See also *American National Standard Code for Information Interchange* on page 2-8.

ASCII flat file See *flat file* on page 2-92.

ASCIIZ format A string ending with a null character.

ASN.1 See *Abstract Syntax Notation One* on page 2-3.

aspect ratio The ratio of the height of a primitive to its width. A rectangle with a width of 10 inches and a height of 5 inches has an aspect ratio of 10/5 or 2.

assemble To translate an assembly program into a computer language. Assembling is usually accomplished by substituting the computer language operation codes for the assembly language operation code, and by substituting absolute addresses, immediate addresses, relocatable addresses, or virtual addresses for symbolic addresses.

assembler A computer program that converts assembly language instructions into object code. Synonymous with *assembly program*.

assembler language
A symbolic programming language in which the set of instructions includes the instructions of the machine and whose data structures correspond directly to the storage and registers of the machine.

assembly program
Synonym for *assembler* on page 2-12.

assertion See *program assertion* on page 2-187.

assignment compatible
Indicates whether the type of a value allows it to be assigned to a variable. See also *compatible types* on page 2-44.

assignment conversion
A change to the form of the right operand that makes the right operand have the same data type as the left operand.

assignment expression
In C language, an expression that assigns the value of the right operand expression to the left operand variable and has as its value the value of the right operand.

assignment statement
In programming languages, a statement that assigns the value of an expression to a variable.

associative transformation
An optimization that involves changing the order of an expression. An associative transformation yields mathematically identical results but not always bitwise identical results.

associativity The order for grouping operands with an operator (either left–to–right or right–to–left).

async See *asynchronous transmission* on page 2-13.

asynchronous Not synchronized in time. For example, input events are controlled by the user; the program can read them later.

asynchronous device
A device using data transmission in which transmission of a character or a block of characters can begin at any time, but in which the bits that represent the character or block have equal time duration.

asynchronous operation
An operation that does not of itself cause the process requesting the operation to be blocked from further use of the CPU. This implies that the process and the operation are running concurrently.

asynchronous terminal
A computer terminal using asynchronous signals to communicate with a host machine.

Asynchronous Terminal Emulation
program that provides emulation of a remote asynchronous terminal.

asynchronous transmission
Data transmission in which transmission of a character or block of characters can begin at any time, but in which the bits that represent the character or block have equal time duration. Contrast with *synchronous transmission* on page 2-219. See also *start–stop* on page 2-225.

ATE See Asynchronous Terminal Emulation on page 2-13.

ATM Asynchronous transfer mode. A cell–switching, connection–oriented technology. In ATM networks, end stations attach to the network using dedicated full duplex connections.

atom (1) A unique ID corresponding to a string name. Atoms are used to identify properties, types, and selections.
(2) A 32–bit number that represents a string value. See also *X Atom* on page 2-266.

atomic operation
An operation in which signals cannot occur between the operations of setting the masks and waiting for the signal.

ATR See *Address Translation Register* on page 2-6.

attachment (1) The physical connection to the network that makes it work.
(2) A type of resource that controls CPs, logical link control, and physical link control.
(3) In CDE, In Mailer, a data object within an electronic mail message that is displayed as an icon in the Attachments list. An attachment can be text, sound, or a graphic. Multiple messages can be added (attached) to a single electronic mail message.

attachment class
The attachment class specifies the mode in which a station will connect to the FDDI network. A station may be a dual attachment station (DAS) or a single attachment station (SAS).

attachment profile
Contains parameters that associate other defined profiles with the attachment of the LU to the network. These parameters also define the type of network being used.

attenuation (1) A decrease in magnitude of current, voltage, or power of a signal in transmission between points. It may be expressed in decibels or nepers.
(2) In 3D graphics, the fall off of light intensity with distance.

attribute (1) A characteristic or property of one or more objects or entities. For example, the attribute for a displayed field could be ”blinking.”
(2) In GL, a parameter that can affect the appearance of a drawing primitive. For instance, color is an attribute. If the color is set to ”RED,” it will remain red until changed, and everything that is drawn will be drawn in red. Other attributes include linestyle, linewidth, pattern, and font. For a list of attributes and pipeline options, see also *pipeline options* on page 2-176.
(3) In devices, a characteristic of a defined or configured device.

attribute file In system configuration, a text file that is organized into stanzas, each of which has a stanza name and a set of attribute definitions in the form of *Attribute=Value* pairs. Configuration files have the attribute file format.

attribute (of file)
Some portion of the information about a file that determines its access and organizational characteristics.

attribute pair See *attribute* on page 2-14, *value* on page 2-255, and *resource value* on page 2-203.

attribute value See *attribute* on page 2-14, *value* on page 2-255, and *resource value* on page 2-203.

audit events Occurrences on the system that may be security violations. These events cause an audit record to be written.

audit trail A collection of audit records.

auditing subsystem
A mechanism that lets an administrator detect potential or actual security violations in the system. Components of this subsystem detect audit events, log and collect audit events in a system audit trail, and process the audit trails.

authentication Verifying the identity of a user when the **login** or **su** command is given. For example, the operating–system method of authentication consists of checking the password entered by a user against the encrypted version of the password previously defined for that user. A secondary authentication method can be added for additional checks, such as verifying the identity of a user to a network.

authoring environment
The organization and equipment that allow the creation of hypertext documents.

authorization (1) The determination of a principal's permissions with respect to a protected object.
(2) The approval of a permission sought by a principal with respect to a protected object.

authorization protocol
A formal procedure for establishing the authorization of principals with respect to protected objects.

authorize (1) To grant to a user the right to communicate with or make use of a computer system or display station.
(2) To give a user either complete or restricted access to an object, resource, or function.

auto–answer The ability of a station to receive a call over a switched line without operator action.

auto–call The ability of a station to place a call over a switched line without operator action. Contrast with *manual call* on page 2-141.

auto–call unit (ACU)
In X.25 communications, a device that automatically makes and answers calls.

autodialer See *automatic dialing unit* on page 2-15.

autoexec A command or list of commands run at login time.

AUTOLOG A menu–driven utility program provided in the 3270 Host Connection Program 2.1 and 1.3.3 and used to create logon procedures.

automatic calling unit (ACU)
A device that allows a host to automatically dial the number of a remote device.

automatic dialing unit (ADU)
A device that can automatically generate dialing digits.

automatic scrolling
The scrolling action that takes place automatically when a cursor is moved to the border of a pane.

automatic variable
A variable allocated on entry to a routine and deallocated on the return. Contrast with *static variable* on page 2-226.

autonomous system
A group of networks and gateways for which one administrative authority has responsibility. An autonomous system can be small or very large.

autonumber A unique number associated with the prefix of each component in the Interleaf desktop publishing software. Because each autonumber has a unique value, these numbers are used to create references to their components elsewhere in the text. In hardcopy, an autonumber can be used to create a page reference that renumbers automatically as its associated component moves to a different page. In hypertext, autonumbers are used to create the hypertext links that jump to the corresponding component. For the ordered–list item component and the figure caption component, the autonumbers are visible and are used to number a series of these items in the proper order.

autonumber reference
A hardcopy or softcopy reference created from an autonumber. The autonumber reference automatically renumbers to reflect the status of its corresponding autonumber. In softcopy, an autonumber reference is associated with a hypertext link so that the link jumps to the corresponding autonumber. See also *autonumber* on page 2-16.

autopush A STREAMS mechanism that enables a prespecified list of modules to be pushed automatically onto the stream when a STREAMS device is opened.

available state The state a device is in when it is configured. The device status field in the Customized Devices Object Class in the ODM reflects whether a device is in the available state or not.

AZERTY keyboard
A keyboard in which the keys in the second–from–top row (row D) are labeled (from left to right): A, Z, E, R, T, Y, U, I, O, and P. See also *QWERTY keyboard* on page 2-193.

azimuthal angle
In GL, if a primitive is sitting on the ground, with its *z* coordinate straight up, the azimuthal viewing angle is the angle the observer makes with the *y* axis in the *x–y* plane. If the observer walks in a circle with the primitive at the center, the azimuthal angle is the only thing that varies.

azizo In Performance Toolbox, a tool used to analyze performance recordings.

B

back margin The margin of a page that is closest to the binding edge. Normally, this is the left margin of the recto page and the right margin of the verso page.

back up To copy information, usually onto diskette or tape, for safekeeping.

backdrop In CDE, the pattern that covers the *workspace background*.

backend The program that sends output to a particular device. There are two types of backends: friendly and unfriendly.

backend program
See *backend* on page 2-17.

backfacing polygon
In GL, a polygon whose vertices appear in clockwise order in screen space. If backface culling is enabled, such polygons are not drawn.

background (1) In multiprogramming, the conditions under which low–priority, noninteractive programs are run. Contrast with *foreground* on page 2-94. See also *program level* on page 2-187.
(2) In CDE, the underlying area of a window on which elements, such as buttons and lists, are displayed.

background activity
See *background process* on page 2-17.

background color
In AIXwindows and Enhanced X–Windows, the single electronic color assigned to the graphic field that appears behind the foreground elements inside the border of a displayed widget or gadget. Contrast with *foreground color* on page 2-94.

background process
(1) A process that does not require operator intervention but can be run by the computer while the workstation is used to do other work.
(2) A mode of program execution in which the shell does not wait for program completion before prompting the user for another command. Contrast with *foreground process* on page 2-94.

backing store The collection of off–screen, saved pixels maintained by the Enhanced X-Windows server.

backscrolling (1) Reversing the normal (top–to–bottom) direction of flow of paper through a printer.
(2) In reference to a video display, moving text through the viewing area from top to bottom.

Backtrack In CDE, in Help Manager, a button and Navigate menu item you use to follow links backward, in the reverse order they were traversed.

backup Pertaining to a system, device, file, or facility that can be used in the event of a malfunction or loss of data.

backup copy A copy, usually of a file or group of files, that is kept in case the original file or files are unintentionally changed or destroyed.

backup format When the **backup** command makes a copy of a file, it writes the file in this format. A file in this format must be restored by the **restore** command before it can be used.

backup format file
A file in backup format.

backup system
See *dump* on page 2-76 and *restore* on page 2-204.

bad block A portion of a disk that can never be used reliably.

bandwidth Data rate transfer in K bits, K bytes, M bits, and M bytes per second.

base address The beginning address for resolving symbolic references to locations in storage.

base address register
Synonym for *base register* on page 2-18.

base class A C++ class from which other classes are derived. A base class may itself be derived from another base class.

base line See *baseline* on page 2-18.

base name (1) The last element to the right of a full path name.
(2) A file name specified without its parent directories.
(3) In Ada language, a compilation unit name specified without its type qualifier of **lib/** or **sec/**.
(4) In CDE, The file name of an icon file minus the file–name suffixes for size (.l, .m, .s, .t) and type (.bm, .pm). For example, the base name of an icon file named myicon.m.pm is myicon.

base number The part of a self–check field from which the check digit is calculated.

Base Operating System (BOS) installation
The process of installing and configuring the minimum amount of software needed to bring a machine to the running state.

base permission
An access mode that is assigned to a file owner, file group, or others who want access to the file. Access modes include read (**r**) permission, write (**w**) permission, and execute/search (**x**) permission. See also *discretionary access control* on page 2-71.

base register A general purpose register that the programmer chooses to contain a base address. Synonym for *base address register*. See also *index* on page 2-113.

base scalar type
In Pascal, the type from which a set type or subrange type is derived. See also *subrange scalar type* on page 2-231.

baseband system
A communications system whereby information is encoded, modulated, and impressed on the transmission medium without shifting or altering the frequency of the information signal. At any point on the medium, only one information signal at a time is present.

baseline In a font, the imaginary line on which the bottom of each character is aligned.

BASIC (beginner's all–purpose symbolic instruction code)
(1) A programming language designed for interactive systems. Originally developed at Dartmouth College to encourage people to use computers for simple problem–solving operations.
(2) A high–level programming language with a small number of statements and a simple syntax. BASIC is designed to be easily learned and used and is widely used for interactive applications on microcomputers.

basic conversation
A connection between two transaction programs that allows them to exchange logical records that contain a 2–byte prefix that specifies the length of the record. LUs 1, 2, and 3 do not use the 2–byte prefix; however, LU 1, 2, and 3 conversations must be basic conversations. This conversation type is used by service transactions and LU 1, 2, and 3 application transaction programs. Contrast with *mapped conversation* on page 2-141.

Basic Encoding Rules (BER)
A set of rules used to encode ASN.1 values as strings of octets.

basic increment
The smallest unit of motion of which a device is capable.

Basic Input/Output System
See *BIOS* on page 2-18.

Basic Networking Utilities (BNU)
A group of programs and files, sometimes called the UNIX–to–UNIX Copy Program, that provides basic networking utilities, such as the **uucp** command. BNU includes a set of directories, files, programs, and commands that allow the user to communicate with a remote UNIX system over a dedicated line or a telephone line. See also *UNIX–to–UNIX Copy Program* on page 2-252 and *UUCP login ID* on page 2-254.

basic real constant
A string of decimal digits containing a decimal point and expressing a real value.

basis In GL, a curve or patch basis is a 4x4 matrix that controls the relationship between control points and the approximating spline. B–splines, Bezier curves, and Cardinal splines all differ in that they have different bases.

batch printing Queueing one or more documents to print in a separate job as a background process. The operator can type or revise additional documents at the same time. This is a background process. See also *background process* on page 2-17.

batch processing
A processing method in which one or more programs process records with little or no operator action. This is a background process. Contrast with *interactive processing* on page 2-118.

battery backup A power source that allows the processor drawer and other data storage devices to continue operating during a primary AC power source outage.

baud (1) The number of changes in signal levels, frequency, or phase per second on a communication channel. If each baud represents 1 bit of data, baud is the same as bits per second. Informally (as used by programmers) synonymous with ”bits per second.” Named for J. Baudot, 1845–1903, French inventor. However, it is possible for one signal change (1 baud) to equal more than 1 bit of data.
(2) A unit of signaling speed equal to the number of discrete conditions or signal events per second. For example, 1 baud equals one–half dot cycle per second in Morse code, 1 bit per second in a train of binary signals, and one 3–bit value per second in a train of signals that can each assume one of eight different states.
(3) In asynchronous transmission, the unit of modulation rate corresponding to one unit interval per second; for example, if the duration of the unit interval is 20 milliseconds, the modulation rate is 50 baud. See also *bps* on page 2-22.

baud rate In remote communications, the transmission rate that is synonymous with signal events. The baud rate is usually expressed in bits per second.

BCUG See *bilateral closed user group* on page 2-20.

BEL The bell character. A control character that activates an alarm or other attention devices when human attention is required. Synonym for *console bell*.

benchmark A program designed to test the relative performance of computers of different architectures, or of different implementations of an architecture. The combination of a rigorously specified workload and a method of quantifying the performance of a system when processing that workload. The performance metric is usually derived from the time required to process the workload.

BER See *Basic Encoding Rules* on page 2-19.

Bezier cubic curve
In GL, a cubic spline approximation to a set of four control points that passes through the first and fourth control points, and has a continuous slope where two spline segments meet. See also *parametric cubic curve* on page 2-169.

BGP Border Gateway Protocol.

bibliography A list of documents referred to within a document. For each document, the following is generally listed: the author, the document, notes about the document, and sometimes notes about its author.

bid In the contention form of invitation or selection, an attempt to gain control of a line to transmit data.

bidirectional See *shared port* on page 2-217.

big endian An attribute of data representation that reflects how multi–octet data are stored in memory. In big endian representation, the lowest addressed octet of a multi–octet data item is the most significant. See also *endian* on page 2-80 and *little endian* on page 2-131.

big endian order
The method of storage in which integer values are stored most significant byte first. See also *little endian order* on page 2-131.

big word In the vi editor, a contiguous set of alphanumeric characters bounded at the beginning and end by blank spaces, tabs, or new–line indicators. See also *small word* on page 2-220.

bilateral closed user group (BCUG)
In X.25 communications, an optional facility that allows calls to be made only between two designated DTEs. See also *closed user group* on page 2-39.

bin collection A method of collecting auditing data that writes audit records to a temporary bin file. After the data is processed by the **auditbin** daemon, records are written to an audit trail file for storage.

binary (1) Pertaining to a system of numbers to the base two. The binary digits are 0 and 1.
(2) Involving a choice of two conditions, such as on–off or yes–no.
(3) In ODM, a terminal descriptor type used to define a variable as a bit string that is not null–terminated. See also *terminal descriptor* on page 2-239.

binary constant
A constant that is made up of one or more binary digits.

binary digit Synonym for *bit* on page 2-22.

binary expression
An expression containing two operands and one operator.

binary file A file that contains codes that are not part of the ASCII character set. Binary files can utilize all 256 possible values for each byte in the file.

binary–image transfer
See *bit block transfer* on page 2-19.

binary operator
(1) A symbol representing an operation to be performed on two data items, arrays, or expressions. The four types of binary operators are numeric, character, logical, and relational. Contrast with *unary operator* on page 2-250.
(2) An arithmetic operator that has two terms. Synonym for *dyadic operator*.

binary search tree
A search structure in which, at each step of the search, the set of data elements is divided by two; some appropriate action is taken in the case of an odd number of data elements.

binary synchronous communication (BSC)
A form of telecommunication line control that uses a standard set of transmission control characters and control character sequences for binary synchronous transmission of binary–coded data between stations. Contrast with *synchronous data link control* on page 2-219.

bind (1) To associate a variable with an absolute address, identifier, or virtual address, or with a symbolic address or label in a program.
(2) In SNA, a request to activate a session between two logical units. See also *bind session* on page 2-17.
(3) For information on how to set a binding, see also *set* on page 2-217.

BIND See *bind session* on page 2-17.

bind image In SNA, the session parameters that the system services control point (SSCP) sends to the primary logical unit (PLU) and the PLU sends in the BIND request to the secondary logical unit (SLU); these parameters specify the proposed protocol options for an LU–LU session.

BIND password
One of the two communication security passwords. In an LU–LU session, it is the password that the system checks against the remote system to verify that the program to which the user is connected is the correct one. See also *node verification* on page 2-154 and *communications authority password* on page 2-44.

bind session (BIND)
In SNA products, a request to activate a session between two logical units. See also *bind* on page 2-18.

binder See *linkage editor* on page 2-130.

binding (1) A temporary association between a client and both an object and a server that exports an interface to the object. A binding is meaningful only to the program that sets it and is represented by a bound handle.
(2) In a multiprocessor context, constraining a thread to a specific physical processor to gain the benefit of processor affinity.

binding edge The edge of a page, perforated for loose binding, to be bound, stapled, or drilled.

BIOS (Basic Input/Output System)
In the Personal Computer AT, microcode that controls basic hardware operations such as interactions with diskette drives, fixed–disk drives, and the keyboard.

bis Describes a secondary recommendation that is an alternative to a primary recommendation.

BIST Built–in self–test.

bit Either of the binary digits 0 or 1 used in computers to store information. Synonymous with *binary digit*. See also *byte* on page 2-17.

bit block transfer (BLT)
The movement of a binary image (bitmap or pixmap) by specifying the lower–left and upper–right corners of the image and the destination address.

bit BLT See *bit block transfer* on page 2-19.

bit clocking In an EIA–232C or D interface, the field that indicates which piece of equipment, either the modem (DCE) or the computer (DTE), provides the clock signal for synchronized data transactions.

bit field A member of a structure or union that contains 1 or more named bits.

bit gravity In Enhanced X–Windows, the attraction of window contents for a location in a window. When a window is resized, its contents can be relocated. The server can be requested to relocate the previous contents to a region of the window. See also *gravity* on page 2-103.

bit planes In computer graphics, a bitplane supplies one bit of color information per pixel on the display. Thus, an eight bitplane system allows 2 to the eighth power different colors to be displayed at each pixel.

bit rate The speed at which bits are transmitted, usually expressed in bits per second.

bit scattering A technique for mapping memory bits to ensure that a minimum number of bits in a memory word are stored in a single DRAM. With a minimum number of bits, the ECC is better able to detect and correct errors caused by a bad DRAM.

bitmap A pixmap with a depth of one bit plane.

bitmap file The file containing the height and width instructions for creating a bitmap.

bitmapped display
A display with a display adapter that has a hardware representation of each separately addressable point on the display. The hardware representation can be processor memory or adapter memory. See also *All Points Addressable Display* on page 2-8.

bits per character
The number of bits in a data character.

blank common In FORTRAN, an unnamed common block.

blind folio A document in which the pages of the document are counted but not numbered. See also *folio* on page 2-93, *dropped folio* on page 2-76, and *expressed folio* on page 2-86.

blit Bit block transfer.

block (1) A group of contiguous records recorded or processed as a unit. Blocks are separated by interblock gaps and each block may contain one or more records.
(2) In data communications, a group of records that is recorded, processed, or sent as a unit.
(3) In programming languages, a compound statement that coincides with the scope of at least one of the declarations contained within it. A block may also specify storage allocation or segment programs for other purposes.

block data subprogram
In FORTRAN, a subprogram headed by a BLOCK DATA statement that is used to initialize variables in named common blocks.

block device (1) One of the types of files in the file system, described by an i–node.
(2) A device that is accessed by means of a device driver.

block file A file listing the usage of blocks on a disk. See also *special file* on page 2-223 and *character special file* on page 2-34.

block input/output communication area (BIOCA)
A block of storage in the kernel address space that is used to communicate with a block I/O subsystem.

block I/O Input/output operations on blocks of data stored in random locations.

block special file
A special file for a block device. The file provides access to an input or output device that uses in–core buffers and is capable of supporting a file system. See also *character special file* on page 2-34.

block statement
Any number of data definitions, declarations, and statements that appear between the symbols { (left brace) and } (right brace). The C language, for example, reads a block statement as a single C language statement. In Ada language, a block statement is a single statement that may contain a sequence of statements. It may also include a declarative part, and exception handlers; their effects are local to the block statement. See also *statement* on page 2-225.

blocking An optimization that involves changing the access order of loops that access large arrays, so that each array element is accessed as infrequently as possible.

blocking call A call in which a caller is suspended until a called procedure completes.

blocking factor The number of iterations of an inner loop that are executed for each pass of a corresponding blocking loop.

BLT See *bit block transfer* on page 2-19.

BNC A connector used with some coaxial cables.

BNU See *Basic Networking Utilities* on page 2-19.

body (1) On a printed page, the portion of the page that contains the main text, excluding the margins, headers, footers, and footnotes.
(2) In a book, the portion between the front matter and the back matter.
(3) In Ada language, a body defines the execution of a subprogram, package, or task. A body stub is a form of body that indicates that this execution is defined in a separately compiled subunit.

boldface A heavy–faced type, generally a heavier version of a regular text font. Also, the process of printing in this type. See also *double–strike* on page 2-75 and *emphasized* on page 2-79.

Boolean A binary numbering system named after mathematician George Boole in which zero and one are the only two values that can be returned. Traditionally, a value of zero represents **FALSE** while a value of one represents **TRUE**. A value of TRUE or FALSE, where TRUE=1 AND FALSE=0.

boot See *initial program load* on page 2-114.

boot block In a file system, the first block where the bootstrap program resides. See also *bootstrap block* on page 2-22.

boot device The device that assigns the fixed disk within the root volume group (rootvg) that will contain the startup (boot) image.

boot image An image containing the kernel, file systems, libraries, and programs. The boot image is loaded after the machine is turned on or reset and brings it to a running state.

boot processing
The type of processing that occurs when a boot image is loaded. The type of boot device (disk, tape, or network) determines the type of boot processing that occurs. Boot processing over the network brings a machine to the running state, the BOS installing state, or the diagnostic state depending on the configuration file.

bootstrap A small program that loads larger programs during system initialization.

bootstrap block
Synonym for *boot block* on page 2-18.

border A visual boundary that separates a displayed object from everything else on a screen.

BOS (Base Operating System)
The collection of programs that controls the resources and the operations of the computer system.

bosinst.data The file that controls the actions of the BOS installation program.

bottleneck An area of code within a program that uses CPU resources inefficiently and therefore causes unnecessary delays.

bottom shadow
In AIXwindows, a narrow band of a dark color across the bottom of a rectangular graphical object (a widget or gadget) that creates a three–dimensional appearance when the object is manipulated.

boundary alignment
The position in main storage of a fixed–length field, such as halfword or doubleword, on an integral boundary for that unit of information. For example, a word boundary is a storage address evenly divisible by four.

bounding box (1) In computer graphics, the bounding box of a character is the smallest rectangle that encloses the shape at the *x*, *y* origin.
(2) In GL, a two–dimensional rectangle that bounds a primitive. A bounding box can be used to determine whether the primitive lies inside a clipping region. See also *character cell* on page 2-33 and *clipping* on page 2-38.

bounds violation
An attempt to access an array using an index or pointer that references storage outside of the array.

box A line enclosure around text or a table.

bpi Bits per inch, a measure of linear density for storage products.

bps Bits per second. In serial transmission, the instantaneous bit speed with which a device or channel transmits a character. See also *baud* on page 2-19.

braces (1) The characters { (left brace) and } (right brace), also known as ”curly brackets.” These ASCII characters are primary symbols in the C programming language in which the operating system is written.
(2) In Pascal, comment delimiters.

bracket read lock
A shared lock that is obtained, held only for the duration of the current operation, and then immediately dropped. This lock mode ensures that only committed data is read. It does not guarantee that successive reads of the data will yield the same value, because other transactions can immediately modify the value after the read operation is complete and the lock is dropped.

brackets (1) The ASCII characters [(left bracket) and] (right bracket), also known as ”square brackets.” (This usage does not follow the British usage of ”brackets” to mean parentheses, the characters that enclose this sentence.)
(2) In SNA, one or more chains of request units and their responses, which are exchanged between two LU–LU half–sessions and represent a transaction between them. A bracket must be completed before another bracket can be started. Examples of brackets are database inquiries and replies, update transactions, and remote–job–entry output sequences to workstations.

branch In a computer program, an instruction that selects one of a number of alternative sets of instructions. A conditional branch occurs only when a specified condition is met. An instruction that selects one of a number of alternative sets of instructions.

branch processing unit
A processing unit that processes branch instructions and dispatches fixed– and floating–point instructions to the fixed–point unit and floating–point unit.

break condition
In the TTY subsystem, a character framing error in which the data is all zeros.

break signal A signal sent over a remote connection to interrupt current activity on the remote system.

break statement
A C language control statement that contains the keyword **break** and a semicolon.

break value In allocating data segment space, the address of the first location beyond the current end of the data segment.

breakpoint A place in a computer program, usually specified by an instruction, where execution may be interrupted by external intervention or by a monitor program.

bridge (1) In the connection of local loops, channels, or rings, the equipment and techniques used to match circuits and facilitate accurate data transmission.
(2) A functional unit that connects two local area networks (LANs) that use the same logical link control (LLC) procedure but may use different medium access control (MAC) procedures. Contrast with *gateway* on page 2-99.

broadband Transmission media and techniques that use a broad frequency range, divided into sub–bands of narrower frequency, so that different kinds of transmission can occur at the same time.

broadband channel
A data transmission channel 6 MHz wide. Synonym for *wide band channel* on page 2-261.

broadcast Simultaneous transmission of data to more than one destination.

broadcast topology
The topology in which all stations are connected in parallel with the medium and are capable of concurrently receiving a signal transmitted by any other station connected to the medium.

broken pipe message
A message that occurs if the pipe becomes unsynchronized.

broker In NCS, a server that manages information about objects and interfaces to the objects. A program that wishes to become the client of an interface can use a broker to obtain information about servers that export the interface. See also *Location Broker* on page 2-133.

browse (1) A function available when you select the List of Books button at the bottom of a navigation window. You can move forward and backward through an online book in the same way you can move through a book in hardcopy.
(2) In the **xtalk** and **xroute** commands, to view the details of an entry in the routing list or address list.

BSC See *Binary Synchronous Communication* on page 2-17.

BSD Berkeley Software Distribution.

B–spline cubic curve
In computer graphics, a cubic spline approximation to a set of four control points having the property that slope and curvature are continuous across sets of control points. See also *parametric cubic curve* on page 2-169.

BTU (1) British thermal unit.
(2) In SNA, basic transmission unit.

bucket One or more fields that accumulate the result of an operation.

bucket counts The number of elements greater than or equal to the lowest limit and less than or equal to the higher limit.

buffer (1) A temporary storage unit, especially one that accepts information at one rate and delivers it at another rate.
(2) An adjustable memory storage space, temporarily reserved for performing input or output, into which data is read or from which data is written.

buffer length The maximum length of a data segment that can be stored in a given buffer.

bug An error in a program or a logic problem in the intent of the program.

BUID See *bus unit identification* on page 2-22.

build code The program that runs the build process to convert source files into a hypertext database. The build code defines which items can be built into hypertext and which cannot. For example, an illegal font does not build because it is not included in the build code.

built–in function
A function known to the compiler, whose code is incorporated directly into a program module rather than referenced through a library call.

bulletin board A graphic object that simulates a real–life bulletin board in that it displays text and graphic information in the form of messages to the user from client applications that are currently running.

bundle A collection of software products available for installation.

burst pages On continuous–form paper, pages of output that can be separated at the perforations.

bus (1) In a processor, a physical facility on which data is transferred to all destinations but from which only addressed destinations can read in accordance with appropriate conventions.
(2) A computer configuration in which processors are interconnected in series.
(3) One or more conductors that transmit signals or power.

bus–in A unidirectional data bus that is part of the System 360/370 parallel–I/O interface. This bus passes data from the system unit to the host.

bus–out A unidirectional data bus that is part of the System 360/370 parallel–I/O interface. This bus passes data from the host to the system unit.

bus unit identification
A field in the I/O Segment register that is decode to select the IOCC for the particular hardware implementation.

busy light In CDE, a control that blinks when a Front Panel action has been invoked prior to the appearance of a window or when reloading actions.

button (1) A word or picture on the screen that can be selected. Once selected and activated, a button begins an action in the same manner that pressing a key on the keyboard can begin an action.
(2) Buttons include those on the keyboard, mouse, lightpen, or buttons on the dial and button box.

button grabbing
Enacting an active grab using a mouse button. See also *grab* on page 2-102, *pointer grabbing* on page 2-177, *key grabbing* on page 2-124, *passive grab* on page 2-171, and *active grab* on page 2-5.

button press The initiation of a function by pressing a mouse button.

byte (1) The amount of storage required to represent 1 character; a byte is 8 bits.
(2) A binary character operated on as a unit and usually shorter than one word.
(3) A string that consists of a certain number of bits (usually 8) treated as a unit, and that represents a character.
(4) A group of 8 adjacent binary digits representing 1 EBCDIC character.
(5) In X.25, a byte is called an *octet.* See also *bit* on page 2-22.

byte boundary Memory addressing based on 8–bit intervals. Each memory location contains an 8–bit value that can range from 0 to 255 in decimal notation.

byte constant In FORTRAN, a named constant that is of type byte.

byte order In Enhanced X–Windows, the order of bytes as defined by the server for pixmap or bitmap data. Clients with different native byte ordering must swap bytes as necessary.

byte type In FORTRAN, a binary character operated on as a unit and usually shorter than a computer word.

C

C Celsius or country.

C interface The interface, defined at a level that depends on the variant of C standardized by ANSI.

C language A general–purpose programming language that is the primary language of the operating system.

C library A system library that contains common C language subroutines for file access, string operators, character operations, memory allocation, and other functions.

C++ language A programming language based on the C language and developed by Bjarne Stroustrup. C++ defines data types called classes. Classes provide data abstraction and are composed of data (data members) and operations that operate on that data (member functions). C++ also implements the object–oriented programming concepts of inheritance through class derivation and polymorphism through virtual functions and overloading.

C++ language statement
A C++ language statement contains zero or more expressions. All C++ language statements, except block statements, end with a ; (semicolon) symbol. A block statement begins with a { (left brace) symbol, ends with a } (right brace) symbol, and contains any number of statements.

C++ library A system library that contains common C++ language subroutines for file access, memory allocation, and other functions.

cable The physical media for transmitting signals; includes copper conductors and optical fibers.

cache (1) A high–speed buffer storage that contains frequently accessed data or instructions and that can deliver that data or instructions faster than the storage medium on which that information usually resides. A cache is used to reduce access time. See also *write back cache* on page 2-265.
(2) To place, hide, or store in a cache. A technique (usually software–based) whereby high–speed storage that is not immediately required for other purposes is used to retain data that has been loaded into it once, in the hope that another request for the data will occur before the high–speed storage must be reassigned.

cache coherency
The need to ensure that multiple threads on multiple processors changing a single cache line do not create inconsistent versions of the cache line in the different caches.

cache hit A processor storage reference that is satisfied by information from a cache.

cache line The cache component that is normally loaded, stored, and interrogated during cache lookup. See also *set associativity* on page 2-217.

cache line tag The information kept with each cache line to identify the part of virtual storage it contains.

cache lookup The process of determining whether or not a cache contains the information necessary to satisfy a storage reference. A defined set of bits in the address being referenced identifies the line or lines to be interrogated.

cache miss (1) A delay that occurs when the CPU references data or instructions that are not already in the data cache or instruction cache.
(2) A processor storage reference that cannot be satisfied from a cache and therefore requires a RAM access.

CAD/CAM Computer–Aided Design/Computer–Aided Manufacturing.

CAI See *computer–aided instruction* on page 2-47.

Calculator In CDE, a software application that mimics the function of a handheld calculator.

Calendar In CDE, a software application that enables you to schedule appointments and create To Do lists.

Calendar view In CDE, the day, week, month, or year displayed in the main Calendar window.

call (1) To activate a program or procedure, usually by specifying the entry conditions and jumping to an entry point. See also *load* on page 2-131.
(2) In data communications, the action necessary in making a connection between two stations on a switched line.
(3) In X.25 communications, a conversation between two users.

call–accepted packet
In X.25 communications, a call supervision packet transmitted by a called DTE to inform the DCE of the acceptance of the call. See also *packet* on page 2-166.

call–back A characteristic of the UUCP file USERFILE that tells a remote system whether the local system it tries to access will call back to check its identity.

call collision See *collision* on page 2-40.

call–connected packet
In X.25 communications, a call supervision packet transmitted by a DCE to inform the calling DTE of the complete establishment of the call. See also *packet* on page 2-166.

call redirection notification
In X.25 communications, an optional CCITT–specified facility that informs the caller that the call has been redirected to another DTE.

call request (CRQ)
A signal sent by a computer to request from data communications equipment that a communications connection be established with another computer in the network.

call–request packet
In X.25 communications, a call supervision packet transmitted by a DTE to ask for a call establishment through the network. See also *packet* on page 2-166.

call user data (CUD)
In X.25 communications, data optionally included in the call–request packet by the user application.

callback A procedure that is called if and when certain specified conditions are met. This is accomplished by specifying the procedure in a callback list. Synonymous with *callback function* on page 2-28. See also *callback routines* on page 2-29.

callback function
Synonym for *callback* on page 2-29.

callback list (1) A list of procedures that are called if and when certain specified conditions are met.
(2) In AIXwindows, individual widgets can define callback lists as required.

callback reason
The conditions that, if met, result in a callback procedure being called.

callback routine address
The internal address of a given callback routine.

callback routines
Procedures that are called if and when certain specified conditions are met. See also *callback* on page 2-29.

called In X.25 communications, an adjective applied to the location or user to which a call is made.

called address extension
See *address extension* on page 2-6.

called DLS user
The data link service (DLS) user in connection mode that processes requests for connections from other DLS users.

called line address modified notification
In X.25 communications, an optional CCITT–specified facility.

called NS user A network service (NS) with whom a calling NS user wants to establish a network connection.

called subaddress
See *subaddress* on page 2-229.

calling In X.25 communications, an adjective applied to the location or user that makes a call.

calling address
See *network user address* on page 2-153.

calling address extension
See *address extension* on page 2-6.

calling conventions
Specified ways for routines and subroutines to exchange data with each other.

calling DLS user
The data link service (DLS) user in connection mode that initiates the establishment of a data link connection.

calling NS user
A network services (NS) user that initiates a network connection.

calling tree The tree of control of a program. The main procedure is the calling tree's trunk; any procedure referenced by the main procedure branches off from the trunk, and any procedure referenced by another procedure is a branch off that branch. A branch that does not reference other procedures is known as a leaf.

callout A kernel parameter that establishes the maximum number of scheduled activities that can be pending simultaneously.

callout table A kernel table that keeps track of all sleeping processes and the channel on which each is waiting.

cancel To end a task before it is completed

Cancel In CDE, a push button that removes a window without applying any changes made in that window

canonical processing
Processing that occurs according to a defined set of rules. This is the style of input that is typically used by the shell and simple commands.

caps (1) Capital letters, an uppercase font.
(2) A printing style that uses two type sizes of a single uppercase font. The smaller size is used instead of a lowercase font.

caption Text associated with, and describing, a table or figure.

capture To digitize an image into the video memory of the M–Video Capture Adapter.

capture file A file used by a communications program to capture, or record, data coming in over a connection to a remote system or device.

capture key A toggle control key that starts or stops the process of saving the data displayed on the screen during an active connection.

card An electronic circuit board that is plugged into a slot in the system unit. See also *adapter* on page 2-5.

cardinal spline cubic curve
In computer graphics, a cubic spline whose endpoints are the second and third of four control points. A series of cardinal splines have a continuous slope and pass through all but the first and last control points. See also *parametric cubic curve* on page 2-169.

carriage return (1) In text data, the action that indicates to continue printing at the left margin of the next line. A carriage return is equivalent to the carriage return of a typewriter.
(2) A keystroke generally indicating the end of a command line.

carrier In data communication, a continuous frequency that can be modulated or impressed with an information–carrying signal.

carrier sense multiple access with collision detection (CSMA/CD)
The generic term for a class of medium access procedures that allows multiple stations to access the medium at will, without explicit prior coordination, and avoids contention by way of carrier sense and deference. Contention is resolved by way of collision detection and transmission.

carrier signal A signal with a constant frequency that can be modulated to carry a data signal.

cascade button
In AIXwindows, a rectangular graphic control that can be made to appear from behind another graphic control to provide an additional option or range of options.

cascading menu
A submenu of related choices that is invoked when the parent item, is selected. Usually, a choice that offers a cascading menu is designated by an arrow to the right of the choice. Similar to a *context line* on page 2-52.

case clause In a C For AIX switch statement, a case label followed by any number of statements.

CASE Computer Assisted Software Engineering. A set of tools or programs to help develop complex applications.

CASE label In Pascal, a value or range of values that comes before a statement in a CASE statement branch. When the selector is evaluated to the value of a CASE label, the statement following the case label is processed.

case label The word case followed by a constant expression and a colon. When the selector is evaluated to the value of the constant expression, the statements following the case label are processed.

case–sensitive Able to distinguish between uppercase and lowercase letters.

cast In C language, an expression that converts the value of the operand to a specified scalar data type (the operator).

C.A.T Computer–assisted typesetting.

catalog A set of predefined components and attributes used to create Interleaf documents.

catastrophic cancellation
A programming error in which values with very large negative exponents are added or multiplied together until a zero value is produced, which is then propagated into successive computations.

catch block A block in a C++ program that receives control when an exception matching its argument is thrown. Each catch block is associated with a try block.

cathode ray tube (CRT)
A vacuum tube in which a beam of electrons can be moved to draw lines or to form characters or symbols on its luminescent screen.

cause code In X.25 communications, a 1–byte code included in clear– and reset–indication packets that indicates the origin of the packet and the reason for sending it. Synonymous with *clear cause*. See also *diagnostic code* on page 2-68.

CBEMA Computer and Business Equipment Manufacturers Association.

CCITT Comite Consultatif International Telegraphique et Telephonique. See also *Consultative Committee on International Telegraphy and Telephone* on page 2-51.

CCW Channel control word. This is a defined control encoding that is used to control the operations of I/O units on the System 360/370 channel.

CD Carrier detect. See also *DCD* on page 2-63.

CDE An acronym for Common Desktop Environment, a graphical user interface running on UNIX.

CDLI Common Data Link Interface. Device drivers that interface with kernel services to provide support for sockets and STREAMS interfaces.

CD–ROM High–capacity read–only memory in the form of an optically read compact disc.

CDS Cell Directory Service.

CDS–defined attribute
A standard attribute that CDS associates with names. A specific CDS–defined attribute has the same meaning no matter what type of entry (clearinghouse, directory, object) it is associated with. However, different types of entries can have different CDS–defined attributes. For example, every CDS name has the CDS–defined attributes of Creation Timestamp (**CDS_CTS**), Update Timestamp (**CDS_UTS),** and Access Control Set (**CDS_ACS**). In addition to those attributes, a soft link has unique CDS–defined attributes containing its expiration time and the name it points to.

CDS directory A logical unit for storing entries under one name (the directory name) in a CDS namespace. In addition to object entries, a directory can contain soft links and child pointers. You can copy, delete, and control access to a directory. Each physical instance of a directory is called a replica.

CDSPI Cell Directory Service Portable Interface.

CDSTL See *connect data set to line* on page 2-49.

CEC Central Electronics Complex.

cell The rectangular juncture of a horizontal row and a vertical column. Examples include the cells of an electronic spreadsheet and the cells utilized by an **XmRowColumn** widget in an AIXwindows graphic interface. In Interleaf, a table cell is actually a modified frame that can contain data.

cell–relative name
See *local name* on page 2-132.

centered dot A heavy bullet used as a mark to set off a paragraph or list item.

central processing unit (CPU)
The part of a computer that includes the circuits that control the interpretation and running of instructions.

CFM Cubic feet per minute.

CGA Color Graphics Adapter.

CGM See *Computer Graphics Metafile* on page 2-47.

change bit A bit in each page–frame table entry that denotes the corresponding page has been written to since the last time the operating system cleared the page.

channel (1) A path along which signals or data passes.
(2) The portion of a storage medium that is accessible to a given reading or writing station.
(3) In data communication, a means of one–way transmission.
(4) A functional unit, controlled by a host computer, that handles the transfer of data between processor storage and local peripheral equipment.
(5) The system element that controls a single channel path, whose mode of operation depends on the type of hardware to which it is attached.
(6) A device connecting the processor to input and output devices.
(7) One of 32 bits in a table used to represent which event classes are active or inactive. The most significant bit is called channel 0 and the least significant bit is called channel 31. See also *logical channel* on page 2-134.

channel control word
See *CCW* on page 2-32.

channel ID A channel identification passed back from a multiplexed device to the file system as a result of calling the **ddmpx** entry point.

channel number
A number that identifies the path by which data is transferred between a particular input or output device and the processor of the computer. The major device, minor device, and channel numbers uniquely identify a hardware device.

channel path A single interface attaching one or more control units.

channel–path Identifier (CHPID)
In a System/390 channel subsystem, a value assigned to each installed channel path of the system that uniquely identifies that path to the system.

char In Object Database Manager, a terminal descriptor used to define a variable as a fixed–length, null–terminated string. See also *terminal descriptor* on page 2-239.

char specifier The keywords **char** and **unsigned char**, which describe the type of data a variable represents.

character A letter, digit, or other symbol.

character cell The physical width and height in pels of a font. See also *bounding box* on page 2-17.

character class
(1) Ranges of characters that match a single character in the input stream.
(2) A set of characters enclosed in sequence, or square [], brackets.

character constant
(1) A constant value whose data attribute is character.
(2) In programming languages, a character or an escape sequence enclosed in single quotation marks.

character data indexing
An input/output optimization that reduces I/O access time by searching for a string in a small index file, and using the obtained index to find a record in the main data file.

character delete
In text data, the action that erases the character at the current cursor location and moves any trailing text one character position to the left.

character device
A device that handles data one character at a time. See also *character special file* on page 2-34.

character display
A display that uses a character generator to display predefined character boxes of images (characters) on the screen. This kind of display cannot address the screen any less than one character box at a time. Contrast with *All Points Addressable display* on page 2-8.

character expression
A character constant or variable, character array element, character substring, character–valued function reference, or sequence of the preceding separated by the concatenation operator, with optional parentheses.

character graphics
(1) The visual representation of a character, defined by toned or intoned picture elements (pels).
(2) Graphics that are composed of symbols printed in a monospace font. Some symbols are standalone; others are intended for assembling larger figures.

character key (1) A keyboard key that allows the user to enter the character shown on the key. See also *function keys* on page 2-98.
(2) In word processing, a control used to process text one character at a time.

character literal
A symbol, quantity, or constant in a source program that is itself data, rather than a reference to data. Contrast with *numeric literal* on page 2-157.

character position
On a display, the location of a character.

character set A group of characters used for a specific reason; for example, the set of characters a printer can print or a keyboard can support.

character special file
A special file that provides access to an input or output device. The character interface is used for devices that do not use block I/O. See also *character device* on page 2-34, *block file* on page 2-20, *special file* on page 2-223 *,* and *block special file* on page 2-20.

character string
A sequence of consecutive characters. In the C programming language, a string must be null–terminated.

character substring
A contiguous portion of a character string.

character translation
In international character support, the **dd** command and various conversion subroutines that translate between extended characters and ASCII escape strings to preserve unique character information.

character type A data type that consists of alphanumeric characters. See also *data type* on page 2-62.

character variable
In the C language, a data object whose value can be changed during the running of a program and whose data type is char **or** unsigned char**.**

characteristic attribute
A type of attribute that reflects or affects the behavior of a software entity. You generally can set or change characteristic attributes.

charging requesting service
In X.25 communications, an optional facility that specifies that charging information (segment count data, monetary unit data, or call duration data) is required.

chat script In remote communications, a list of expect–send sequences that a modem uses to establish a communication link with another modem. See also *handshaking* on page 2-105 and *expect–send sequence* on page 2-85.

checkbox A small square box that can be turned on or off to indicate the state of an option.

checkpoint A snapshot of the current state of the recoverable data being used by an application. Checkpoints are used to capture the state of recoverable data between backups, thus minimizing the time involved in restarting systems which use that data by providing a more recent image of that data.

checksum (1) The sum of a group of data associated with the group and used for checking purposes.
(2) On a diskette, data written in a section for error detection purposes.

child (1) Pertaining to a secured resource, either a file or library, that uses the user list of a parent resource. A child resource can have only one parent resource.
(2) In the operating system, a child is a process, started by a parent process, that shares the resources of the parent process. Contrast with *parent* on page 2-170.
(3) In Enhanced X-Windows and AIXwindows, a first–level subwindow. A widget managed by another widget is said to be the child of the managing parent widget. For example, **Composite** widgets typically manage the **Primitive** children widgets attached to them. The parent widget typically controls the placement of the child as well as when and how it is mapped.

child device A hierarchical location term. It indicates what can be connected to a parent device. For example, an SCSI disk can be a child device of an SCSI adapter.

child gadget A windowless child widget. See also *child widget* on page 2-36.

child process In the operating system, a process, started by a parent process, that shares the resources of the parent process.

child resource Pertaining to a secured resource, either a file or library, that uses the user list of a parent resource. A child resource can have only one parent resource.

child widget In AIXwindows and Enhanced X–Windows, a widget managed by another widget is said to be the child of the managing parent widget. For example, **Composite** widgets typically manage the **Primitive** children widgets attached to them. The parent widget typically controls the placement of the child as well as when and how it is mapped. When a parent widget is deleted, all the children controlled by that parent are automatically deleted as well. See also *child gadget* on page 2-35.

children Plural of *child* on page 2-35.

children spacing
In AIXwindows and Enhanced X–Windows, widgets managed by another widget are said to be the children of the managing parent widget. The parent widget typically controls the physical spacing and placement of the children within the border of the parent.

choice An option in a pop–up or menu used to influence the operation of the system.

chord In graphics, a short line segment whose end points lie on a circle. Chords are a means for producing a circular image from straight lines. The higher the number of chords per circle, the smoother the circular image.

CHPID Channel path identifier.

CICS See *Customer Information Control System* on page 2-58.

CID See *connection identifier* on page 2-50.

ciphertext The output of an encryption function. Encryption transforms plaintext into ciphertext.

circuit See *virtual circuit* on page 2-258.

circuit switching
A process that, on demand, connects two or more data terminal equipments (DTEs) and permits the exclusive use of a data circuit between them until the connection is released. Synonym for *line switching*. See also *packet switching* on page 2-167.

C–ISAM C–language Indexed Sequential Access Method.

CLA See *communications line adapter* on page 2-43.

class (1) Pertains to the I/O characteristics of a device. In this operating system, devices are classified as block or character.
(2) In Enhanced X–Windows, a general group to which a specific object belongs. See also *widget class* on page 2-262 and *class record* on page 2-37.
(3) In AIXwindows, an object–oriented data structure containing generalized information about a group of similar graphical objects known as widgets or gadgets. Each class of graphical objects inherits some or all of the appearance characteristics and behavior characteristics of the classes that precede it in the object hierarchy.
(4) A C++ class is a user–defined data type. A class data type can contain both data representations (data members) and functions (member functions). See also *object class* on page 2-158.
(5) In Workload Management, a collection of processes (and their associated threads) that have a single set of resource limitation values and target shares applied to them.

class assignment rule
Indicates what set of process attribute values determine which class a process is assigned to.

class key One of the C++ keywords: **class**, **struct**, and **union**.

class library A collection of C++ classes.

class member operators
Used to access C++ class members through class objects or pointers to class objects. They are ., –>, .*, and –>*.

class name (1) For widgets and gadgets, the name in the code corresponding to the resource database containing the generic properties for all objects in the class.
(2) A unique identifier of a C++ class type that becomes a reserved word within its scope.

class record A particular widget record that contains the data objects pertaining to the class of any given widget. See also *record* on page 2-196, *widget record* on page 2-262, and *class* on page 2-36.

class scope The scope of C++ class members.

class template A blueprint describing how a set of related C++ classes can be constructed.

class tier The tier value of Workload Management classes specify which classes are most important. If no tier value is used, all classes are equally important. Resource limits (including the resource targets) of all classes in a higher tier are favored over those of classes in any lower tier.

classification mechanism
A set of Workload Management class assignment rules that determine the classes to which processes are assigned.

CLAW See *Common Link Access to Workstation* on page 2-43.

clean up The clean–up procedure instructs the system to attempt to remove software products that were partially installed. The system also attempts to revert to the previous version of the removed product. If the system successfully reverts to the previous version, it becomes the currently active version. If this cannot be done, then the software product is marked as broken. After the clean–up procedure is complete, you can attempt to install the software again.

clean–up code In loops that have been unrolled or blocked, an additional loop or set of loops that ensures that all iterations of the original code are executed in the unrolled or blocked code.

clear In X.25 communications, to reject a call (if it has not yet been accepted) or end a call.

clear cause See *cause code* on page 2-32.

clear collision A condition that occurs when a STE and a DCE simultaneously transmit a clear request packet and a clear indication packet over the same logical channel. See also *collision* on page 2-40 .

clear–confirmation packet
In X.25 communications, a packet transmitted by the DTE to inform the DCE that a call has been cleared. See also *packet* on page 2-166.

clear diagnostic
See *diagnostic code* on page 2-68.

clear–indication packet
In X.25 communications, a call supervision packet transmitted by a DCE to inform a DTE of the clearing of a call. See also *packet* on page 2-166.

clear–request packet
In X.25 communications, a call supervision packet transmitted by a DTE to ask for a call to be cleared. See also *packet* on page 2-166.

clear user data In X.25 communications, data optionally included in the clear–request packet by the user application.

click In CDE, to press and release a mouse button without moving the mouse pointer. Unless otherwise specified, mouse button 1 is assumed.

client (1) In a distributed file system environment, a system that is dependent on a server to provide it with programs or access to programs.
(2) In Enhanced X–Windows, an application program that connects to an Enhanced X-Windows server by an inter–process communication (IPC) path, such as a Transmission Control Protocol (TCP) connection or a shared memory buffer. The program can be referred to as the client of the server, but it is actually the IPC path itself. Programs with multiple paths open to the server are viewed as multiple clients by the protocol. See also *inter–process communication* on page 2-120.
(3) In Enhanced X–Windows, a Toolkit routine that uses a widget in an application or for composing another widget.
(4) In AIXwindows, a software application that fills the role of the client in the traditional client–server model upon which Enhanced X-Windows and AIXwindows are based. See also *client application* on page 2-38.
(5) In NCS, a program that uses an interface to make remote procedure calls (RPCs).

client agent See *Location Broker Client Agent* on page 2-133.

client application A type of application. See also *client* on page 2-38 and *application* on page 2-9.

client program A program that uses a C++ class. The program is said to be a client of the class.

client–side caching A high–speed buffer storage that contains frequently accessed information associated with a client application. The primary purpose of client–side caching is to reduce access time to key information.

clip In computer graphics, to remove those parts of a display image that lie outside of a given boundary.

clip list In Enhanced X–Windows, a list of rectangles designated for clipping.

clipboard A storage space set aside for the temporary storage and retrieval of text or graphics during cut–and–paste operations. Data in the clipboard is available to other applications.

clipboard selection Data selected and pasted to the clipboard that can be pasted or passed to a function. Data can include such elements as text, graphics, and widgets.

clipping In GL, if a primitive overlaps the boundaries of a window, it is *clipped*. The part of a primitive that appears in the window is displayed and the rest is ignored. There are several types of clipping that occur in the system. Three–D drawing primitives are clipped to the boundaries of a frustum (for perspective transformations) or to a rhombohedron (for orthographic projections). This 3–D clipping applies as well to the origin of character strings, but not to the characters themselves. A 2–D clipping is also performed, where all drawing is clipped to the boundaries of the AIXwindows window. The area of 2–D clipping can be controlled with the screenmask. See also *clipping planes* on page 2-39, *fine clipping* on page 2-91, *gross clipping* on page 2-103, *screenmask* on page 2-210, *bounding box* on page 2-17, *culling* on page 2-57, *transformation* on page 2-245, and *window* on page 2-263.

clipping planes
In GL, before clipping occurs, primitive space is mapped to normalized device coordinates. The clipping planes $x = +/- w$, $y = +/- w$, or $z = +/- w$ correspond to the left, right, top, bottom, near, and far planes bounding the viewing frustum. See also *gross clipping* on page 2-103, *clipping* on page 2-38 and *frustum* on page 2-97.

clipping region In Enhanced X–Windows, a type of graphics output. In a graphics context, the image defined by the bitmap or rectangles used to restrict output to a particular region of a window.

CLK See *clock* on page 2-39.

Clock In CDE, a Front Panel control that displays the local time.

clock (CLK) (1) A device that generates periodic signals used for synchronization.
(2) In data communication, equipment that provides a time base used in a transmission system to control the timing of certain functions, such as sampling, and to control the duration of signal elements.

clocking (1) In binary synchronous communication, the use of clock pulses to control synchronization of data and control characters.
(2) In data communications, a method of controlling the number of data bits sent on a communications line in a given time.

clone device A STREAMS device that returns an unused major or minor device when initially opened, rather than requiring the minor device to be specified by name in the open call.

close (1) To end an activity and remove that window from the display.
(2) A data manipulation function that ends the connection between a file and a program. Contrast with *open* on page 2-161.

closed user group (CUG)
In X.25 communications, a subgroup of users that is assigned to a facility that enables a member of one subgroup to communicate only with other members of the subgroup. A DTE can belong to more than one closed user group. See also *bilateral closed user group* on page 2-20 and *optional facility* on page 2-162.

cluster (1) Any configuration of interconnected workstations for the purpose of sharing resources (for example, local area networks, host attached workstations, and so on).
(2) A group of storage locations allocated at one time.
(3) A station that consists of a control unit (cluster controller) and the workstations attached to it.
(4) A page–size (4096–byte) buffer provided by the mbuf management facility to the various layers of communication software. (Also called ”cluster mbuf,” ”mbuf cluster,” and ”mapped page.”)

clustered file Files in which records with adjacent key values are physically clustered together. The clustered file organization optimizes sequential access through the primary index to records in the file.

CMOS Complementary metal–oxide semiconductor. CMOS is a technology that combines the electrical properties of n–type semiconductors and p–type semiconductors.

coaxial cable A cable consisting of one conductor, usually a small copper tube or wire, within and insulated from another conductor of larger diameter, usually copper tubing or copper braid.

COBOL Common business–oriented language. A high–level programming language, based on English, that is used primarily for business applications.

code (1) Instructions to the computer.
(2) To write instructions for the computer; to program.
(3) A representation of a condition, such as an error code.

code page (1) An assignment of graphic characters and control function meanings to all code points.
(2) Arrays of code points representing characters that establish the ordinal sequence (numeric order) of characters. This operating system uses 256–character code pages. See also *code point* on page 2-40 and *extended character* on page 2-86.
(3) An ordered set of up to 256 predefined display symbols. The first 32 code points of each code page are reserved for control codes and are the same for all code pages, leaving up to 224 distinct display symbols per page.

code point A character within a code page. See also *code page* on page 2-40 and *extended character* on page 2-86.

code segment See *segment* on page 2-213.

code server A system that is providing a code service for other computers on a network.

code set In the XPG4 system interface, a set of unambiguous rules that establish a character set and the one–to–one relationship between each character of the set and its bit representation.

col typesetter postprocessor that buffers typeset output to allow printing on printers and workstations that do not support backscrolling.

collapse To remove the contents of a directory from the display (close it) using the CDS Browser. To collapse an open directory, you double–click on its icon. Double–clicking on a closed directory expands it.

collating element
One or more characters that match a sequence in a *regular expression* on page 2-198.

collating sequence
The sequence in which characters are ordered within the computer for sorting, combining, or comparing.

collation The process of character and string sorting based on alphabetical order and equivalence class.

collation table Provides an ordered character set and character equivalence classes used by functions.

collection In Ada language, the entire set of objects created by evaluation of allocators for an access type.

colliding find request
A condition that occurs when two link stations attempt to call each other at the same time.

collision (1) An unwanted condition caused by concurrent transmissions on the medium that results in garbled data.
(2) In X.25 communications, a condition that occurs when a DTE and a DCE simultaneously transmit packets (for instance, a clear–request packet and a clear–indication packet) over the same logical channel. This can be a clear collision, call collision, or reset collision. See also *clear collision* on page 2-37.

colon format A format into which data files can be organized. Each data record consists of one line in the colon file, and data fields in each data record are separated by colons.

color cell In Enhanced X–Windows, an entry in a colormap that consists of three values based on red, green, and blue intensities. The values are 16–bit, unsigned numbers. Zero represents the minimum intensity. The values are scaled by the server to match the particular display in use.

color display A display device capable of displaying more than two colors and the shades produced by combinations of two colors, as opposed to a monochrome display.

color expansion operation
A graphics programming operation that occurs automatically when the source pixel map data area contains only 1 byte per pixel and the destination pixel map data area is a color display adapter buffer frame defined to have more than 1 bit per pixel.

color graphics adapter
An adapter that allows a computer to use a color display.

color lookup table
Synonym for *color map* on page 2-41.

color map (1) In computer graphics, a lookup table where each index is associated with a red, green, and blue value. Synonymous with *color lookup table*, *color palette*, and *color table*.
(2) In Enhanced X–Windows, a set of color cells. A pixel value indexes the color map to produce RGB–value intensities. A color map consists of a set of entries defining color values that, when associated with a window, is used to display the contents of the window. Depending on hardware limitations, one or more color maps can be installed at one time such that windows associated with those maps display correct colors. The two classes of color maps are direct color and pseudocolor.
(3) In GL, a lookup table that translates color indexes into RGB triplets. The lookup table is sandwiched between the frame buffer and the digital–to–analog converters (DACs) and serves to translate the color index value stored in the frame buffer into the red, green, and blue values required by the DACs. On most hardware configurations, the color map is either 8 or 12 bits deep, allowing the simultaneous display of 256 or 4096 colors. On most hardware configurations, the DACs have an 8–bit per color accuracy, allowing the user to choose among 16,777,216 colors.

color map mode
A configuration of the hardware that passes the values stored in the frame buffer through a color lookup table (color map), from which the red, green, and blue values are obtained for display. Entries in the color map are referred to as color indexes. In color map mode, the values stored in the frame buffer are treated as color map indexes. See also *RGB mode* on page 2-205.

color palette Synonym for *color map* on page 2-41.

color ramp A progression of colors in a color map. Most color ramps are smooth and have only a small number, if any, of discontinuities. For instance, if the full set of colors of the rainbow were loaded into the color map, that would constitute a color ramp.

color table Synonym for *color map* on page 2-41.

color viewer In AIXwindows, an interface that allows the user to choose a color as a value for a widget property.

column A vertical arrangement of characters or other expressions.

column headings
Text appearing near the top of a column of data for the purpose of identifying or titling the data in the column.

column inch A unit of measure for printed text. One column inch is the amount of text contained in an inch of type depth, one column wide.

column–major order
A way of storing array elements such that the leftmost subscript varies most rapidly as memory–adjacent elements are accessed.

combined I and D cache
A cache that contains both instructions and data, distinguishable only by the cache line tag.

comma expression
An expression that contains two operands separated by a comma. Although the compiler evaluates both operands, the value of the right operand is the value of the expression. If the left operand produces a value, the compiler discards this value.

command (1) A request to perform an operation or run a program. When parameters, values, flags, or other operands are associated with a command, the resulting character string is a single command.
(2) In data communication, an instruction represented in the control files of a frame and transmitted by a primary or combined station. It causes the addressed station to run a data link control function.

command frame
A link–level frame or packet that is serviced as a command and (in most cases) expects a response.

command history
An automatic listing of previously issued commands.

command interpreter
A program that sends instructions to the kernel. Synonym for *interface* on page 2-119. See also *shell* on page 2-218.

command line The area of the screen where commands are displayed as they are typed.

command line editing keys
Keys for editing the command line.

command mode
A state of a system or device in which the user can enter commands. See also *text input mode* on page 2-240.

command module
A file that executes a command or process. The file may be activated when the user enters a command at the command line or by another command module.

command name
(1) The first or principal term in a command. A command name does not include parameters, values, flags, or other operands.
(2) The full name of a command when an abbreviated form is recognized by the computer (for example, **print working directory** for **pwd**).

command programming language
Facility that allows programming by the combination of commands rather than by writing statements in a conventional programming language.

command string
A request to perform an operation, along with the operands that provide all instructions needed for running the operation.

command substitution
The ability to capture the output of any command as a value to another command by placing that command line within ‘ ‘ (grave accents). The shell first runs the command or commands enclosed within the grave accents and then replaces the whole expression, including grave accents, with their output. This feature is often used in assignment statements.

command word
The name of the 16–bit units used for storing graphic primitive strings. The first command word determines the primitive type and sets the length of the string. Subsequent command words contain information in multiples of quid, or 4 bits of data.

commit (1) To make permanent all changes that have been made to the database file since the last commitment operation and to unlock the records so they are available to other users. Contrast with *reject*.
(2) When you commit software, you are making a commitment to that version of the software product. When you commit a product, the saved files from all previous versions of the software product are removed from the system, thereby making it impossible to return to a previous version of the software product. In SMIT, software can be committed at the time of installation by setting the COMMIT software? question to **yes** (or by using the **–ac** flags with the **installp** command). Note that committing already applied software does not change the currently active version of the software product. It merely removes saved files for the previous version of the software product. The rejection of the installation level of the product does *not* have the same meaning as the rejection of updates to the product. Once you commit a new version of a product, you must reinstall the previous version if you want to use that version again. Compare to *apply* on page 2-10 and contrast with *reject* on page 2-198 and *remove* on page 2-201.

commit operation
An operation that saves a file to permanent storage.

common block In FORTRAN, a storage area that can be referred to by a calling program and one or more subprograms.

common carrier
Any government–regulated company that provides communication services to the general public.

Common Link Access to Workstation (CLAW)
The architecture that defines the channel commands used between the host and the channel attachment adapter.

common subexpression enhancement
An optimization that enables a compiler to detect that two or more distinct subexpressions within an expression or loop are identical and need only be computed once.

communications
The transmission of data according to a protocol between computers or remote devices, usually over a long distance.

communications adapter
A circuit card with associated software that enables a processor, controller, or other device to be connected to a network. See also *adapter* on page 2-5.

communications authority password
One of the two communications security passwords. It controls access to communication configuration menus so that only authorized persons can change the profiles, encrypt a portion of the communication profile database, or control the startup of SNA processes. The password must be a 30–to–80–character phrase, with interior blanks allowed. See also *BIND password* on page 2-17.

communications channel
An electrical path that facilitates transmission of information from one location to another.

communications co–processor
A microprocessor on an expansion board that supplements the operations of the processor in the system unit, enabling a computer to use communication services in parallel with other operations.

communications endpoint
In X.25, the local communication channel between a DLS user and a DLS provider.

communications line
The line over which data communications take place; for example, a telephone line. See also *X.25 line* on page 2-266.

communications line adapter (CLA)
A functional unit that converts the serial–by–bit output of a station to a parallel bit form and from a parallel bit form to serial–by–bit input to a station. See also *line adapter* on page 2-129.

communications link
See *data link* on page 2-61.

communications service
The service performed by the Sockets Application Programming Interface, which allows data packets to be delivered to the specified destination. There are three types of communications services offered: reliable stream delivery, connectionless datagram delivery, and raw socket delivery.

compatibility (1) The ability to perform tasks identically in different environments without major modifications.
(2) The capability of a functional unit to meet the requirements of a specified interface.

compatible Pertaining to computers on which the same program can be run without appreciable alteration.

compatible types
Different data types that can be operands for the same operation.

compilable unit
In Pascal, synonymous with *compilation unit* on page 2-45. See also *unit* on page 2-251.

compilation In Ada language, a compilation is generally the translation of an Ada source program into an executable object module. When using the Ada language debugger, a compilation consists of one or more compilation units in a single file. If you include three package specifications and two package bodies in one file, that file represents one compilation consisting of five compilation units: three library units and two secondary units. There is usually only one compilation unit in a compilation.

compilation time
The time during which a source program is translated from a high–level language into a machine language.

compilation unit
A portion of a computer program sufficiently complete to be compiled correctly. In Pascal, there are two types of units: the program unit and the segment unit. In Ada language, a compilation unit is the declaration or the body of a program unit, presented for compilation as an independent text. It is optionally preceded by a context clause, naming other compilation units upon which it depends by means of one more with clauses. See also *compilable unit* on page 2-44 and *external variable* on page 2-88.

compile To translate a program written in a high–level programming language into an intermediate language, assembly language, or a machine language. See also *interpreted routine* on page 2-120.

compiler (1) A program that translates a source program into an executable program (an object program).
(2) A program that translates instructions written in a high–level programming language into machine language.

compiler directing statement
Synonym for *compiler directive* on page 2-45.

compiler directive
A statement that controls what the compiler does rather than what the user program does.

complement of a number
The value that when added to the number equals a given value.

complete class name
The complete qualification of a nested C++ class name including all enclosing class names.

complete overwrite installation
An installation method that completely overwrites an existing version of BOS that is installed on your system. This procedure may impair recovery of data or destroy all existing data on your hard drives. Be sure to back up your system before doing a complete overwrite installation.

complete packet sequence
Either an individual X.25 data packet or a sequence of packets with the M–bit set to 1 and the D–bit set to 0, followed by a further data packet with the M–bit set to 0 and the D–bit set as required.

complex constant
In FORTRAN, an ordered pair of real or integer constants separated by a comma and enclosed in parentheses. The first constant of the ordered pair represents the real part of a complex number; the second represents the imaginary part.

Complex Mathematics Library
A C++ class library that provides the facilities to manipulate complex numbers and perform standard mathematical operations on them.

complex number
A number consisting of an ordered pair of real numbers, expressible in the form $a + bi$, where a and b are real numbers and i squared equals minus one. A complex number is made up of two parts: a real part and an imaginary part, where a is the value of the real part and b is the value of the imaginary part and where i is the square root of –1.

complex type In FORTRAN, a data type that represents values of complex numbers. A value is expressed as an ordered pair of real data items separated by a comma and enclosed in parentheses. The first item represents the real part of the complex number, and the second item represents the imaginary part.

component (1) One part of a structured type or value, such as an array element or a record field.
(2) In AIXwindows or Enhanced X–Windows, the widget, gadget, or other graphical object that makes up an interactive user interface.
(3) In Ada language, a value that is a part of a larger value, or an object that is part of a larger object.

component bar
In an Interleaf document, the black bar at the left margin that contains the component identifiers for the components contained in that document.

component dump table
A structure used by kernel components to identify data structures that should be collected by the kernel dump program.

compose To set type.

Compose window
In CDE, in Mailer, the window you use to create new electronic mail messages. Several message–composing options are available from the Compose window menu bar.

Composite Manager
In AIXwindows, a manager widget with special knowledge about the handling of one or more of its children widgets. Normally, a manager widget has no knowledge of its children, but a **TitleBar** widget and a **ScrollBar** widget can be registered as children of a certain type of Composite Manager widget known as a **Panel** widget, and the **Panel** widget will correctly control the positioning of the **TitleBar** and **ScrollBar** widgets.

composite type
In Ada language, a composite type is one whose values have components. There are two kinds of composite type: array types and record types.

composite video
The combined luna, chroma, and sync signals in accordance with the NTSC standards in the U.S., and the PAL standards in Europe. Also called NTSC or PAL (Europe).

composite widget
In Enhanced X–Windows, a widget that is a container for an arbitrary, implementation–defined collection of children. These children may be instantiated by the composite widget itself, by other clients, or by a combination. Composite widgets contain methods for managing the geometry (layout) of any child widget. A composite widget is a subclass of the Core widget. See also *widget* on page 2-262.

Composite widget class
In Enhanced X–Windows, a metaclass that does not instantiate any widgets of its own but provides the resources and functionality that allow parent widgets to manage the layout and mapping of their children widgets and gadgets.

compound license
In License Use Management, a type of license that allows a system administrator to generate license passwords for a given number of licenses. Such a license is valuable when an administrator needs a certain number of licenses, but does not yet know what machines or who will use them. A compound license can generate either nodelocked or non–nodelocked licenses, not both.

compound object
In AIXwindows, a graphical object made up of several widgets and gadgets collected within a single container widget.

compound string
A type of string designed to simplify foreign language support by allowing text to be displayed without hard–coding the language–dependent attributes (character set, text, and direction).

compress (1) To move files and libraries together on disk to create one continuous area of unused space.
(2) In data communications, to delete a series of duplicate characters in a character string.

compressed output
Synonym for *compression* on page 2-47.

compression (1) A technique for removing strings of duplicate characters, gaps, empty fields, and trailing blanks before transmitting data. Synonymous with *compressed output*.
(2) In SNA, the replacement of a string of up to 64 characters by an encoded control byte to reduce the length of the data stream sent to the LU–LU session partner.

computational memory
The set of all virtual–memory pages in real memory that are part of working–storage or program–text segments.

computed time The result of the synchronization process—the time value that the clerk or server process computes according to the values it receives from several servers.

computer aided instruction (CAI)
A data processing application in which a computing system is used to assist in the instruction of students.

Computer Graphics Metafile
A device–independent graphics file format used for storing object–oriented graphics.

computer instruction
An instruction that can be recognized by the processing unit of the computer for which it is designed. Synonymous with *machine instruction* on page 2-139.

computer language
Synonym for *machine language* on page 2-139 and *machine instruction* on page 2-139.

computer word
Synonym for *word* on page 2-263.

concatenate (1) To link together.
(2) To join two character strings.

concatenation (1) Linking together.
(2) Joining two character strings.
(3) In GL, combining a series of geometric transformations such as rotations, translations, and scaling. Concatenation of transformations corresponds to matrix multiplication.

concave and convex polygons
In computer graphics, a polygon is convex if a line segment joining any two points in the figure is completely contained within the figure. Nonconvex polygons are sometimes called concave. Algorithms that render only convex polygons are much simpler than those that can render both convex and concave polygons.

concentrator An FDDI node that has additional parts beyond those required for its own attachment to a FDDI network. These additional parts (type M) are for attaching other FDDI nodes (type S) in a tree topology. Primarily, a concentrator is used to allow more than two single attachment stations (SAS) to communicate. It can also connect multiple SAS to a dual attachment station (DAS) ring.

concrete class An OM class of which instances are permitted.

concurrent–use license
In License Use Management, a type of license administered by the license server that can be used by different users at any node that is connected to a license server node. Concurrent–use licenses allow as many users to use a software product concurrently as there are licenses.

condensed print
A print format where characters are smaller and spaced closer together horizontally, typically at a density of 17 characters per inch.

condition An expression in a program or procedure that can be evaluated to a value of either true or false when the program or procedure is running.

condition code Synonym for *flag* on page 2-92.

conditional Conditionals test for certain circumstances to carry out particular commands.

conditional branch
A branch that is taken when a specified condition is met.

conditional compilation statement
A preprocessor statement that causes the preprocessor to process specified code in the file depending on how a specified condition evaluates.

conditional expression
A C language expression that contains a condition (the first expression), an expression to be evaluated if the condition has a nonzero value (the second expression), and an expression to be evaluated if the condition has the value 0 (zero).

conditional statement
(1) A statement that runs if a specified expression evaluates to a nonzero value.
(2) A statement that permits execution of one of a number of possible operations, with or without a transfer of control.
(3) A statement used to express an assignment or branch based on specified criteria.

condition variable
A synchronization object used in conjunction with a mutex. A condition variable allows a thread to block until some event happens.

conditioning (1) The use of indicators to control when calculations or output operations are to be performed
(2) In data communications, the addition of equipment to a nonswitched voice–graded channel to provide minimum values of line characteristics required for data transmission.

conduit A pipe for protecting electrical wires or cables.

configuration (1) The group of machines, devices, and programs that make up a data processing system or network.
(2) The process of describing to a system the devices, optional features, and program products that have been installed so that these features can be used. Contrast with *customization* on page 2-58 and *system customization* on page 2-220.

Configuration Assistant
A graphical interface application used to perform post–installation system configuration tasks.

configuration feedback window
In AIXwindows, a window displayed in the center of the screen when AIXwindows Manager is restarted or when a behavior switch is requested. The window contains such variables as the size and location of a client window.

configuration file
A file that specifies the characteristics of a system or subsystem; for example, the operating system queueing system.

configuration manager
A program to supervise device configuration during initial program load (IPL).

configuration operation/procedure
The multistep process, performed in the host computer, of constructing a configuration image for a 3601 Finance Communication Controller.

Configuration Rules Object Class
An object class that contains the configuration rules used by the configuration manager during initial program load (IPL).

configure To describe to a system the devices, optional features, and program products installed on a system.

configure method
Takes a device from the defined state to the available state. If a device has a device driver, the configure method is responsible for loading and binding the driver into the kernel. If the device supports the optional stopped state, the configure method takes the device from the defined state to the stopped state.

confirm In X.25 communications, to respond to the arrival of a clear–indication or reset–indication packet.

CONFIRM A request that asks the remote transaction program to tell whether the last transmission was received successfully.

confirmation A transmission by a receiver that permits a sender to continue

CONFIRMED A response to the CONFIRM request indicating that the remote site received the transmission without detecting any errors.

conformant string
In Pascal, a string whose declared length does not match that of a formal parameter. See also *formal parameter* on page 2-94.

congruence class
In a cache, the group of lines to which a given memory location can be mapped.

connect In X.25 communications, to connect a port to the X.25 network.

connect data set to line (CDSTL)
In SNA, an option that determines how the data terminal ready (DTR) signal to the modem operates. It is used if DTR indicates an unconditional command from the DTE (data terminal equipment) to the attached DCE (data circuit–terminating equipment) to connect to or remove itself from the network.

connect–time accounting
The record of the amount of time each user spends logged in to the system.

connected unit In FORTRAN, a unit that is connected to a file by either an OPEN, READ, or WRITE statement.

connection (1) In SNA, the network path that links together two LUs in different nodes to provide communications channels between them for the application programs running at the respective LUs.
(2) In X.25 communications, the existence of a virtual circuit between two data terminal equipments (DTEs). A switched virtual circuit (SVC) connection is for the duration of a call; a permanent virtual circuit (PVC) connection is a permanent connection between the DTEs.
(3) In Enhanced X–Windows, the IPC path between the server and a client program. A client program typically, but not necessarily, has one connection to the server over which requests and events are sent.
(4) In system communications, a communication link over which data can be passed between two systems or between a system and a device.

connection close
All events made by the client are discarded and the server resets its state to having no connections.

connection establishment
The phase in connection mode that enables two data link service (DLS) users to create a data link connection between them.

connection identifier (CID
(1) A value used to identify a resource. The value is returned to the connecting program after connect processing has established a session and must be used on subsequent requests to the resource.
(2) In the X.25 API, the name used to identify a call that has been made or received.

connection key
Identifies a subclass of devices that can connect to the intermediate device at the specified location.

connection location
Identifies a specific location on the intermediate device where a child device can be connected.

connection management stream
In X.25, a special stream that receives all incoming connect indications destined for DLSAP addresses that are not bound to any other streams associated with a particular PPA.

connection mode
A circuit–oriented mode of transfer in which data is passed from one user to another over an established connection in a sequenced manner.

connection–oriented protocol
A connection–based, reliable, virtual–circuit transport protocol, such as TCP; an RPC protocol that runs over a connection–based transport protocol.

connection profile
A data management file that contains parameters that associate other defined profiles to the connection of two logical units.

connection type
This is a field in the Predefined Connection Object Class. For an intermediate device, it identifies the subclass of devices that can be connected to it.

connectionless mode
A mode of transfer in which data is passed from one user to another in self–contained units with no logical relationship required among the units.

connectionless packet delivery
A method of data packet delivery that treats each packet of information individually and does not guarantee delivery.

connectivity An algorithm that determines if two machines on different networks can communicate. If the machines can communicate, connectivity also determines which host names should be used and which TCP/IP routing information must be added.

connector (1) An electrical part used to join two other electrical parts.
(2) A flowchart symbol that represents a break in a flow line and indicates where the flow line is continued.
(3) A means of establishing electrical flow.

consistent Pertaining to a file system, without internal discrepancies.

console The main operating system display station. Synonym for *system console* on page 2-220.

console bell Synonym for *BEL* on page 2-20.

console device During the installation of the Base Operating System (BOS), the system console is the display device at the system on which you are installing the software.

console display
A display at a system console on which an operator can display, send, and reply to messages and use all control commands.

constant A data item with a value that does not change during the running of a program. Contrast with *variable* on page 2-255. For Ada programming, see also *object* on page 2-158.

constant expression
An expression having a value that can be determined during compilation and that does not change during the running of the program.

constant folding
Performing operations in which operands are all constants at compilation time and treating the results as constants. See also *fold* on page 2-93.

constant–width characters
A character set designed so each character is the same width as the other characters.

constraint In Ada language, a constraint determines a subset of the values of a type. A value in that subset satisfies the constraint.

Constraint In AIXwindows, a class of objects from which a unique resource set can be inherited. For example, a **PanedWindow** widget can specify the size of its children by using the inherited **XtNmin** and **XtNmax** Constraint resources. The reference material associated with each widget specifies those that inherit resources from the Constraint class.

constraint widget
In Enhanced X–Windows, a widget that is a subclass of a composite widget. It manages the geometry of its children based on constraints associated with each child.

constructor A special C++ member function that has the same name as a class. It is used to construct class objects and may initialize them.

Consultative Committee on International Telegraphy and Telephone (CCITT)
A United Nations Specialized Standards group whose membership includes common carriers concerned with devising and proposing recommendations for international telecommunications representing alphabets, graphics, control information, and other fundamental information interchange issues.

contact port Synonym for *well–known port* on page 2-261.

container In CDE, a control intended to hold objects or data. A folder and a notebook are examples of containers. In Mailer, your electronic mailbox and filing system that contains all mail messages. Once a message is put in a container, you can display, modify, delete, print, include, forward, and reply to it.

containment A pointer is ”contained” if the pointer is located in the window, and not within an inferior of the window, and the cursor hotspot is within a visible region of a viewable window or one of its inferiors. The border of the window is considered part of the window.

contention (1) In a local area network, a condition on a communications channel when two or more stations are allowed by the protocol to start transmitting concurrently and thus risk collision.
(2) A condition on a session when two programs try to start a conversation at the same time.

contention scope
The group of threads against which a given thread must compete for the CPU. If *local*, the thread competes against other threads in the same process. If *global*, the thread competes against all other threads in the system.

context address
A regular expression enclosed in slashes (/).

context clause For Ada programming, see *compilation unit* on page 2-45.

context line In the Performance Toolbox, menu items ending in a slash and three dots (/...). The slash and three dots signify that the line itself represents a list at the next hierarchical level. Contrast with *statistic line* on page 2-226. See also *cascading menu* on page 2-31.

context structure
An ordered group of variables specifying the interface properties (notably location) of a shadow widget.

continuation line
A line of a source statement into which characters are entered when the source statement cannot be contained on the previous lines.

continuation reference
A continuation reference describes how the performance of all or part of an operation can be continued at a different DSA or DSAs.

continue statement
A C language control statement that contains the keyword **continue** and a semicolon.

control In CDE, a generic term for a variety of elements (such as buttons, check boxes, and scroll bars) that perform an action or indicate an option setting. See *Front Panel control* on page 2-96.

control block A storage area used by a program to hold control information.

control character
(1) A character that is not a graphic character such as a letter, number, or punctuation mark. Such characters are called control characters because they frequently act to control a peripheral device. RETURN and FORM–FEED are control characters that control a workstation or printer. Synonymous with *nonprinting character*.
(2) The Ctrl key on the keyboard.
(3) A character, occurring in a particular context, that initiates, modifies, or stops any operation that affects the recording, processing, transmission, or interpretation of data (such as carriage return, font change, and end of transmission).
(4) A nonprinting character that performs formatting functions in a text file.

control commands
Commands that allow conditional or looping logic flow in shell procedures.

control state A state that represents the current Network Installation Management (NIM) operation being performed on a machine. This state is one of two machine states.

control key (1) The keyboard key labeled Ctrl.
(2) A key combination, made by pressing the Ctrl key followed by another key on the keyboard, that performs a function or makes a special character.

control path The set of line, hardware, and control disciplines that determine the current characteristics of a particular TTY.

control point profile name
The name of the control point profile that defines the node ID of the physical unit associated with the attachment.

control points In computer graphics, points in real space that control the shape of a spline curve. The system provides hardware support for wire frame rational cubic splines, and for NURBS surfaces, the specifications of which require four control points.

control program (CP)
Part of the operating system that determines the order in which basic functions should be performed.

control statement
In programming languages, a statement that is used to alter the continuous sequential execution of statements. A control statement can be a conditional statement or an imperative statement.

control station The primary or controlling computer on a multipoint line. The control station controls the sending and receiving of data.

control unit terminal (CUT) mode
A protocol used for communications with a 3174/3274 Controller or other appropriate interface unit. In this protocol, a program in the workstation is emulating a 3278/79 terminal for a user, and the interface unit is responsible for enforcing the protocol.

controlling terminal
This term refers to an active workstation at which a user is authorized to enter commands that affect system operation. The controlling terminal for any process normally is the active workstation from which the process group for that process was started. A workstation can have no more than one controlling process group and a process group can have no more than one controlling terminal. The controlling process group receives certain interrupt signals from the controlling terminal.

convenience creation subroutine
Creates certain useful combinations of widgets known as convenience widgets or convenience dialogs, including the appropriate **Shell** widgets where necessary. See also *convenience interface* on page 2-54.

convenience dialog
A widget or collection of widgets created by an AIXwindows Dialog convenience subroutine.

convenience function
A function that performs a frequently needed series of tasks automatically to create or manage widgets or other code structures. Convenience functions are included in the AIXwindows Toolkit.

convenience interface
An interface created by an AIXwindows convenience creation subroutine. See also *convenience creation subroutine* on page 2-54.

converged peripheral node
In SNA, a type of physical unit that has limited addressing and path control routing capabilities. It provides general connectivity to other SNA nodes and supports parallel sessions, multiple sessions per LU, primary and secondary LUs, and multiple lines per node.

conversation (1) In SNA, the logical connection between a pair of transaction programs for serially sharing a session between type 6.2 logical units from transaction to transaction. While a conversation is active, it has exclusive use of an LU–LU session as delimited by a distinct bracket; successive conversations may use the same session.
(2) An interchange of information between two application programs.
(3) A pathway between two application programs that allows them to exchange information.
(4) Interaction between a computer and a user by means of a keyboard.

conversation correlator
An internal SNA identifier used by the LU services to track which applications are using which conversations. An identifier of 1 to 8 bytes that is assigned by the attach function and maintained by LU services.

conversation key
A short–lived encryption key provided by the Authentication Service to two principals for the purpose of ensuring secure communications between them. See also *session key* on page 2-216.

conversation mode
A mode of operation of a computer system in which a sequence of alternating entries and responses between a user and the system takes place in a manner similar to a dialog between two persons.

conversion (1) In programming languages, the transformation between values that represent the same data item but belong to different data types.
(2) A change in the type of value. For example, when you add values having different data types, the compiler converts both values to the same form before adding them. See also *transaction program* on page 2-244.

conversion code
In a print function call, a specification of the type of the value, as the value is to be printed (in octal format, for example).

conversion function
A C++ member function that specifies a conversion from its class type to another type.

conversion specification
In a print function call, a specification of how the system is to place the value of zero or more format parameters in the output stream. Each conversion specification contains a % (percent) symbol that is followed by conversion modifiers and a conversion code.

converter A device that converts data from one form to another without altering the underlying information.

converter cache
A high–speed buffer storage that contains frequently accessed information associated with a client application. The primary purpose of a converter cache is to reduce access time to key information.

coordinate system
A given convention for locating pixels on a given display or window, where, in AIXwindows, *X* is the horizontal axis and *Y* is the vertical axis. The origin is [0,0] at the upper–left or lower–left corner, depending on the convention in use. For a window, the origin is at the upper left or lower left (depending on the convention in use), inside the border. Coordinates are discrete and specified in pixels. Each window and pixmap has its own coordinate system.

coprocessor (1) A supplementary processor that performs operations in conjunction with another processor.
(2) In personal computers, a microprocessor on an expansion board that extends the address range of the processor in the system unit or adds specialized instructions to handle a particular category of operations.

copy (1) The action by which the user makes a whole or partial duplicate of an already existing data object.
(2) Either a copy of an entry stored in other DSAs through bilateral agreement, or a locally and dynamically stored copy of an entry resulting from a request (a cache copy).

copy constructor
A C++ constructor used to make a copy of a class object from another class object of the same class type.

copy–link A link established between a target file and the copied version of the same file. Any changes made to the copied version of the target file are automatically made to the original target file.

copy–on–write An option that creates a mapped file with changes that are saved in the system paging space, instead of saving the changes to the copy of the file on the disk.

Core In AIXwindows, Core is the top–level superclass from which all widgets and gadgets are derived. Core consists of three subclasses (**Object**, **RectObject**, and **WindowObj**) that collectively provide the appearance resources and behavioral resources required by all widgets and gadgets in the AIXwindows toolkit.

core sequence controller
One of three control programs for the initial program load (IPL) ROM. The core sequence controller accepts control from the initial sequence controller and passes control to the IPL controller.

core widget In Enhanced X–Windows, the widget that contains the definitions of fields common to all widgets. All widgets are subclasses of the core widget. See also *widget* on page 2-262.

corequisite A product or update that must be installed concurrently with another specified product or update.

correlator A value passed between two or more programs that allows correlation or identification of mutual resources.

counter (1) A register or storage location used to accumulate the number of occurrences of an event.
(2) In the X.25 API, a variable that is increased by one when a packet arrives and is decreased by one when a packet is received; it can be used to notify the application program of incoming packets.

counter identifier
In the X.25 API, the name of a counter.

country code In X.25 communications, the 3–digit number that precedes the national terminal number in the network user address (for public networks).

coupler A device connecting a modem to a telephone network.

courier In DTS, a local server that requests a time value from a randomly selected global server each time it synchronizes.

CP See *control program* on page 2-53.

CPS Characters per second.

CPU See *central processing unit* on page 2-33.

CPU lock See *nodelocked license* on page 2-154.

CPU time The amount of time a program is running in the CPU or is being serviced by the operating system. Does not include time associated with the program's I/O or time in which other processes preempt the program's use of the CPU.

crash An unexpected interruption of computer service, usually due to a serious hardware or software malfunction.

CRC See *cyclical redundancy check character* on page 2-30.

Create Action In CDE, a software application that enables you to associate an icon with a command so that the command can be issued by clicking on the icon. Create Action is also used to define specific data types for an application's data files and to associate icons with those data types.

creation date The date when the file was created. See also *session date* on page 2-216 and *system date* on page 2-220.

critical resource
The system resource whose speed and/or size limits the speed with which a particular workload can be processed.

critical sections
Portions of shared data to which simultaneous access by multiple threads or applications must be prevented.

cross–referencer
In Ada language, a tool that provides a listing of all places where symbols are declared, assigned to, or referenced within a compilation unit. Symbols are identified by name, class, and enclosing unit, and references to the symbol are identified by source file line numbers.

CRQ See *call request* on page 2-29.

CRT See *cathode ray tube* on page 2-32.

CSC See *core sequence controller* on page 2-39.

CSMA/CD See *carrier sense multiple access with collision detection* on page 2-31.

CSX Host–based diagnostics program.

C–stub The part of the DUA that implements the connection with the communications network.

CTC Channel–to–channel.

CTS Clear to send. Used with EIA–232 protocol.

CUD See *call user data* on page 2-29.

CUG See *closed user group* on page 2-39.

culling (1) In GL, if a primitive is smaller than the minimum size specified in the command, it is *culled*: no further commands in the primitive are interpreted. See also *clipping* on page 2-38 and *pruning* on page 2-174.
(2) In the graPHIGS API, if a polygon is backfacing, it may be culled, or not rendered.

currency time The time at which a user reads news items. The news command considers only the items posted after this time to be current for the user.

current For an Ada–language compilation unit, the state where none of that unit's supporters has been recompiled since the unit itself was compiled. This implies that all of the supporters exist, since the Ada language requires this in order for the program to be compiled. Currency implies that all supporters of a unit were compiled in the correct order, as defined by Ada language rules.

current character position
The two–dimensional screen coordinates where the next character string or pixel read/write operation will occur.

current color The color that is employed to color all subsequent drawing primitives. All drawing primitives are drawn with this color until it is changed.

current directory
The directory that is active and can be displayed with the **pwd** command. Relative path name resolution begins in the current directory. Synonymous with *current working directory* and *working directory*.

current file (1) The file being edited. If multiple windows are in use, the current file is the file containing the cursor.
(2) In the **make** command, the file that the **make** command is working with at a given moment. The **make** command replaces the **$*** macro with the name of the current file.

current folder In CDE, the currently opened folder in an active File Manager view.

current graphics position
The homogeneous three–dimensional point from which geometric drawing commands draw. The current graphics position is not necessarily visible.

current graphics window
In GL, the window to which the system directs the output from graphics routines.

current heap In Pascal, the area of storage where dynamic variables allocated by calls to NEW reside. Other heaps can exist at the same time, but only one is current.

current host Synonym for *local host* on page 2-132.

current line The line on which the cursor is located.

current record (1) The record pointed to by the current line pointer.
(2) The record that is currently available to the program.

current record pointer (CRP)
A logical indicator used when sequentially processing SFS files or selected ranges of records from those files. The CRP tracks which of the records in the selected range has just been processed and which will be processed next.

current selection
A highlighted text block or element.

current session
In CDE, the session saved by Session Manager when you log off. At the next login, unless you specify otherwise, this session automatically opens, enabling work to continue where you left off. Contrast with *home session* on page 2-108.

current transformation matrix
The transformation matrix on top of the matrix stack. All points passed through the graphics pipeline are multiplied by the current transformation matrix before being passed on. The current transformation matrix is a concatenation of the current modeling and viewing matrices. See also *transformation* on page 2-245 and *matrix stack* on page 2-142.

current window
The window to which the system directs the output from graphics routines. See also *window* on page 2-263.

current working directory
Synonym for *current directory* on page 2-57.

cursor (1) A movable symbol (such as an underline) on a display that indicates to the user where the next typed character will be placed or where the next action will be directed.
(2) In Enhanced X–Windows, the visible shape of the pointer on a screen. A cursor consists of a hotspot, a source bitmap, and a pair of colors.
(3) A primitive such as an arrowhead that can be moved about the screen by means of an input device (typically a mouse).

cursor glyph In GL, a 16x16 or 32x32 raster pattern (bitmap that determines the shape of the cursor. A GL cursor glyph can be one or two bits deep; thus, a GL cursor can use up to three colors. Color 0 is always transparent.

cursor ID In Enhanced X–Windows, a unique identification number that is associated with each unique type of cursor.

cursor movement keys
The directional keys used to move the cursor without altering text.

cursor stability In file systems other than one managed by SFS, the ability to simultaneously maintain multiple contexts within a single file system. The SFS equivalent of a cursor is provided by the ability to simultaneously obtain multiple OFDs on a single file.

Customer Information Control System (CICS)
A licensed program that enables transactions entered at remote workstations to be processed concurrently by user–written application programs. It includes facilities for building, using, and maintaining databases.

customization (1) In the NIM environment, this is optional software installation.
(2) The process of describing optional changes to defaults of a software program that is already installed on the system and configured so that it can be used. Contrast with *configuration* on page 2-48. See also *system customization* on page 2-220.

customization profile
A file containing the descriptions of optional changes to the default settings of a device or a software program. See also *profile* on page 2-186.

customize (1) To describe to the system the devices, programs, users, and user defaults for a particular data processing system or network.
(2) To describe optional preferences or changes to defaults in a software program that is already installed and configured. Contrast with *configure* on page 2-49.

Customized Database
An entity within the ODM that contains configuration data for defined or available devices in the system. See also *Device Configuration Database* on page 2-67 and *Predefined Database* on page 2-181.

Customized Devices Object Class
A representation within the ODM of each device instance as distinguished by a unique logical name. The Customized Devices Object Class contains basic information about the device such as device status and how to access the information contained in other object classes.

CUT See *control unit terminal mode* on page 2-53.

cycle time (1) The time elapsed during one cycle of the processor. Cycle time varies from one type of processor to another.
(2) The minimum time interval between starts of successive read/write cycles of a storage device.

cyclic redundancy check (CRC) character
A character code used in a modified cyclic code for error sensing and correction.

cylinder All fixed disk or diskette tracks that can be read or written without moving the disk drive or diskette drive read/write mechanism.

cylindrical coordinate system
An array where the left edge of each row is functionally adjacent to the right edge of the same row.

Cyrillic Cyrillic alphabet. An alphabet used for writing Old Church Slavonic and for Russian and various other Slavic languages.

D

DAC See *digital–to–analog converter* on page 2-69.

daemon A program that runs unattended to perform a standard service. Some daemons are triggered automatically to perform their task; others operate periodically. An example is the **cron** daemon, which periodically performs the tasks listed in the **/var/spool/cron/crontabs** directory. Synonym for *demon*.

daemon process
A process begun by the root user or the root shell that can be stopped only by the root user. Daemon processes generally provide services that must be available at all times, such as sending data to a printer.

DAF Destination Address Field.

dangling else A condition arising as a result of nesting an IF statement in the IF part of an IF–ELSE statement. The ELSE statement is associated with the closest IF statement, in this case, the inner one. Placing an empty ELSE statement in the nested statement prevents misinterpretation by forcing the outer ELSE statement to associate with the outer IF statement.

DAS Dual–attachment station. A station that connects to both the primary and secondary FDDI rings

DASD Direct access storage device. A device in which access time is effectively independent of the location of the data. Information is entered and retrieved without reference to previously accessed data. DASDs include both fixed and removable storage devices.

data A representation of facts or instructions in a form suitable for communication, interpretation, or processing by human or automatic means. Data includes constants, variables, arrays, and character strings.

data area An area of memory that contains specific control variables that are normally predefined in structures or vectors.

data block See *block* on page 2-19.

data cache A cache for providing data to the processor faster than it can be obtained from RAM.

data cache unit (DCU)
See *cache* on page 2-28.

data circuit A pair of associated transmit and receive lines that provide a means of two–way data communications.

data–circuit–terminating equipment (DCE)
In a data station, the equipment installed at the user's premises that provides all the functions required to establish, maintain, and end a connection, and the signal conversion and coding between the data terminal equipment (DTE) and the line.

data communications
See *communications* on page 2-43.

data consumer In Performance Toolbox, a description of a program that receives statistics over the network from the **xmservd** daemon and prints, post–processes, or otherwise manipulates the raw statistics. Synonymous with client on page 2-38. *Contrast with data supplier* on page 2-62.

data definition A program statement that describes the features of, specifies relationships of, or establishes the context of, data. A data definition can also provide an initial value. Definitions appear outside a function or at the beginning of a block statement.

data dependency
A situation in which a source operand for a computation is the result of a preceding computation.

data description
For data objects that are not self–describing, components of the data object that describe the data so that it may be processed.

Data Encryption Standard (DES)
A data encryption algorithm widely used in the United States.

data item A unit of data to be processed that includes constants, variables, array elements, and character substrings.

data link (1) The assembly of parts for two data workstations that are controlled by a link protocol and the interconnecting data circuit, which enables data to be transferred from a data source to a data sink.
(2) The interconnecting data circuit and the link protocol between two or more workstations, not including the data source or data sink.
(3) The physical connection and the connection protocols between units that exchange data over a telecommunications line. See also *X.25 link* on page 2-266.

data link control layer
In SNA and X.25, the layer that consists of the link stations that schedule data transfer over a link between two nodes and perform error control for the link.

data link control (DLC) protocol
In SNA, the set of rules used by two nodes on a data link to accomplish an orderly exchange of information.

data link escape (DLE) character
In BSC, a transmission control character usually used in transparent text mode to indicate that the next character is a transmission control character.

data–link level In the hierarchical structure of a data station, the conceptual level of control or processing logic between high level logic and the data link that maintains control of the data link. The data link level performs such functions as inserting transmit bits and deleting receive bits; interpreting address and control fields; generating, transmitting, and interpreting commands and responses; and computing and interpreting frame check sequences. Synonym for *frame level* on page 2-96. See also *packet level* on page 2-167 *, physical level* on page 2-175.

data lock (1) The insurance of data availability to a single application program as a protection against conflicting updates to a data record.
(2) The system lock that locks data segment into memory.

data object A collection of data referred to by a single name. See also *object* on page 2-158.

data packet In X.25 communications, a packet used for the transmission of user data on a virtual circuit at the DTE/DCE interface. See also *packet* on page 2-166.

data storage interrupt
An interrupt posted when a fault is encountered accessing storage or I/O space. A typical data storage interrupt is a page fault or protection violation.

data stream (1) All information (data and control information) transmitted over a data channel in a single read or write operation.
(2) A continuous stream of data elements being transmitted, or intended for transmission, in character or binary–digit form using a defined format.
(3) All information sent to the terminal device driver with a **write** subroutine. Synonymous with *stream* on page 2-227.

data supplier In Performance Toolbox, a program that supplies statistics across a network. Synonymous with *server*. Contrast with *data consumer* on page 2-60.

data terminal equipment (DTE)
(1) The part of data processing unit that serves as a data source, data sink, or both.
(2) The user of the network.

data terminal ready (DTR)
A signal to the modem used with EIA–232 protocol.

data transfer The movement, or copying, of data from one location and the storage of the data at another location.

data type (1) In programming languages, a set of values together with a set of permitted operations.
(2) The mathematical properties and internal representation of data and functions.
(3) An attribute used for defining data as numeric or character.
(4) The type, format, or classification of a data object.
(5) In CDE, a mechanism that associates particular data files with the appropriate applications and actions. Data types can determine the type of a file based on file–naming conventions, such as a particular extension name, or on the contents of the file. See also *character type* on page 2-35 and *type* on page 2-248.

database A collection of facts and instructions comprising at least one file that is sufficient for a given purpose.

datagram (1) In packet switching, a self–contained packet, independent of other packets, that carries information sufficient for routing from the originating data terminal equipment (DTE) to the destination DTE without relying on earlier exchanges between the DTEs and the network.
(2) In DCE, an unreliable network data packet that is independent of all other packets and lacks any guarantees of delivery or sequentiality.

datagram protocol
A connectionless, datagram–based transport protocol, such as UDP; an RPC protocol that runs over a connectionless transport protocol.

dataless A workstation without local file systems or local boot images that accesses some of its resources remotely. Dataless clients use a local disk used for paging and dump devices.

dB See *decibel* on page 2-63.

DB Database.

DBA See *direct–bus attached* on page 2-70.

DBCS See *Double–Byte Character Set* on page 2-74.

D–bit In X.25 communications, the bit in a data packet or call–request packet that is set to 1 if end–to–end acknowledgment (delivery confirmation) is required from the recipient.

DC See *DCU* on page 2-63.

DCD Data carrier detect used with EIA–232 protocol. See also *CD* on page 2-32 and *decode* on page 2-64.

DCE See *Distributed Computing Environment* on page 2-73 and *data–circuit–terminating equipment* on page 2-60.

DCN Distributed Computer Network.

DCU Data cache unit. See also *cache* on page 2-28.

DD See *device driver* on page 2-68.

DDN Department of Defense Network.

dead code elimination
A compiler optimization that removes code that is never referenced, or that is always branched over. A compiler optimization that removes store instructions for data entities whose final values are not used.

dead letter file A file containing mail messages that could not be sent to a proper destination file.

dead variable In Ada language, a variable that is initialized, but is not used within the context of the program. Like unreachable code, dead variables are detected and removed by the optimizer.

dead zone An area of a tablet from which no input reports are generated. Each virtual terminal can set its own dead zones. Synonymous with *no–input zone*.

deadlock (1) An error condition in which processing cannot continue because each of two elements of the process is waiting for an action by or a response from the other.
(2) Unresolved contention for the use of a resource. (3.) An impasse that occurs when multiple processes are waiting for the availability of a resource that does not become available because it is being held by another process that is in a similar wait state.

deallocate To release a resource assigned to a specific task. Contrast with *allocate* on page 2-8.

DEALLOCATE A request to remove the allocation of the specified conversation from the local transaction program.

deallocation An operation that removes a client's permission to use a resource.

debug To detect, locate, and correct errors in the configuration of a computer system or a software program.

debugger A program or programs used to detect, trace, and eliminate errors in computer programs or software.

debugging Acting to detect and correct errors in software or system configuration.

debugging mode
A special mode in which a program provides detailed output about its activities to aid a user in detecting and correcting errors in the program itself or in the configuration of the program or system.

decibel (dB) (1) One tenth of a bel.
(2) A unit of signal strength, such as the signal on a data communications channel.
(3) A unit for measuring relative power. The number of decibels is 10 times the logarithm (base 10) of the ratio of the measured power levels.

decimal (1) Pertaining to a system of numbers to the base 10. The decimal digits range from 0 through 9.
(2) Characterized by a selection, choice, or condition that has 10 possible different values or states.

decimal constant
A number containing any digits 0 through 9.

declaration (1) A description that makes a defined object available to a function or a block.
(2) In programming languages, the mechanism for establishing a language object. A declaration normally involves attaching an identifier and allocating attributes to the language object concerned.
(3) In a programming language, a meaningful expression that affects the interpretation of other expressions in that language.
(4) In Ada language, a declaration associates an identifier (or some other notation) with an entity. This association is in effect within a region of text called the *scope* of the declaration. Within the scope of a declaration, there are places where it is possible to use the identifier to refer to the associated declared entity. At such places the identifier is said to be a *simple name* of the entity; the *name* is said to *denote* the associated entity. See also *declare* on page 2-64, *simple name* on page 2-219, *scope* on page 2-210, and *name* on page 2-150.

declarative part
In Ada language, a sequence of *declarations.* It may also contain related information such as *subprogram bodies* and *representation clauses*.

declarator An identifier and optional symbols that describe the data type.

declare A selection available from the Interpreter's Interpret menu that allows a user to assign variable names and structure definitions. See also *declaration* on page 2-64.

decode (1) To convert data by reversing the effect of some previous encoding.
(2) To interpret a code. See also *DCD* on page 2-63.

default A value, attribute, or option that is assumed when no alternative is specified by the user. See also d *efault value* on page 2-65.

default accelerators
See *accelerator* on page 2-3.

default arguments
Arguments that are declared with default values in a C++ function prototype or declaration. If a call to the function omits these arguments, default values are used. Arguments with default values must be the trailing arguments in a function prototype argument list.

default button labels
In AIXwindows, **XmLabel** widgets or gadgets that are used when no other button label has been specified.

default cell (global access)
An NCS cell that allows access from any node in the network. This is the most common cell used in iFOR/LS configurations because it allows all other iFOR/LS servers in the default cell to communicate freely.

default clause In a C switch statement, the keyword default followed by a colon and one or more C statements. When the conditions of the specified case labels in the switch statement do not hold, the default clause is chosen.

default constructor
A C++ constructor that takes no arguments, or if it takes any arguments, all its arguments have default values.

default device The device attached to your computer (such as a printer or disk drive) that is used when no alternative is specified by the operator.

default directory
The directory name supplied by the operating system if none is specified.

default drive The drive name supplied by the operating system if none is specified.

default files Data files in which resource default values are stored in ASCII form to permit the assignment of alternative resource values at run time without need for rewriting or recompiling source code.

default initialization
The initial value assigned to a data object by the compiler if no initial value is specified by the programmer. In C language, external and static variables receive a default initialization of zero, while the default initialization for auto and register variables is undefined. See also *default value* on page 2-65.

default label See *label* on page 2-126.

default printer A printer that accepts all the printed output from a display station assigned to it.

default shell In AIXwindows, the shell that is used when no other shell properties have been specified.

default value A value stored in the system that is used when no other value is specified. See also *default* on page 2-64 and *default initialization* on page 2-65.

defaults file See *default files* on page 2-65.

define Creates an entry in the ODM Customized Devices Database and establishes the parent device and connection location.

define method Used to create a device instance in the ODM Customized Database. It takes a device from the undefined or nonexistent state to the defined state.

define statement
A preprocessor statement that causes the preprocessor to replace an identifier or macro call with specified code.

defined state The state a device is put into when its defined method is run or when an available device's unconfigure method is run. The device is not a usable device at this point.

DEL See *delete character* on page 2-65.

delayed port A port that is enabled like a shared port except that the login herald is not displayed until you type one or more characters (usually carriage returns). A port directly connected to a remote system or intelligent modem is usually enabled as a delayed port.

delete (1) To remove. For example, to delete a file. See also *erase* on page 2-82.
(2) The C++ keyword **delete** identifies a free store deallocation operator. In C++, the delete operator is used to destroy objects created by **new**.

delete character (DEL)
(1) A control character used primarily to obliterate an erroneous or unwanted character.
(2) A character that identifies a record to be removed from a file.

delimiter (1) A character or sequence of characters that marks the beginning or end of character string or unit of data.
(2) A character that groups or separates words or values in a line of input.

deliver (callback)
Delivering a callback or upcall means to cause its invocation.

delivery–confirmation bit
See *D–bit* on page 2-62.

delta The finite increment of a variable.

demangling The conversion of mangled C++ names back to their original source code names. During compilation, identifiers such as function and static class member names are mangled (encoded) with type and scoping information to ensure type–safe linkage. These mangled names appear in the object file and the final executable file. Demangling converts these names back to their original names to make program debugging easier.

demon Synonym for *daemon* on page 2-60.

denote See *declaration* on page 2-64.

density In printing, refers to the number of characters per inch horizontally.

dependency line
The first line of an entry in a description file. It contains a list of target files followed by a colon and an optional list of prerequisite files or dependencies.

dependent A software product that requires another product or update to be installed *before* or *at the same time* it is installed. Contrast with *prerequisite* on page 2-182

dependent workstation
A workstation having little or no stand–alone capability that must be connected to a host or server in order to provide any meaningful capability to the user.

dependents Ada–language compilation units that would have to be recompiled if another unit were to be recompiled because of the compilation order imposed by the Ada language.

depth (1) In Enhanced X–Windows, the number of bits per pixel for a window or pixmap.
(2) In a three–dimensional context, the second dimension.

depth–cueing In 3D computer graphics, varying the intensity of a line with depth. Typically, the points on the line further from the eye are darker, so the line seems to fade into the distance.

dequeue To remove items from a queue. Contrast with *enqueue* on page 2-80.

dereferenced pointer
In Pascal, an expression using the –> or @ operator used to locate a dynamic variable from a pointer.

derivation The process of deriving a C++ class from an existing class, called a base class.

derived class A C++ class that inherits the properties of a base class. You can add additional data members and member functions to the derived class. A derived class object can be manipulated as if it were a base class object. The derived class can override virtual functions of the base class.

derived type In Ada language, a type whose operations and values are replicas of those of an existing type. The existing type is called the parent type of the derived type. See also *parent type* on page 2-170.

DES See *Data Encryption Standard* on page 2-61

descendant See *child* on page 2-35.

descending key sequence
The arrangement of data in order from the highest value of the key field to the lowest value of the key field.

descriptor (1) In ODM, a named and typed variable that defines a single characteristic of an object. See also *terminal descriptor* on page 2-239, *link descriptor* on page 2-130, and *method descriptor* on page 2-144.
(2) In information retrieval, a parameter word used to categorize or index information.
(3) In XOM, the means by which the client and service exchange an attribute value and the integers that denote its representation, type, and syntax.
(4) In XDS, a defined data structure that is used to represent an OM attribute type and a single value.

deselect To cancel the selection of a button. With a mouse, you deselect a highlighted area with the Select (left) button. Otherwise, you can use the Select key on the keyboard. To deselect a default button, select an alternate button in the selection list.

deserialize (1) To change from serial–by–bit to parallel–by–byte.
(2) In XDR, to change from XDR format to a particular machine representation.

desktop A visual representation of a group of objects in your system, brought together to help you organize your work.

destination cursor
A point or location marked by the cursor to which data is to be pasted or inserted.

destination disk
The disk to which you are installing.

destructor A special member function of a class with the same name as the class with a ~ (tilde) preceding the name. You cannot specify arguments or a return type for this function. A destructor ”cleans up” after an object by doing such things as freeing any storage that was dynamically allocated when the object was created.

device (1) A mechanical, electrical, or electronic machine that is designed for a specific purpose and that attaches to your computer, such as a printer, plotter, or disk drive.
(2) A valuator, button, or the keyboard. Buttons have values of 0 or 1 (up or down); valuators (mouse, dials) return values in a range, and the keyboard returns ASCII values.

device class Functional grouping of devices. The generic name for a group of device types, for example, all display stations belong to the same device class. Contrast with *device type* on page 2-68.

Device Configuration Database
Stores all information relevant to support the device configuration process. It consists of a Predefined Database and a Customized Database. See also *Predefined Database* on page 2-181 and *Customized Database* on page 2-29.

device definition
Information about a device that is in the Customized Database including attributes and connection locations.

device description
Text used to give a short description of the device. For example, the device description for the token–ring adapter might be ”Token–Ring High–Performance Adapter.”

device driver (DD)
(1) A program that operates a specific device, such as a printer, disk drive, or display.
(2) A collection of subroutines that control the interface between I/O device adapters and the processor.

device handler The component of a device driver that communicates directly with the hardware. Synonymous with *virtual device driver*.

device head The component of a device driver that implements the application program interface to a device.

device instance
When a device is defined, a Customized Devices Object Class entry is created. This entry is considered a device instance. There is a device instance for each device defined in the system.

device location
Indicates the location path of a device. This is a field in the Customized Devices Object Class.

device manager
For complex interfaces, a collection of routines that acts as an intermediary between drivers and virtual machines. For example, supervisor calls from a virtual machine are examined by a device manager and routed to the appropriate subordinate device drivers.

device name (1) The logical or symbolic name reserved by the system to refer to a specific device.
(2) SNA uses the operating–system device name of the network adapter to get information that defines the interface. See also *logical name* on page 2-135.

device number The reference number assigned to any external device.

device stanza Defines a device attached to a queue in the print spooling system. A device stanza contains all information pertaining to the device (usually a printer) and is found in the **/etc/qconfig** file.

device state Indicates the current configuration status of a device instance. Possible values are defined, available, and stopped. This is a field in the Customized Devices Object Class.

device subclass
Distinguishes devices within the same functional class. It is used to indicate different interfaces. For example, the printer class has three subclasses: rs232, rs422, and parallel.

device switch table
(1) A table that is used as an interface to the device drivers.
(2) A table that contains a pointer to the entry points for each device head.

device type The general name for a kind of device sharing the same model number; for example, 2311, 2400, 2400–1. Contrast with *device class* on page 2-67.

DFT See *distributed function terminal* on page 2-73.

DHCP Dynamic Host Configuration Protocol. An application–layer protocol that allows a machine on the network, the client, to get an IP address and other configuration parameters from the server.

diacritic A diacritic is a mark added to a letter to indicate a special phonetic value. Diacritics are implemented by a nonspacing character sequence, a two–key sequence consisting of one of 13 diacritics followed by an alphabetic character or a space. The terminal device driver converts this two–key sequence into a single code point.

diagnostic Pertaining to the detection and isolation of errors in programs and faults in equipment.

diagnostic aid A tool (procedure, program, or reference manual) used to detect and isolate a device or program malfunction or error.

diagnostic code
In X.25 communications, a 1–byte code included in clear– and reset–indication packets that gives information about the reason for sending the packet. See also *cause code* on page 2-32.

Diagnostic Control Program
The top–level control program and configuration manager for diagnostics. It traverses the configuration database, testing resources and their interdependencies. It analyzes conclusions from diagnostic applications and generates a problem report.

diagnostic output
Error or status messages produced by processes, in addition to those messages produced by standard output. Synonymous with *error output*.

dial (1) A computer input device that allows a user to set parameter values. Dials are a type of valuator. See also *valuator* on page 2-255.
(2) An I/O device used to input variables by way of thumbwheels.

dial and switch box
An I/O device with 8 dials (valuators) and 32 switches. The switch box is also called a ”button box” or the ”lighted programmable function keys (LPFKs).”

dial–code In BNU, a code representing a telephone number or portion of a telephone number.

dialing directory
In ATE, a list of telephone numbers that can be called with Asynchronous Terminal Emulation (ATE). It is similar to a page in a telephone directory.

dialog In an interactive system, a series of related inquiries and responses similar to a conversation between two people.

Dialog In AIXwindows, a two–way text interface between an application and its user. The interface takes the form of a collection of widgets and gadgets, including a **DialogShell** widget, a **BulletinBoard** widget (or a subclass of a **BulletinBoard** widget or some other container widget), plus various children, including **Label**, **PushButton**, and **Text** widgets.

dialog box A window that is displayed when further information is needed from the user, or when the system needs to display information. A line enclosure on a structure, such as a widget or shell, that contains a specific, two–way text interface.

dialog shell A subclass of transient shell, this shell is used for dialog boxes. See also *dialog widget* on page 2-69.

dialog widget Any of a class of widgets chosen through the Create menu’s Dialog submenu; dialog widgets always have the dialog shell as their implicit shell. See also *dialog shell* on page 2-69.

digest Data that has been organized into a format that provides for quick access to each piece of data.

digit (1) A character that represents a nonnegative integer. Synonymous with *numeric character*.
(2) A symbol that represents one of the nonnegative integers smaller than the radix.
(3) Any of the numerals from 0 through 9.

digital–to–analog converter (DAC)
(1) A highly specialized chip that converts the digital values coming out of the frame buffer into the rapidly varying analog voltage levels that are required by the monitor.
(2) That portion of the display subsystem that converts pixels into colors or grayscale.

dimension The attribute of size given to arrays and tables.

direct addressing
(1) An addressing method that uses an expression as an operand entry to represent an instruction address.
(2) A method of addressing in which the address part of an instruction contains a direct address.

direct–bus attached
Used in relation to the fixed disk drives attached directly to the system board rather then through a SCSI adapter card.

direct color (1) In Enhanced X–Windows, a class of color map in which a pixel value is decomposed into three separate subfields for indexing. One subfield indexes an array to produce red intensity values, the second indexes another array for blue intensity values, and the third for green intensity values. The RGB values can be changed dynamically. This is mutually exclusive to the *Pseudocolor* color map color.
(2) Also DirectColor, a value.

direct connection
The attachment of a system, workstation, or other I/O device through a selected communication interface and a limited–length cable. No modem is required.

direct–mapped cache
A cache in which exactly one line corresponds to each possible value of the virtual–address field that identifies the line to be interrogated.

direct visibility See *visibility* on page 2-259.

directional component
In AIXwindows, a portion of a compound string that specifies a direction with a given value. The directional component is created with the **XmStringDirectionCreate** function.

directory (1) A type of file containing the names and controlling information for other files or other directories.
(2) A table of identifiers and references to the corresponding items of data.
(3) An index used by a control program to locate blocks of data that are stored in separate areas of a data set in direct access storage.
(4) Contrast with *special file* on page 2-223.
(5) A listing of related files arranged in a useful hierarchy.
(6) In CDS, a logical unit for storing entries under one name (the directory name) in a CDS namespace. In addition to object entries, a directory can contain soft links and child pointers. You can copy, delete, and control access to a directory. Each physical instance of a directory is called a replica.
(7) In XDS, a collection of open systems that cooperate to hold a logical database of information about a set of objects in the real world.
(8) In CDE, a collection of files and other subdirectories. In graphical user interface applications, may be called a *folder*.

directory ID See *directory identifier* on page 2-71.

directory identifier (directory ID)
An identifier for distinguishing several configurations of the directory service within an installation.

directory mask A pattern of characters that controls which portions of a directory will be retained and which portions will not be retained.

disable (1) To make nonfunctional. In interactive communications, to disconnect or stop a subsystem. Contrast with *enable* on page 2-79.
(2) To bring a queue or a device attached to a queue off line so that no print jobs get sent to it.

disabled port In Asynchronous Terminal Emulation (ATE), a port configuration indicating that a port is ready to call out.

DISC Disconnect.

discipline Ordering method used to line up jobs for printing, FCFS (first–come–first–served) or SJN (shortest–job–next). See also *first–come–first–served* on page 2-91 and *shortest–job–next* on page 2-219.

disconnect In X.25 communications, to disconnect a port from the X.25 network.

disconnected mode (DM)
In SDLC, a response from a secondary station indicating that it is disconnected and wants to be online.

discrete type In Ada language, a type that has an ordered set of distinct values. The discrete types are the enumeration and integer types. Discrete types are used for indexing and iteration, and for choices in case statements and record variants.

discretionary access control
A security mechanism that protects information from unauthorized disclosure or modification through owner–controlled access to files. See also *access control list* on page 2-3, *base permission* on page 2-18, and *extended permission* on page 2-87.

discriminant In Ada language, a distinguished component of an object or value of a record type. The subtypes of other components, or even their presence or absence, may depend on the value of the discriminant.

discriminant constraint
In Ada language, on a record type or private type, specifies a value for each discriminant of the type.

discriminated union
In XDR, a C language union that holds several data types, with one arm of the union being an enumeration value, or discriminant, which holds a specific object to be processed over the system first.

disk A storage device made of one or more flat, circular plates with magnetic surfaces on which information can be stored.

disk adapter The hardware used by a computer to access and control disk drives.

disk drive The mechanism used to read and write information on a disk.

disk I/O Fixed–disk input and output.

Disk Operating System
See *DOS* on page 2-74.

disk–usage accounting
The record of the number of disk blocks occupied by a user's files. Disk–usage accounting is performed by the **acctdisk** command.

diskette A thin, flexible magnetic plate that is permanently sealed in a protective cover. It can be used to store information copies from the disk or another diskette.

diskette drive The mechanism used to read and write information on diskettes.

diskless A workstation without local file systems or local boot images that accesses some of its resources remotely. Diskless clients boot remotely from a diskless server and use the server for remote paging.

Diskless Workstation Manager (DWM)
Operating–system software that initializes and maintains resources for diskless clients and diskless servers. It is a group of commands, **awk** command scripts, and source code.

dispatch To allocate processing time on a specific device for a job that is ready to run.

dispatch method

dispatcher In XOM, the software that implements the service interface functions using workspace interface functions.

displacement (1) A positive or negative number that can be added to the contents of a base register to calculate an effective address.
(2) The distance from the beginning of a record, block, or segment to the beginning of a field.

display (1) A visual presentation of data.
(2) To present data visually.
(3) A computer output screen on which visual information is displayed.
(4) In Enhanced X–Windows, a set of one or more screens and input devices that are driven by a single X Server. Synonym for *monitor* on page 2-147.

display device See *display* on page 2-72.

display list (object)
In GL, also called an object. It is a sequence of drawing commands that have been compiled into a unit. Conceptually, a display list is like a macro: it can be invoked multiple times simply by referring to its name. The object can be instantiated at different locations, sizes, and orientations by appropriate use of the transformation matrices. For instance, series of polygons arranged in the shape of a bolt can be compiled into an object. The bolt can then be drawn multiple times by invoking its display list.

Display PostScript (DPS)
Extension to X server created by Adobe.

display screen The part of the display device that displays information visually. Synonymous with *terminal screen*. See also *screen* on page 2-210.

display session
A 3270 Host Connection Program 2.1 and 1.3.2 (HCON) mode of operation during interaction with a host computer that emulates a 3278/79 terminal display.

display station An input/output device that includes a keyboard from which an operator can send information to the system and a display screen on which an operator can see also the information sent to or received from the computer.

display symbol
A predefined printable graphics symbol (such as, characters, numbers, math symbols, Greek letters, and so on) that can be displayed on a graphics display.

display symbol set
A set of display symbols placed in a table. There are up to 1024 display symbols. Display symbols 0 through 31 represent control functions and have no graphic representation.

distinguished encoding
The restrictions to the Basic Encoding Rules designed to ensure a unique encoding of each ASN.1 value, defined in the X.500 Directory Standards (CCITT X.509).

distortion In data communications, an undesirable change in a wave form that can occur between two points in a transmission system. The six major forms of distortion are bias, characteristic, delay, end, fortuitous, and harmonic.

distributed The programs and computerized sources of information that make up a computing environment can be physically located on different computer systems, while still working together as a single logical unit. Transaction processing systems are easily moved to a distributed computing environment, because these systems traditionally involve modifying a centralized source of information by submitting modification or information requests from remote terminals.

Distributed Computing Environment (DCE)
A computing environment in which the resources and data may be located on different processors.

distributed file system
A file system composed of files or directories that physically reside on more than one computer in a communications network.

distributed free space
Synonym for *free space* on page 2-96.

Distributed Function Terminal (DFT)
A workstation that performs operations previously accomplished by the processing unit, such as managing data links, controlling devices, and formatting data.

distributed transaction
A transaction which can update data in many user processes on many machines.

distribution medium
The medium on which the operating system software, a licensed program, or an application program is distributed to the user. The distribution medium can be any of several different media supported by the hardware, such as streaming cartridge tape, 9–track tape, or 3.5–inch diskette.

dithering In computer graphics, a technique of interleaving dark and light pixels so that the resulting image looks smoothly shaded when viewed from a distance.

diversion In text formatting, a command used to save text for printing later in a document, such as for footnotes.

DIX connector A device that connects an Ethernet network adapter to a standard (”thick”) Ethernet local area network (LAN). Its name is derived from the names of the principal developers of Ethernet (Digital Equipment Corporation, Intel, and Xerox).

DLC See *data link control protocol* on page 2-61.

DLE See *data link escape character* on page 2-61.

DLPI Data Link Provider Interface.

DLS Data link service.

DLS provider In X.25 communications, the data link layer protocol that provides the services of the Data Link Provider Interface (DLPI).

DLS user In X.25 communications, the user–level application or user–level or kernel–level protocol that accesses the services of the data link layer.

DLSAP (data–link–service access point)
In X.25 communications, a point at which a data link service (DLS) user attaches itself to a DLS provider to access data link services.

DLSAP address
In X.25 communications, an identifier used to differentiate and locate specific DLS user access points to a DLS provider.

DM See *disconnected mode* on page 2-71.

DO loop In FORTRAN, a range of statements called repetitively by a DO statement.

do statement (1) In C language, a looping statement that contains the keyword **do** followed by a statement (the action), the keyword **while**, and an expression in parentheses (the condition).
(2) A statement used to group a number of statements in a procedure.

DO variable In FORTRAN, a variable, specified in a DO statement, that is incremented or decremented on each iteration of the relative DO loop and controls the number of iterations of the loop.

domain (1) That part of a network in which the data processing resources are under common control.
(2) In a database, all the possible values of an attribute or a data element.
(3) In TCP/IP, the naming system used in hierarchical networks. The domain naming system uses the DOMAIN protocol and the **named** daemon. In a domain system, groups of hosts are administered separately within a tree–structured hierarchy of domains and subdomains.

domain name A level in the hierarchy of names used throughout the Internet.

DOS (Disk Operating System)
A disk operating system used on personal computers.

dot A symbol (.) that indicates the current directory in a relative path name.

dot dot A symbol (..) in a relative path name that indicates the parent directory.

dot matrix A printer with wires or other means that uses a matrix of dots for printing characters.

dotted decimal A common notation for Internet host addresses that divides the 32–bit address into four 8–bit fields. The value of each field is specified as a decimal number and the fields are separated by periods (for example, 010.002.000.052 or 10.2.0.52).

double buffer mode
In GL, a mode in which two buffers are alternately displayed and updated. A new image can be drawn into the back buffer while the front buffer (containing the previous image) is displayed. See also *single buffer mode* on page 2-219.

Double–Byte Character Set (DBCS)
A set of characters in which each character is represented in 2 bytes of storage. Languages such as Japanese, Chinese, and Korean, which contain more symbols than can be expressed in a single byte, are represented in this character set.

double–click To click twice in rapid succession without moving the mouse pointer.

double precision
(1) Pertaining to the use of two computer words to represent a number in accordance with the required precision.
(2) A specification that causes a floating–point value to be stored internally in the long format. See also *precision* on page 2-167.

double–strike A process of printing a character twice to create the appearance of bold type, used frequently with impact printers. A more flexible form of double–strike is emphasized printing. See also *boldface* on page 2-18 and *emphasized* on page 2-79.

double–wide print
A print format in which characters are twice as wide as they normally are.

download To transfer data from one computer for use on another one. Typically, users download from a larger computer to a diskette or fixed disk on a smaller computer or from a system unit to an adapter.

downstream The direction from stream head to driver.

downward jump
For the **bfs** command, the act of moving from the current location in a file toward the bottom or end of the file.

DPS See *Display PostScript* on page 2-72.

DR1I Definite response 1 indicator.

DR2I Definite response 2 indicator.

drag To point to an object with the mouse pointer, hold down the mouse button, move the mouse, and then release the mouse button. A method of ”dragging” an object to a desired point.

DRAM See *dynamic random access memory* on page 2-77.

drawable In Enhanced X–Windows, a collective term for both windows and pixmaps when used as destinations in graphics operations. However, an InputOnly window cannot be used as a source or destination drawable in a graphics operation.

drawn button In AIXwindows, a graphic object that simulates a real–world button with a symbol or other image drawn on its face.

driver The end of a stream closest to an external interface. The principal functions of the driver are handling any associated device and transforming data and information between the external device and stream. It can also be a pseudo–driver, not directly associated with a device, which performs functions internal to a stream, such as a multiplexer or log driver.

driving table A table that describes all the printer–specific information for the **nroff** command.

drop In CDE, after starting the drag of an object, the act of releasing the mouse button. If the object is dropped in an appropriate area, an action is initiated.

drop–in Some applications can detect that an icon has been dropped on to their window to perform an appropriate action, such as opening a document.

drop target In CDE, a rectangular graphic that represents the drop zone in an application.

drop zone In CDE, an area of the workspace, including the Trash Can, Print Manager, and Mailer Front Panel controls, that accepts a dropped icon. Icons can be dropped on the workspace for quick access.

dropped folio A page numbering style in which the page number is printed at the foot of the page. See also *folio* on page 2-93, *blind folio* on page 2-19, and *expressed folio* on page 2-86.

dropping locks Releasing the locks that a transaction holds on data.

DSI See *data storage interrupt* on page 2-61.

DSR Data set ready; used with EIA–232 protocol. See your modem manual for more information.

DTE See *data terminal equipment* on page 2-62.

DTR See *data terminal ready* on page 2-62.

dummy argument
In FORTRAN, a variable within a subprogram or statement function definition with which actual arguments from the calling program or function reference are positionally associated. See also *formal parameter* on page 2-94.

dump (1) To copy the contents of all or part of storage onto another data medium or to an output device.
(2) Data that has been dumped.

dump data The data collected by the kernel dump program. It is obtained from memory locations used by kernel components.

dump table entry
A record in the master dump table that identifies the location of a component dump table. All kernel components that need to have special data collected by the dump program need to generate a dump table entry.

duplex Pertains to communications data that can be sent and received at the same time. Synonymous with *full duplex* and *FDX*. Contrast with *half duplex* on page 2-105.

duplex connector
In a fiber channel link environment, the component that terminates both jumper cable fibers in one housing and provides physical keying for attachment to a duplex receptacle.

duplexed output
Output that uses both the front and back of each sheet of paper for printing.

DWM See *Diskless Workstation Manager* on page 2-72.

dyadic operator
Synonym for *binary operator* on page 2-20.

dynamic A style of creating pop–up menus

dynamic block header
A data structure used by a compiler to link dynamic variables that are in the same heap.

dynamic connection
A connection that is established when needed rather than being predetermined or fixed.

dynamic license
A license that specifies multiple nodelocked licenses. The license is installed at a license server, and then the license server derives license passwords (each of which specifies a single nodelocked license) from the dynamic license. The server automatically installs the individual licenses at the user nodes from which the licensed product is invoked.

dynamic linking
Linking of a program in which library procedures are not incorporated into the load module but are dynamically loaded from their library each time the program is loaded.

dynamic random access memory
A storage in which the cells require repetitive application of control signals to retain stored data. Such repetitive application of the control signals is normally called a refresh operation.

dynamic routing
A method of setting paths between hosts, networks, or both by using daemons that update the routing table as needed.

dynamic string See *string* on page 2-228.

dynamic variable
A variable allocated only when needed. Explicit allocations and deallocations are required. In Pascal, the predefined procedures NEW and DISPOSE are provided for this purpose.

dynamic window
A variable that can change dynamically within a certain window or range of values.

E

Easy Install An application used to install optional software or service updates in the form of software bundles.

EBCDIC See *extended binary–coded decimal interchange code* on page 2-86.

EBCDIC character
Any one of the symbols included in the 8–bit EBCDIC set.

EC Engineering Change level.

ECB (1) See *event control bit* on page 2-83.
(2) Event control block.
(3) Electronic codebook.

ECC (1) See *error–checking and correction* on page 2-82.
(2) Error correction code.

echo (1) A reflected signal on a communications channel. On a communications terminal, each signal is displayed twice, once when entered at the local terminal and again when returned over the communications link. This allows the signals to be checked for accuracy.
(2) In computer graphics, the immediate notification of the current values provided by an input device to the operator at the display console.
(3) In word processing, to print or display each character or line as it is typed in.

ECM Entity coordination management.

ECMA European Computer Manufacturers' Association.

edit (1) To add, change, delete, rearrange, or modify the form or format of data.
(2) To check the accuracy of information and to indicate if an error is found.

edit buffer A temporary storage area used by an editor.

effective rate The average sustained speed at which a device operates under real–world conditions, when processing a representative workload.

effective root directory
The point where a system starts when searching for a file. Its path name begins with a / (slash). The **chroot** subroutine causes the directory named by the path parameter to become the effective root directory.

effective user ID
(1) The user ID associated with the last authenticated user or the last setuid program. Equal to either the real or the saved user ID.
(2) The current user ID, but not necessarily the user's login ID. For example, a user logged in under a login ID may change to another user's ID. The ID to which the user changes becomes the effective user ID until the user switches back to the original login ID. All discretionary access decisions are based on the effective user ID.

EGA The Enhanced Graphics Adapter.

EGP Exterior Gateway Protocol. The mechanism that allows the exterior gateway of an autonomous system to share routing information with exterior gateways on other autonomous systems.

EIA Electronic Industries Association.

EIA–232D An EIA interface standard that defines the physical, electronic, and functional characteristics of an interface line that connects a communication device and associated workstation. It uses a 25–pin connector and an unbalanced line voltage.

EIA–422A An EIA interface standard that defines the physical, electronic, and functional characteristics of an interface line connecting a computer to communications equipment. It uses a balanced line voltage for noise reduction and longer distance capability. The system unit uses the send and receive pins from the set of 40 pins defined by the EIA–422A interface.

elaborated type specifier
Typically used in C++ in an incomplete class declaration or to qualify types that are otherwise hidden.

elaboration In Ada language, the elaboration of a declaration is the process by which the declaration achieves its effect (such as creating an object); this process occurs during program execution.

electromagnetic compatibility
The design and test of products to meet legal and corporate specifications dealing with the emissions and susceptibility to frequencies in the radio spectrum. Electromagnetic compatibility is the ability of various electronic equipment to operate properly in the intended electromagnetic environment.

electronic mail Synonym for *mail* on page 2-139.

element (1) The smallest unit of data in a table or array.
(2) In a set, an object, entity, or concept having the properties that define a set. Synonymous with *member* on page 2-143.
(3) In SNA, the particular resource within a subarea that is identified by an element address.
(4) The component of an array, subrange, enumeration, or set.
(5) In AIXwindows, an object or similar data structure having the properties that define a class.

else clause The part of an **if** statement that contains the keyword **else** followed by a statement. The **else** clause provides an action that is started when the **if** condition evaluates to a value of 0 (false).

em In a specific type size, an em is the same number of points as there are to that specific size. See also *en* on page 2-79.

embedded blanks
Blanks that are surrounded by any other characters.

EMC See *electromagnetic compatibility* on page 2-79.

emphasized A form of double–strike printing in which characters are printed in multiple passes (usually two) with a slight offset, creating an artificial bold type. Emphasized printing is used to fill gaps and rough appearance in dot–matrix character forms. It also prints a bold font without changing the mounted character set. See also *boldface* on page 2-18 and *double–strike* on page 2-75.

emulation (1) The use of programming techniques and special machine features to permit a computing system to run programs written for another system.
(2) Imitation. For example, when one computer imitates the characteristics of another computer.

emulator A program that causes a computer to act as a workstation attached to another system.

en In the **troff** command, an en is half the size, in points, of an em. In the **nroff** command, an em and an en are identical size. See also *em* on page 2-79.

enable (1) To make functional.
(2) In interactive communications, to load and start a subsystem. Contrast with *disable* on page 2-71.

encapsulation Hiding the internal representation of data objects and implementation details of functions from the client program. This enables the end user to focus on the use of data objects and functions without having to know about their representation or implementation.

encrypt To convert clear data into cipher text.

encryption key A key generated by the **makekey** command to use with programs that perform encryption. Its input and output are usually pipes.

end signal In an online conference, a mutually agreed upon character that indicates the end of a comment by a participant. Common end signals are **o** and **oo**.

endian An attribute of data representation that reflects how certain multi–octet data are stored in memory. See also *big endian* on page 2-20 and *little endian* on page 2-131.

end–to–end transit delay
In X.25 communications, an optional CCITT–specified facility.

enforced lock A type of lock that a process holds on a region of a file preventing any other process from accessing that region with read or write system calls. In addition, the **create** command is prevented from truncating the files. See also *advisory lock* on page 2-6.

enhanced graphics adapter (EGA)
An adapter, such as the Enhanced Graphics Adapter, that provides high–resolution graphics, allowing the use of a color display for text processing as well as graphics applications.

Enhanced X-Windows
A collection of basic functions for developing a variety of application environments. Toolkit functions manage Toolkit initialization, widgets, memory, events, geometry, input focus, selections, resources, translation of events, graphics contexts, pixmaps, and errors. See also AIXwindows Environment on page 2-7.

enqueue To place items in a queue. Contrast with *dequeue* on page 2-66.

enter (1) To send information to the computer by pressing the Enter key.
(2) To place a message on the line to be transmitted from a terminal to the computer.

Enterprise Systems Connection
See ESCON on page 2-83.

entry (1) In FORTRAN, a language construct within a procedure, designating the start of the execution sequences of the procedure.
(2) In Ada language, an entry is used for communication between tasks. Externally, an entry is called just as a subprogram is called; its internal behavior is specified by one or more accept statements specifying the actions to be performed when the entry is called.
(3) An element of information in a table, list, queue, or other organized structure of data or control information. (4.) A single input operation on a workstation.

entry poin (1) An address or label of the first instruction performed upon entering a computer program, routine, or subroutine. A program may have several different entry points, each corresponding to a different function or purpose.
(2) In a routine, any place to which control can be passed.

entry point vector (EPV)
A record in which fields are pointers to procedures that implement the operations defined by an interface.

entry sequence number (ESN)
The number corresponding to the order in which the record was entered into an entry–sequenced file. The primary index of an entry–sequenced file is based upon the ESNs of its records. See also *entry–sequenced file* on page 2-81.

entry–sequenced file
A record–oriented file in which the records are stored in the order in which they were entered into the file. The primary index of an entry–sequenced file is based on the entry sequence numbers (ESNs) corresponding to the order of record insertion. Also referred to as *sequential file*. See also *entry sequence number* on page 2-81.

enumerated scalar type
A scalar defined by enumerating the elements of the type. Each element is represented by an identifier.

enumeration constant
In C language, an identifier (with an associated integer value) defined in an enumerator. You can use an enumeration constant anywhere an integer constant is allowed.

enumeration data type
A type that represents integers and a set of enumeration constants. Each enumeration constant has an associated integer value.

enumeration tag
The identifier that names an enumeration data type.

enumeration type
In Ada language, a discrete type whose values are represented by enumeration literals which are given explicitly in the type declaration. These enumeration literals are either identifiers or character literals.

enumerator An enumeration constant and its associated value.

environment (1) The settings for shell variables and paths set when the user logs in. These variables can be modified later by the user.
(2) A named collection of logical and physical resources used to support the performance of a function.
(3) In Common Desktop Environment, the set of rule files, resources, and message files that define the appearance and behavior of a specific desktop configuration.

environment variable
(1) A variable that describes the operating environment of the process. Common environment variables describe the home directory, command search path, the terminal in use, and the current time zone (the HOME, PATH, TERM, and TZ variables, respectively).
(2) A variable that is included in the current software environment and is therefore available to any called program that requests it.

EOF End of file.

ephemeral application
An ephemeral application is one which does not contain any recoverable data.

epoch A timestamp that identifies directory replicas as being part of the same set.

EPOW Emergency power off warning.

EPROM Erasable programmable read–only memory.

EPV See *entry point vector* on page 2-80 or *Entry Point Vector*.

equivalence class
A grouping of characters or character strings that are considered equal for purposes of collation. For example, many languages place an uppercase character in the same equivalence class as its lowercase form, but some languages distinguish between accented and unaccented character forms for the purpose of collation.

erase To remove text from a data medium, leaving the medium available for recording new text. See also *delete* on page 2-65.

erase character
A character that indicates that the previous character on the command line has been erased.

error analysis facility
A program that provides information about the probable cause of errors.

error–checking and correction (ECC)
In a processing unit, the detection and correction of all single–bit errors, plus the detection of double–bit and some multiple–bit errors.

error class Identifies whether an error log entry is for a hardware or software failure.

error condition The state that results from an attempt to run instructions in a computer program that are not valid or that operate on data that is not valid.

error counter A type of error entry generated by device driver components. Certain device drivers can generate retry operations if an operation is not successful on the first attempt. They use counters to monitor the number and cause of retry operations, and they contain algorithms that decide when these counters should be sent to the error log.

error device driver
A special file (psuedo device) driver used by the error logging facilities. Error entries are written to the error device driver by the **errlog()** subroutine and the **errsave()** kernel service. Error entries are read from the error device driver by the error daemon process and saved in the system error log file.

error ID See *error identifier* on page 2-82.

error identifier (error ID)
An 8–character code used to identify a particular failure. There is a unique error identifier for each error record template.

error log (1) A data set or file in a product or system where error information is stored for later access.
(2) A form in a maintenance library that is used to record error information about a product or system.
(3) A data set used in a processor to record information about certain hardware and programming events
(4) A record of machine checks, device errors, and volume statistical data.

error log entry A record in the system error log describing a hardware or software failure and containing failure data captured at the time of the failure.

error message An indication that an error has been detected.

error output Synonym for *diagnostic output* on page 2-69.

error record template
Describes the error class, error type, error description, probable causes, recommended actions, and failure data for an error log entry.

error type Identifies whether an error log entry is for a permanent failure, temporary failure, performance degradation, impending loss of availability, or undetermined failure.

ESC See *escape character* on page 2-83.

escape character (ESC)
(1) In shell programming and TTY programming, the \ (backslash) character, which indicates that the next character is not intended to have the special meaning normally assigned to it.
(2) In general, a character that suppresses or selects a special meaning for one or more characters that follow.

escape function
The code of the form 'ESC *Something* '. One type of multibyte control function.

escape sequence
(1) A character that is preceded by a \ (backslash) and is interpreted to have a special meaning to the operating system.
(2) A sequence sent to a terminal to perform actions such as moving the cursor, changing from normal to reverse video, and clearing the screen. The **terminfo** file defines these escape sequences. Synonymous with *multibyte control*.
(3) Multicharacter code specifying a string variable. See also *unescaped* on page 2-251.

ESCD console The ESCD input/output device used to perform connectivity tasks at the ESCD.

ESCON (Enterprise Systems Connection)
A System/390 I/O architecture that uses both laser and LED fiber–optic technology and the concept of dynamic connectivity, which allows for path sharing.

ESDI Enhanced Small Device Interface.

ESN See *entry sequence number* on page 2-81.

Ethernet A 10–megabit baseband local area network using CSMA/CD (carrier sense multiple access with collision detection). The network allows multiple stations to access the medium at will without prior coordination, avoids contention by using carrier sense and deference, and resolves contention by using collision detection and transmission.

evaluation In Ada language, the evaluation of an expression is the process by which the value of the expression is computed. This process occurs during program execution.

event (1) The enqueueing or dequeueing of an element.
(2) An occurrence of significance to a task.
(3) In computer graphics, information generated either asynchronously from a device or as the side–effect of a client request. Events are grouped into types and are not sent to a client by the server unless the client has issued a specific request for information of that type. Events are usually reported relative to a window. See also *callback* on page 2-29.

event class A number assigned to a group of trace points that relate to a specific subject or system component. The defined event classes are listed in the trace profile.

event control bit (ECB)
A bit assigned to each queue to signal the arrival or departure of an element.

event loop A sequence of steps performed cyclically to accomplish a task. It must contain at least one input (source) and an ending point.

event mask In computer graphics, the set of event types that a client requests relative to a window.

event queue In computer graphics, a queue that records changes in input devices—buttons, valuators, and the keyboard. The event queue provides a time–ordered list of input events.

exception (1) In programming languages, an abnormal situation that may arise during the running of a program, perhaps causing a deviation from the normal run sequence, and for which handling facilities exist.
(2) An abnormal condition such as an I/O error encountered in processing a data set or a file.
(3) One of five types of errors that can occur during a floating–point exception. These are an operation that was not valid, overflow, underflow, division by zero, and inexact results.
(4) In C++ language, any user, logic, or system error detected by a function that does not itself deal with the error but passes the error on to a handling routine. Passing this error is called throwing an exception.
(5) In Ada language, an error situation which may arise during program execution. To raise an exception is to abandon normal program execution so as to signal that the error has taken place. An exception handler is a portion of program text specifying a response to the exception. Execution of such a program text is called handling the exception. Contrast with *interrupt* on page 2-120 and *signal* on page 2-219. See also *handler* on page 2-105 and *trap handler* on page 2-246.

exception handler
(1) A set of routines used to detect deadlock conditions or to process abnormal condition processing. An exception handler allows the normal running of processes to be interrupted and resumed.
(2) Exception handlers are **catch** blocks in C++. **catch** blocks catch exceptions when they are thrown from a function enclosed in a try block. **try** blocks, **catch** blocks and **throw** expressions are the constructs used to implement formal exception handling in C++.

exception handling
A type of error handling that allows control and information to be passed to an exception handler when an exception occurs. **try** blocks, **catch** blocks, and **throw** expressions are the constructs used to implement formal exception handling in C++.

exchange identification (XID)
The ID that is exchanged with the remote physical unit when an attachment is first established.

exchange identification (XID) frame
In a logical link control (LLC) header, the frame that conveys the characteristics of the sending host.

exclusive lock A type of lock in which only the transaction holding the lock can access the data in any way. See also *shared locks* on page 2-217.

exec To overlay the current process with another executable program. See also *fork* on page 2-94.

EXEC Remote Command Execution Protocol.

executable A file that can be loaded into memory and executed as a program. An executable is produced by the binder (**ld**) from one or more object (**.o**) files. The default processing of compilation commands includes invoking the binder to produce an executable whose name is **a.out**.

executable file A file that contains programs or commands that perform operations on actions to be taken.

executable program
A program that can be run as a self–contained procedure. It consists of a main program and, optionally, one or more subprograms.

executable statement
A statement that causes an action to be taken by the program. For example, to calculate, to test conditions, or to alter normal sequential execution.

execute In CDE, a file or folder (directory) access permission. For files, execute permission enables the user to run a program or shell script file. For folders (directories), execute permission enables the user to access the folder (directory) contents.

existing file A file that has been defined and that resides on a storage medium.

Exit button In CDE, a Front Panel control used to log out of the desktop.

exit value (1) A code sent to either standard output or standard error on completion of the command.
(2) A numeric value that a command returns to indicate whether it completed successfully. Some commands return exit values that give other information such as whether a file exists. Shell programs can test exit values to control branching and looping.

expanded name
In Ada language, a name that denotes an entity which is declared immediately within some construct. An expanded name has the form of a selected component: the prefix denotes the construct (a program unit; or a block, loop, or accept statement); the selector is the simple name of the entity.

expect–send sequence
In remote communications, a list of characters or signals a program or modem should expect to receive from a remote system, followed by the characters or signals the program or modem should send to the remote system after it receives the expected input. The sequence can also include a subsequence that tells the program or modem what to send if it does not receive the expected input. See also *handshaking* on page 2-105 and *chat script* on page 2-35.

expedited data negotiation
In X.25 communications, an optional CCITT–specified facility.

expedited data transfer
In X.25 communications, an optional CCITT–specified facility.

explicit binding
A form of binding that gives the client application control over what server machines are involved.

explicit shell Any of a class of shells created manually by the developer through the Create menu's Shells submenu.

exponent A number indicating the power to which another number (the base) is to be raised.

exponentiation The operation in which a value is raised to a power.

export (1) In NCS, to provide the operations defined by an interface. A server exports an interface to a client. Contrast with *import* on page 2-111.
(2) To copy data onto removable media.

exposure event
In Enhanced X–Windows, an event sent to clients to inform them when contents have been lost, as when windows are obscured or reconfigured. Servers do not guarantee the preservation of window contents when they are obscured or reconfigured.

expressed folio
A page numbering style in which each page (possibly excluding the first page) is numbered. See also *folio* on page 2-93,

blind folio on page 2-19, and *dropped folio* on page 2-76.

expression (1) A representation of a value. For example, variables and constants appearing alone or in combination with operators.
(2) In programming languages, a language construct for computing a value from one or more operands, such as literals, identifiers, array references, and function calls. In Ada language, an expression defines the computation of a value.
(3) A configuration of signs.

expression statement
In C language, an expression that ends with a ; (semicolon). You can use an expression statement to assign the value of an expression to a variable or to call a function.

extended address
Synonym for *address extension* on page 2-6.

extended binary–coded decimal interchange code (EBCDIC)
A code developed for the representation of textual data. EBCDIC consists of a set of 256 eight–bit characters.

extended character
A character other than a 7–bit ASCII character. An extended character can be a 1–byte code point with the eighth bit set (ordinal 128 through 255). See also *code page* on page 2-40 and *code point* on page 2-40.

extended common object file format (XCOFF)
The object file format for Version 3 of the operating system. XCOFF combines the standard common object file format (COFF) with the TOC module format concept, which provides for dynamic linking and replacement of units within an object file.

extended curses
A system library(the **libcurses.c** library) that contains the control functions for writing data to and getting data from the terminal screen. It supports color, multiple windows, and an enhanced character set.

extended family
The set of all of an Ada–language compilation unit's supporters and all of their families, including the associated bodies and subunits. The supporters of a unit include only the library units required for its compilation, not the associated bodies or subunits.

extended font (1) A font in which the characters are wider than its corresponding normal font.
(2) A collection of font sections to support languages requiring more than 256 graphic characters at one time.

extended interface
Provides a set of full–function system calls (the **readx** and **writex** system calls) to communicate with SNA Server. These calls contain an extra parameter on the call (a pointer to the structure containing extra function requests). See also *interface* on page 2-119 and *limited interface* on page 2-129.

extended permission
An access mode that modifies the base permissions to a file for specified individuals or groups. An extended permission can deny or permit an access mode. See also *discretionary access control* on page 2-71.

extended precision constant
A processor approximation to the value of a real number that occupies 16 consecutive bytes of storage and can assume a positive, negative, or zero value. The precision is greater that of type double precision.

extended result
An exception notification that has data defined in the **result_ext** field of the **dlc_getx_arg** structure.

extended selection
In CDE, to add multiple items to a selected set by augmenting the selection technique. For example, you can add to a selection by moving the pointer to a new item and pressing Control and the SELECT mouse button simultaneously.

Extended Services
A group of optionally installed operating system functions and programs.

extension In Enhanced X–Windows, to extend the system, the named extensions can be defined for the Core protocol, including extensions to output requests, resources, and event types.

extent A continuous space on disk or diskette that is occupied by or reserved for a particular data set, data space, or file.

exterior gateway
A gateway on one autonomous system that communicates with other autonomous system.

external clocking
In data communications, the ability of a modem to provide data clocking.

external data definition
A description of a variable appearing outside a function. It causes the system to allocate storage for that variable and makes that variable accessible to all functions that follow the definition and are located in the same file as the definition.

eXternal Data Representation (XDR)
A standard for the description and encoding of data; it uses a languages to describe data formats, but the language is used only for describing data and is not a programming language. Protocols such as RPC and NFS use XDR to describe their data formats.

external function
In FORTRAN, synonymous with *external routine* on page 2-88.

external modem
A modem that is separate from the unit with which it operates.

external name (1) A name that can be referred to by any control section or separately assembled or compiled module; a control section name or an entry name in another module.
(2) In a program, a name whose scope is not necessarily confined to one block and its contained blocks.

external procedure
Synonymous with *external routine* on page 2-88.

external reference
A reference to a symbol defined as an external name in another program or module.

external routine
A procedure or function called from outside the program in which the routine is defined. Synonymous with *external procedure*.

external symbol
A symbol that is defined in a file other than the file in which the symbol occurs. An ordinary symbol that represents an external reference.

external variable
A variable accessible to another compilation unit. See also *compilation unit* on page 2-45.

extract To obtain. For example, to extract information from a file.

eye coordinates
In GL, the coordinate system in which the viewer's eye is located at the origin, and thus all distances are measured with respect to the eye. Viewing transformations map from world coordinates into eye coordinates, and projection transformations map from eye coordinates to normalized device coordinates. Synonym for *eye space*, *viewing coordinates*, and *viewer coordinates*. See also *modeling coordinates* on page 2-146, *primitive coordinates* on page 2-184, *world coordinates* on page 2-265, *screen coordinates* on page 2-210, and *transformation* on page 2-245.

eye space Synonym for *eye coordinates* on page 2-88.

F

F Fahrenheit.

facilities See *optional facilities* on page 2-162.

facilities extension
In the X.25 API, an extension to the optional facilities field in a packet that allows further, non–CCITT–specified, optional facility information to be added.

factorization An arithmetic transformation in which common factors are removed from subexpressions within an expression, and then multiplied by the resulting expression.

fake target nam
A control name used in a **makefile** file that looks like a target name, but actually tells the **make** command to perform some operation differently.

family An Ada–language library unit together with its body and subunits (if any). Not all potential family members need be present; a library unit may have a missing optional body, and a main program procedure may have only an implicit specification. The family of a secondary unit is that unit together with any subunits.

fast select In X.25 communications, an optional facility that allows inclusion of data in call–request and clear–request packets. See also *optional facilities* on page 2-162.

FCC Federal Communications Commission.

FCFS See *first–come–first–served* on page 2-91.

FD Full duplex. See *duplex* on page 2-76.

FDDI Fiber–optic Distributed Data Interface. A 100–Mbit/sec optical LAN interface. An ANSI standard for a high–speed, 100–Mbit/sec, general–purpose network for the interconnection of computers, networks, and peripheral equipment using optical fiber cable in a dual–ring configuration. FDDI can connect as many as 500 stations with a maximum link–to–link distance of 2 kilometers and a total LAN circumference of 100 kilometers.

FDX Full duplex. See *duplex* on page 2-76.

feature A programming or hardware option, usually available at an extra cost.

fiber optics The branch of optical technology concerned with the transmission of radiant power through fibers made of transparent materials such as glass, fused silica, and plastic.

field (1) An area in a record or panel used to contain a particular category of data.
(2) The smallest identifiable component of a record. An individually addressable subdivision of a record containing a specific portion of the data in the record. For example, a record of data about an employee might be subdivided into fields containing the employee's name, identification number, and salary.
(3) An area in a presentation space into which the program accepts input.
(4) In video, a field is one–half of a frame. Fields are drawn 60 times per second. See also *structured field* on page 2-229.

field of view In GL, the extent of the area which is under view. The field of view is defined by the *viewing matrix* on page 2-257 in use.

field return The action that moves a data cursor from field to field in a reverse direction, as determined by the panel layout.

FIFO See *first–in–first–out* on page 2-91.

file (1) A collection of related data that is stored and retrieved by an assigned name. Contrast with *special file* on page 2-223.
(2) A sequence of records. If the file is located in internal storage, it is an internal file; if it is on an input/output device, it is an external file.
(3) A collection of related information stored in a single location for organizational purposes and processed as a unit. See also *filename* on page 2-91.

file descriptor A small positive integer that the system uses instead of the file name to identify an open file. See also *shared memory ID (shmid)* on page 2-217.

file format A description of the entries to be made in an ASCII file, such as a configuration or customization profile.

file index 64 bytes of information describing a file. The file index contains the type and size of the file and the location of the file data on the physical device. This index is the same as the operating system i–node. Synonym for *i–node* on page 2-114.

file manager A program that manipulates files or directories.

file memory Virtual–memory pages that are currently in real memory that are not part of computational memory. Normally these are pages of nonexecutable files.

file mode creation mask
See *mask* on page 2-141.

file name (1) A name assigned or declared for a file.
(2) The name used by a program to identify a file. See also *label* on page 2-126.

file name substitution
The process in which the shell recognizing a word (character string) that contains any of the *, ?, [, or { characters, or begins with the ~ character, and replaces it with an alphabetically sorted list of file names that match the pattern of the word. Synonymous with *globbing*.

file owner The user who has the highest level of access authority to a file, as defined by the file.

file pointer An identifier that indicates a structure containing the file name.

file scope A C++ name declared outside all blocks and classes has file scope and can be used after the point of declaration in a source file.

file server In CDE, a host computer that stores data files used by applications.

file spec See *file specification* on page 2-90.

file specification (filespec)
he name and location of a file. A file specification consists of a drive specifier, a path name, and a file name.

file suffix In CDE, a suffix added to the end of a file name, often used in file typing or to categorize files for the user.

file system The collection of files and file management structures on a physical or logical mass storage device, such as a diskette or minidisk.

file transfer In remote communications, the transfer of a file or files from one system to another over a communications link.

File Transfer Protocol (FTP)
In TCP/IP, the protocol that makes it possible to transfer data among hosts and to use foreign hosts indirectly.

file tree The complete directory and file structure of a particular node, starting at the root directory. A file tree contains all local and remote mounts performed on directories and files.

file type In the operating system, one of the five possible types of files: ordinary file, directory, block device, character device, and first–in–first–out(FIFO or named pipe).

filename In DOS, that portion of the file name that precedes the extension. See also *file* on page 2-90.

fileset (1) An individually installable option or update. Options provide specific function and updates correct an error in, or enhance, a previously installed option.
(2) One or more separately installable, logically grouped units in an installation package.

filetab A kernel parameter establishing the maximum number of files that can be open simultaneously.

fill characters (1) Characters used to fill fields in storage.
(2) Visual representations of enterable character positions on the display (for example, dots in each position or vertical bars between positions).

filter (1) A command that reads standard input data, modifies the data, and sends it to the display screen.
(2) A device or program that separates data, signals, or materials in accordance with specified criteria.

filter primitive A program that separates data in accordance with specified criteria.

fine clipping In GL, fine clipping masks all drawing commands to a rectangular region of the screen. It would be unnecessary except for the case of character strings. The origin of a character string after transformation may be clipped out by gross, or 3–D, clipping, and the string would not be drawn. By doing gross clipping with the viewport and fine clipping with the screen masks, strings can be moved smoothly off the screen to the left or bottom. See also *clipping* on page 2-38 and *gross clipping* on page 2-103.

FINGER Name/Finger Protocol. An application–level Internet protocol that provides an interface between finger command and the fingered daemon.

firewall A system or machine that controls the access between outside networks and private networks.

first–come–first–served (FCFS)
In general, a queuing technique in which the next item to be retrieved is the item that has been in the queue for the longest time. Synonym for *first–in–first–out* on page 2-91. Contrast with *shortest–job–next* on page 2-219. See also *discipline* on page 2-71.

first–in–first–out (FIFO)
In the operating system, a named permanent pipe. A FIFO allows two unrelated processes to exchange information through a pipe connection. Synonym for *first–come–first–served* on page 2-91.

first–level interrupt handler (FLIH)
A routine that receives control of the system as a result of a hardware interrupt. One FLIH is assigned to each of the six interrupt levels.

fix number The fix level of a program, which is an indicator of small updates that are to be built into a regular modification or release at a later time. The version, release, modification, and fix levels together comprise the program level or version of a program. See also *program level* on page 2-187, *modification number* on page 2-141, *release number* on page 2-199, *version* on page 2-256, and *version number* on page 2-257.

fixed box (1) A geometry management technique.
(2) A type of bounding box that has a fixed number of children created by the parent. These managed children do not make geometry manager requests.

fixed disk (1) A flat, circular, nonremovable plate with a magnetizable surface layer on which data can be stored by magnetic recording. A rigid magnetic disk used in a fixed–disk drive.
(2) The term fixed disk is also used loosely in the industry for boards and cartridges containing microchips or bubble memory that simulate the operations of a fixed–disk drive.

fixed–disk drive
The mechanism used to read and write information on a fixed disk.

fixed icon An icon that is fixed on the desktop. Fixed icons are ordinarily used for files that are necessary to use Common Desktop Environment and therefore always appear on the desktop, such as the Home directory.

fixed part (of a record)
In Pascal, the part of a record that is common to all instances of a particular record type.

fixed point type
See *real type* on page 2-196.

fixed storage space
Any storage device defined during system configuration to be an integral part of system DASD. If a fixed storage device is not available at some time during normal operation, the operating system detects an error.

flag (1) A modifier that appears on a command line with the command name that defines the action of the command. A dash usually precedes a flag.
(2) An indicator or parameter that shows the setting of a switch.
(3) A character that signals the occurrence of some condition, such as the end of a word.
(4) An internal indicator that describes a condition to the processing unit. In the processing unit of the PC AT, flags indicate if the result of an operation is zero, if interrupts are enabled, and other conditions. Synonymous with *condition code*.

flat file (1) A file that has no hierarchical structure.
(2) A one–dimensional or two–dimensional array. A list or table of items.

flat network A network in which all hosts are administered by one central authority.

flattened character
An ASCII character created by translating an extended character to the ASCII character most like it. The code point information is lost and the character cannot be retranslated to an extended character. For example, a cedilla character would be flattened to a plain ”c.”

FLIH See *first–level interrupt handler* on page 2-91.

float constant A number containing a decimal point, an exponent, or both a decimal point and an exponent. The exponent contains an ”e” or ”E,” an optional sign (+ or –), and one or more digits (0 through 9).

floating display
In text formatting, a block of text that the **nroff** command keeps on one page. However, if there is no room for a floating display on the current page, the **nroff** command sets aside the display and finishes filling the page with the text from the input file that follows the display. When the page is full, the **nroff** command places the display at the top of the next page and then continues with the text from the previous page. See also *static display* on page 2-226.

floating keep A keep ensures that bracketed text is not broken across a page boundary, by comparing the size of the text block against the space remaining on the page. If it does not fit, the block is printed at the first possible opportunity on the following page. A fixed keep leaves the remainder of the page blank and a floating keep moves subsequent text onto that page.

floating license
See *concurrent–use license* on page 2-48.

floating point A way of representing real numbers (that is, values with fractions or decimals) in 32 bits or 64 bits. Floating–point representation is useful to describe very small or very large numbers.

floating–point constant
A constant representation of a floating–point number expressed as an optional sign followed by one or more digits and including a decimal point. See also *floating–point number* on page 2-93.

floating–point exception
See *exception* on page 2-84.

floating–point number
A real number represented by a pair of distinct numerals. The real number is the product of the fractional part, one of the numerals, and a value obtained by raising the implicit floating–point base to a power indicated by the second numeral. See also *floating–point constant* on page 2-93.

floating point type
See *real type* on page 2-196.

flow control In X.25 communications, the procedure for controlling the data transfer rate.

flow diagram Deprecated term for flow chart.

flyback A movement similar to a reversing line feed.

FM See *frequency modulation* on page 2-96.

FM Header Function Management Header.

focus window Synonym for *input focus* on page 2-115.

fold (1) To compact data by combining parts of the data; for example, to transform a two–word numeric key by adding the numeric equivalents of the letters.
(2) To translate the lowercase characters of a character string into uppercase. See also *constant folding* on page 2-51.
(3) To place on the next line a portion of a line that does not fit on the line. Contrast with *truncate* on page 2-247.

folder (1) In Message Handler, a directory of messages.
(2) In CDE, an icon that represents a *directory*.

folio A page numbering format that places the page number at the outside of the running head at the top of the page. See also *blind folio* on page 2-19, *dropped folio* on page 2-76, and *expressed folio* on page 2-86.

font (1) A set of characters in a particular style. See also *raster font* on page 2-194 and *primitive font* on page 2-184.
(2) In Enhanced X–Windows, a set of glyphs, usually characters. The protocol does not translate or interpret character sets. The client indicates values used to access the glyph arrays. A font contains additional metric information to determine inter–glyph and inter–line spacing.
(3) A complete set of graphic characters of the same size, style, and typeface. For example, a Times 12 point font is different in size from a Times 24 point font, different in style from a Helvetica 12 point font, and different in typeface from a Times 12 point boldface font.

font ascent Height of a character in a specified font.

font list In AIXwindows, a list of available fonts specified by the *fontlist* parameter. Also, the list of fonts to be used in formatting a source document. See also *font table* on page 2-94.

font structure A data structure that contains all the information necessary to create a font set.

font table A list of all fonts that have been loaded into the system. See also *font list* on page 2-94.

font unit Any group of characters regarded as a whole, especially for the determination of values, variables, and other data pertaining to font characteristics.

footer Text that appears at the bottom of every page of a document, for example, a page number.

footnote A note of reference, explanation, or comment placed below the text of a column or page, but within the body of the page above the footer.

footnote text The text within the footnote.

for statement In programming languages, a statement that executes one or more statements for each of a set of values assigned to one or more variables.

foreground (1) A mode of running a program in which the shell waits for the program specified on the command line to complete before responding to user input.
(2) In multiprogramming, the environment in which high–priority programs are run. Contrast with *background* on page 2-17.

foreground color
A single color assigned to all of the graphic elements that appear in front of all the background graphic elements within a displayed image. Contrast with *background color* on page 2-17.

foreground process
A process that must run to completion before another command is issued to the shell. The foreground process is in the foreground process group, which is the group that receives the signals generated by a terminal. Contrast with *background process* on page 2-17.

foreign cell A cell other than the one to which the local machine belongs. See also *local cell* on page 2-132.

foreign host Synonym for *remote host* on page 2-200.

fork To create and start a child process. See also *exec* on page 2-84.

formal parameter
A parameter declared in a routine heading. It specifies what can be passed to a routine as an actual parameter. See also *conformant string* on page 2-49 and *dummy argument* on page 2-76. For Ada programming, see also *parameter* on page 2-169. Contrast with *actual parameter* on page 2-5.

format (1) A defined arrangement of such things as characters, fields, and lines, usually used for displays, printouts, or files.
(2) The pattern that determines how data is recorded.
(3) To arrange such things as characters, fields, and lines.
(4) In programming languages, a language construct that specifies the representation, in character form, of data objects in a file. See also *formatted data* on page 2-94.

formatted data Data that is transferred between main storage and an input/output device according to a specified format. See also *list–directed data* on page 2-131 and *format* on page 2-95.

formatted diskette
A diskette on which track and control information for a particular computer system has been written but that may or may not contain any data.

formatter (1) A computer program that prepares a source document for printing.
(2) The part of a text processor that formats input lines for printing or display on a particular type of device.

FORTRAN (FORmula TRANslation)
A high–level programming language used primarily for scientific, engineering, and mathematical applications.

forward difference matrix
In GL, 4x4matrix that is iterated by adding each row to the next and the bottom row is output as the next point. Points so generated generally fall on a rational cubic curve.

FP See *FPU* on page 2-95.

FPU (FP) Floating–point unit.

fragment A unit of disk storage that is smaller that a (4KB) page.

frame (1) In a high–level data link control (HDLC), the sequence of contiguous bits bracketed by and including opening and closing flag (01111110) sequences.
(2) A set of consecutive digit time slots in which the position of each digit time slot can be identified by reference to a frame alignment signal.
(3) In an interlaced video monitor, a frame consists of two fields, called ”even” and ”odd”, each of which is one sixtieth of a second in duration.
(4) In block–multiplexer–channel–adapter and ESCON transmissions, the block of data that is transferred between the S/370 and the workstation in one CCW.
(5) A block of information corresponding to the FDDI protocol which is transmitted between two or more stations on a ring. An FDDI frame is similar to a packet.

frame buffer A quantity of video RAM (VRAM) that is used to store the image displayed on the monitor. The frame buffer is the electronic canvas on which every drawing primitive is drawn. It is one of the last stops in the graphics pipeline, where the final image resides in the form of digitally coded intensities and brightnesses. These are converted into analog voltage signals 60 times a second and sent to the electron guns of the monitor. The dimensions of the frame buffer can be changed with GL. Typically, the main frame buffer might be 1024 pixels vertical by 1280 pixels horizontal by 8 color bits. The overlay planes might by 1024x1280x2. The z–buffer is considered a frame buffer, although it is not directly visible from the monitor. (There is no direct means of displaying the contents of the z–buffer, although this can be done indirectly.) The size of the z–buffer is typically 1024x1280x24. The cursor is a very specialized form of a frame buffer; one which can move around. The typical cursor is 32x32x2 in size. See also *raster* on page 2-194.

frame burst A group of transmitter receive frames that are contiguous on the media.

frame level In X.25 communications, the level between the physical level and the packet level, which works according to the high–level data link control procedure (HDLC). Synonymous with *data–link level* on page 2-61 and *level 2*. See also *level* on page 2-127, *packet level* on page 2-167, and *physical level* on page 2-175.

frame–level interface
In X.25 communications, the level of the DTE/DCE interface in packet mode operation relating to the exchange of packets with local error control, where packets are contained in frames. See also *packet–level interface* on page 2-167.

frame window In X.25 communications, the number of frames that can be outstanding without acknowledgment. See also *packet window* on page 2-167 and *window* on page 2-263.

framing error An asynchronous transmission error usually caused by the number of bits per character not being set the same on the sending and receiving workstations.

free list A list of available blocks on each file system.

free–block list See *free list* on page 2-96.

free space Space reserved within the control intervals of a key–sequenced data set or file, used for inserting new records into the data set or file in key sequence; also, whole control intervals reserved in a control area for the same purpose. Synonym for *distributed free space*.

frequency modulation (FM)
Modifying the frequency of a fixed amplitude carrier signal in accordance with an informational signal so that it can carry data signals.

friend class A C++ class in which all the member functions are granted access to the private and protected members of another class. It is named in the declaration of another class and uses the keyword **friend** as a prefix to the class. For example, `class me {` on one line followed by `friend class you;` on the next and then `// ...` and ending with `};` makes all the functions in class `you` friends of class `me`.

friend function A C++ function that is granted access to the private and protected parts of a class. It is named in the declaration of the class and uses the keyword **friend** as a prefix.

front and back buffers
In GL, in double buffer mode, the main frame buffer bit planes are separated into two sets—the front and back buffers. Bits in the front buffer planes are visible and those in the back buffer are not. Typically, an application draws into the back buffer and views the front buffer for dynamic graphics.

Front Panel In CDE, a centrally located window containing controls for accessing applications and utilities, including the *workspace switch*. The Front Panel occupies all workspaces.

Front Panel control
In CDE, an object in the Front Panel used as an interface to basic system services and frequently performed tasks and operations. Controls in the default Front Panel are: Clock, Calendar, Mailer, Text Editor, Help Viewer, File Manager, Printer, Style Manager, Application Manager, and Trash Can. See also *control* on page 2-52.

Front Panel move handles
In CDE, an area on each end of the Front Panel used for moving the Front Panel and for bringing it to the front of the workspace.

frustum In GL, a truncated, four–sided pyramid; that is, a pyramid with the point cut off. In a perspective projection, the shape of the clipping volume is a frustum. The bottom of the frustum is referred to the far clipping plane, the top of the frustum is the near clipping plane, and the sides are respectively the top, left, bottom, and right clipping planes. In an orthographic projection, the clipping volume is a parallelepiped. See also *clipping panes* on page 2-39.

FTP File Transfer Program. See also *File Transfer Protocol* on page 2-91.

full backup Backup copies of all the files on the system. Contrast with *incremental backup* on page 2-112.

full duplex (FDX)
Synonym for *duplex* on page 2-76.

full high Pertains to a storage device that has the original height (82.5 mm or 3.25 inches), width (146 mm or 5.75 inches), and length (203 mm or 8.0 inches). See also *half high* on page 2-105.

full path name The name of any directory or file expressed as a string of directories and files beginning with the root directory. See also *fully qualified name* on page 2-97, *path name* on page 2-172, and relative path name on page 2-199.

full–screen editor
An editor that displays an entire screen at a time and that allows data to be accessed and modified only by moving the cursor and entering commands.

fullword Synonym for *word* on page 2-263.

fully qualified host name
A complete qualified name. Includes all names in the hierarchical sequence above the structure member to which the name refers, as well as the member itself.

fully qualified name
A qualified name that includes all names in the hierarchical sequence above the structure member to which the name refers, as well as the name of the member itself. See also *full path name* on page 2-97.

function (1) A specific purpose of an entity, or its characteristic action.
(2) A machine action such as carriage return or line feed.
(3) A subroutine that returns the value of a single variable and that usually has a single exit, such as subroutines that compute mathematical functions. Synonymous with *procedure* on page 2-185. For Ada programming, see also *subprogram* on page 2-230.

function call An expression that moves the path of execution from the current function to a specified function and evaluates to the return value provided by the called function. A function call contains the name of the function to which control moves and a parenthesized list of values.

function declarator
The part of a function definition that names the function, provides additional information about the return value of the function, and lists the function parameters.

function definition
The complete description of a function. A function definition contains an optional storage class specifier, an optional type specifier, a function declarator, optional parameter declarations, and a block statement (the function body).

function keys Keys that request actions but do not display or print characters. This includes the keys that normally produce a printed character, but produce a function instead when used with the code key. See also *character key* on page 2-34.

function pointer
An identifier that gives the location of a function or procedure.

function reference
The appearance of an intrinsic function name or a user function name in an expression.

function scope C++ labels that are declared in a function have function scope and can be used anywhere in that function.

function subprogram
See *function* on page 2-97.

function template
Provides a blueprint describing how a set of related individual C++ functions can be constructed.

functional equivalence
A situation in which different algorithms for a given problem domain yield identical results.

funnelling Forcing device drivers and kernel extensions that are not known to be MP safe to run only on the master processor.

FX See *FXU* on page 2-98.

FXU (FX) Fixed–point unit.

G

gadget In AIXwindows, a windowless graphical object that looks like its equivalent like–named widget but does not support the translations, actions, or **Popup** widget children supplied by that widget.

gadget ID In AIXwindows, a unique identification number assigned to each instance of a gadget used within a given graphical interface.

gamma correction
In GL, a logarithmic assignment of intensities to lookup table entries for shading applications. This is required since the human eye perceives intensities logarithmically rather than linearly. See also *gamma ramp* on page 2-99.

gamma ramp In GL, a set of three lookup tables, one for each of the colors red, green, and blue, attached to the electron guns of the monitor. Entries in the gamma lookup table can be set to adjust for variations in the phosphor quality between different brands of monitors. Usually, a logarithmic curve is loaded into the gamma lookup tables. See also *gamma correction* on page 2-99. The gamma lookup tables are not a subset of the color map tables, but a separate entity.

gateway (1) An entity that operates above the link layer and translates, when required, the interface and protocol used by one network into those used by another distinct network.
(2) The network that connects hosts. See also *active gateway* on page 2-5.
(3) A device and its associated software that interconnect networks or systems of different architectures. The connection is usually made above the reference model network layer. For example, a gateway allows LANs access to System/390 host computers. Contrast with *bridge* on page 2-25.

gateway host A host that connects independent networks. It has multiple interfaces, each with a different name and address.

Gateway–to–Gateway Protocol (GGP)
The protocol with which a gateway determines connectivity to networks and neighbor gateways and implements the shortest–path routing algorithm.

gather For input/output operations, to read data from noncontiguous memory locations to write to a device. Contrast with *scatter* on page 2-210.

GB See *gigabyte* on page 2-101.

GC See *graphics context* on page 2-102.

GC caching In Enhanced X–Windows, allows independent change requests to be merged into one protocol request.

GContext See *graphics context* on page 2-102.

GCP Graphics Control Processor.

GCR Group Code Recording, a magnetic tape recording format with a density of 6250 bpi.

GDA Global Directory Agent.

GDLC See *generic data link control* on page 2-100.

general purpose register (GPR)
An explicitly addressable register that can be used for a variety of purposes (for example, as an accumulator or an index register). See also *register* on page 2-198.

generation (1) A means of referencing items in terms of time and ancestry so that an item without antecedents is designated as the first (n–th) generation and subsequent derivations are designated as n–1, n–2, and so on.
(2) For some remote systems, the translation of configuration into machine language.

generation number
A number used to differentiate between the uses of the same inode for several files or for different versions of the same file. When a file is created, an inode is allocated for the file, and a new generation number is stored in the inode for that file. Any use of an inode with the wrong generation number indicates that the user's view of the file is out–of–date, or "stale."

generation phase
The part of the sort program that translates the sequence specifications into machine language.

generic data link control (GDCL)
A generic interface definition that provides application and kernel users a common set of commands to control DLC device managers within the operating system. Requirements for entry–point definitions, functions provided, and data structures for all DLC managers are specified. The following DLCs conform to this interface: IEEE 802.3 Ethernet, standard Ethernet, SDLC, and Token Ring.

generic interface
The interface, defined at a level that is independent of any particular programming language.

generic network
A Network Installation Management (NIM) network type used to define networks which are not currently known by NIM to support network boot operations.

generic unit In Ada language, a template either for a set of subprograms or for a set of packages. A subprogram or package created using the template is called an instance of the generic unit. A generic instantiation is the kind of declaration that creates an instance. A generic unit is written as a subprogram or package but with the specification prefixed by a generic formal part which may declare generic formal parameters. A generic formal parameter is either a type, a subprogram, or an object. A generic unit is one of the kinds of program unit. See also *instance.* on page 2-116.

geometric text Text whose character fonts are defined by mathematical descriptions of the strokes required to draw the characters, rather than by raster images. Also called programmable character set on page 2-188 or *stroke text* on page 2-228.

geometry In Enhanced X–Windows, (1.) Window size and position on the screen. (2.) The size of a widget is changed using geometry management routines. Synonym for *layout* on page 2-126 and *geometry management*.

geometry management
Synonymous with *geometry* on page 2-100.

GGP See *Gateway–to–Gateway Protocol* on page 2-99.

ghost icon An icon that indicates that a file is missing.

ghost instrument
In the Performance Toolbox, an empty space in the console where an instrument used to be, usually caused when a console designed for one system contains instruments not available on the current system. Ghost instruments occupy a space and prevent a new instrument from being defined in that same space or other instruments from being moved or resized to use the space.

ghosted A description of an unavailable choice. Menu items are *ghosted* or ”grayed out” to indicate that a standard choice is not available under the current circumstances.

GID See *group number* on page 2-103.

gigabyte (GB) 1 073 741 824 in decimal notation when referring to memory capacity; in all other cases, it is defined as 1 000 000 000.

GL Graphics Library.

GLB See *Global Location Broker* on page 2-101.

glbd The Global Location Broker daemon.

global (1) In programming languages, pertaining to the relationship between a language object and a block in which the language object has a scope extending beyond that block but contained within an encompassing block.
(2) Pertaining to information defined in one subdivision of a computer program and used in at least one other subdivision of the program.
(3) Pertaining to information available to more than one program or subroutine.

global area (1) A storage area used for communication between two or more main programs.
(2) An uninitialized portion of a partition accessible by any program of a task set in the partition at a given time. The same area may be used by other task sets that run in the same partition.

global character
The * and ? special characters that can be used in a file specification to match one or more characters. For example, placing ? in a file specification means any character can be in that position.

global data Data that can be addressed by any process while in kernel mode, for example, tables, such as the open file table and process table, and other data maintained by the kernel, such as buffer pointers.

global forward substitution
An optimization in which the result of an assignment can be propagated forward through a program. For example, the assignment B=C followed by A=B can be replaced by A=C if B is not used elsewhere in the program.

Global Location Broker (GLB)
Part of the NCS Location Broker. A server that maintains global information about objects on a network or an internet.

global variable A symbol defined in one program module but used in other independently assembled program modules.

globbing Synonym for *file name substitution* on page 2-90.

glossary window
A window that contains a glossary term.

glyph (1) An image, usually of a character, in a font.
(2) A graphic symbol whose appearance conveys information; for example, the vertical and horizontal arrows on cursor keys that indicate the directions in which they control cursor movement.

good citizenship
A group of standards established by the *Inter–Client Communications Conventions Manual* (ICCCM) that are supported and implemented by AIXwindows Window Manager. These standards prescribe behavior among clients in a multiclient environment to avoid costly compatibility problems.

goto statement In programming languages, a statement that transfers control to another point in a program.

Gouraud shading
A method of shading polygons smoothly based on the intensities at their vertices. The color is uniformly interpolated along each edge, and then the edge values are uniformly interpolated along each scan line. For realistic shading, colors should be gamma corrected.

GPR See *general purpose register* on page 2-99.

grab (1) The act of selecting keyboard keys, the keyboard, pointer buttons, the pointer, and the server for exclusive use by a client. In general, these facilities are not intended to be used by normal applications, but are intended for various input and window managers to implement various styles of user interfaces. See also *active grab* on page 2-5 *, passive grab* on page 2-171 *, button grabbing* on page 2-17 *, pointer grabbing* on page 2-177, and *key grabbing* on page 2-124.
(2) A procedure by which a window acts upon a key or button event that occurs for it or any of its descendents. This precludes the necessity of setting up translations for all windows.

grab handles (or handles)
In CDE, the small squares displayed at the corners and midpoints of a selected graphic element.

grammar rules The structure rules in a parser program. See also *parser* on page 2-170.

granularity The extent to which a larger entity is subdivided. For example, a yard broken into inches has finer granularity than a yard broken into feet.

graphic character
A character that can be displayed or printed.

graphical user interface
(1) A boundary between two functional units that is displayed to the user by means of fundamental drawing units such as lines and polygons.
(2) A type of computer interface consisting of a visual metaphor of a real–world scene, often a desktop. Within that scene are icons, representing actual objects, that the user can access and manipulate with a pointing device. (Also called *graphical interface*.)

graphics A type of data created from such fundamental drawing units such as lines, curves, polygons, and so forth.

graphics context (GC)
In Enhanced X–Windows, the storage area for various kinds of graphics output, such as foreground pixel, background pixel, line width, and clipping region. Also known as ”GC” and ”GContext,” a graphics context can be used only with drawables that have the same root and the same depth as the graphics context.

graphics pipeline
In GL, the sequence of steps that a graphics primitive goes through before it becomes visible on the screen: transformation from model coordinates to NDC coordinates; 3–D clipping (if out of bounds); perspective division; determination of color through lighting equations or depth–cueing; transformation of NDC coordinates to screen coordinates; 2–D clipping (by the screenmask); rasterization (drawing into the frame buffer); and display of frame buffer.

Graphics Support Library (XGSL)
A two–dimensional graphics application programming interface to various output devices.

graPHIGS API An implementation of PHIGS, based on the American National Standards Institute (ANSI) proposed standard, Programmer's Hierarchical Interactive Graphics System (PHIGS).

gravity In Enhanced X–Windows, the contents of windows or subwindows have an attraction to a location within the window. This determines how the window ID is resized. See also *bit gravity* on page 2-19 and *window gravity* on page 2-263.

gray scale (1) In Enhanced X–Windows, a type of degenerate pseudocolor where the red, green, and blue values in any given color map entry are equal, thus producing shades of gray. The gray values can be changed dynamically.
(2) Also *GrayScale*, a value.
(3) In a grayscale adapter, the different levels of intensity corresponding to the shades of gray produced. See also *monochrome display* on page 2-147.

grayed A characteristic of a command or option that is listed on a menu or list box but cannot be chosen.

gross clipping In GL, also known as 3–D clipping this is the clipping that occurs in normalized device coordinates, against the sides of the perspective frustum. All 3–D primitives undergo this clipping; in particular, the origin of text strings (but not individual letters) are clipped in this way. See also *clipping* on page 2-38, *fine clipping* on page 2-91, and *clipping planes* on page 2-39.

group (1) A collection of users who can share access authorities for protected resources.
(2) A list of names that are known together by a single name.
(3) A series of records logically joined together.
(4) A series of lines repeated consecutively as a set on a full–screen form or full–screen panel.
(5) A set of related records that have the same value for a particular field in all records.

Group Appointment
In CDE, in Calendar, an appointment for a group of people, scheduled using the Group Appointment Editor.

group ID (GID) A number that corresponds to a specific group name. The group ID can often be substituted in commands that take a group name as a value. Synonymous with *group number*.

group name A name, which uniquely identifies a group of users to the system, that contains one to eight alphanumeric characters, beginning with an alphabetic, #, $, or > character.

group number Synonym for *group ID* on page 2-103.

guard expressions

Expressions placed at the beginning of Boolean expressions to check that other operations can be done.

H

half duplex (HD or HDX)
Pertains to communications in which data can be sent in only one direction at a time. Contrast with *duplex* on page 2-76.

half high Pertains to a storage device that has one half the height (41.3 mm or 1.63 inches) with the width and length the same as the full high device. See also *full high* on page 2-97.

half–session A component that provides data flow control and transmission control at one end of a session.

halfword A contiguous sequence of bits or characters that constitutes half a computer word and can be addressed as a unit. Contrast with *word* on page 2-263.

hand tuning The process of modifying source code by hand to improve the performance of the resulting object code. Hand tuning is one aspect of optimization.

handle A data structure that is a temporary local identifier for an object. You create a handle by allocating it. You make a handle identify an object at a specific location by binding it.

handler A software routine that controls a program's reaction to specific external events, such as an interrupt handler. For Ada programming, see *exception* on page 2-84.

handshaking The process used by modems and equipment, before transmitting data, to establish an electrical path and synchronization. See also *chat script* on page 2-35 and *expect–send sequence* on page 2-85.

hard copy A printed copy of machine output in a visually readable form such as printed reports, listings, documents, and summaries.

hard disk See *fixed disk* on page 2-92.

hard–disk drive
See *fixed–disk drive* on page 2-92.

hard link A mechanism that allows the **ln** command to assign more than one name to a file. Both the new name and the file being linked must be in the same file system.

hard return In an Interleaf textfile, a control character created by pressing Shift–Enter that causes text to wrap to the next line.

hard space In an Interleaf file, a nonbreaking space character.

hardware The physical equipment of computing and computer–directed activities. The physical components of a computer system. Contrast with *software* on page 2-221.

hash codes Format in which data is stored in compressed form.

hash signature The fixed–length bit string resulting from hashing a character string. Character strings may be compared quickly by comparing their hash signatures.

hash table A table of information that is accessed by way of a shortened search key (the hash value). Using a hash table minimizes average search time.

hashing (1) A method of transforming a search key into an address for the purpose of storing and retrieving items of data.
(2) Encoding a character string as a fixed–length bit string for comparison. The encoding may not necessarily be unique.

HCON See 3270 Host Connection Program 2.1 and 1.3.3. on page 1-1

HCONMRI 3270 Host Connection Program 2.1 and 1.3.3 Message Catalog.

HCONuser A user who has been given the special permissions necessary to use the 3270 Host Connection Program 2.1 and 1.3.3 (HCON). See also 3270 Host Connection Program 2.1 and 1.3.3 on page 1-1.

HD See *half duplex* on page 2-105.

HDLC See *high–level data link control* on page 2-107.

HDR See *header label* on page 2-106.

HDX See *half duplex* on page 2-105.

header (1) Constant text that is formatted to be in the top margin of printed pages in a document.
(2) System–defined control information that precedes user data.
(3) The portion of a message that contains control information for the message such as destination fields, originating station, and priority level.
(4) In CDE, in Mailer, the basic information about an electronic mail message as it appears in the mailer container. A message header displays the name of the sender, subject, the date and time it was received, and message size.

header file A text file that contains declarations used by a group of functions or users. Synonymous with *include file* on page 2-112.

header label (HDR) A special set of records on a diskette describing the contents of the diskette.

header page A separator page that precedes a printed file or a print job.

header record A record at the beginning of a file that details the sizes, locations, and other information that follows in the file.

heap A collection of dynamically allocated variables.

HELLO DCN Local–Network Protocol. An interior gateway protocol designed for use within autonomous systems.

'Hello, World' The standalone sample program that introduces the fundamental construct provided by Tran–C.

help One or more display images that describe how to use application software or how to do a system operation.

help callback A function that calls a help.

help file A file, separate from the source code of a program, that contains help definitions in a special help format that Base Operating System message services can use.

Help Manager In CDE, a special help volume that lists all the online help registered on your system.

Help Viewer In CDE, the software application that displays online help.

help volume In CDE, a complete body of help information about a subject.

hertz (Hz) A unit of frequency equal to one cycle per second.

heuristics Guidelines that a system administrator uses to intervene where the two–phase commit or abort would otherwise fail.

hex See *hexadecimal* on page 2-107.

hexadecimal (hex)
Pertaining to a system of numbers to the base sixteen; hexadecimal digits range from 0 (zero) through 9 (nine) and A (ten) through F (fifteen).

hexadecimal constant
(1) The characters 0x or 0X (zero X) followed by any digits 0 through 9 and uppercase or lowercase letters A through F.
(2) A constant, usually starting with special characters, that contains only hexadecimal digits.

HIA See System/370 *Host Interface Adapter* on page 2-234.

hidden file An operating system file that is not displayed by a directory listing. The name of a hidden file usually begins with a period.

hidden surface A surface of a geometric primitive that is not visible because it is obscured by other surfaces. See also *z–buffer* on page 2-269.

hierarchical file system
In CDE, a way to organize data on computer systems using a hierarchy of containers, often called folders (directories) and files. In this scheme, folders may contain other folders and files. The successive containment of folders within folders creates the levels of organization, which is the hierarchy.

hierarchical network
A network in which hosts are administered by a tree structure of authority. This network structure relieves the administrative burden of the central authority.

hierarchy A series of elements that have been graded or ranked in some useful manner. In AIXwindows, more than 40 classes of graphical objects are ranked top–down from the simplest to the most complex to determine the relative order of inheritance of appearance resources and behavior resources.

high–level data link control (HDLC)
In data communications, the use of a specified series of bits to control data links in accordance with the International Standards for HDLC: ISO 3309 Frame Structure and ISO 4335 Elements of Procedures.

high–order Most significant; leftmost. For example, bit 0 in a register.

higher layer or level
The conceptual level of control or processing logic existing in the hierarchical structure of a station that is above the link layer and upon which the performance of data link functions are dependent (for example, device control, buffer allocation, station management). See also *level* on page 2-127.

highlight To emphasize an area on the display screen by any of several methods, such as brightening the area or reversing the color of characters within the area.

hiragana A graphic character set consisting of symbols used in one of the two common Japanese phonetic alphabets. Each character is represented by 1 byte.

history A list of Ada debugger commands entered recently. The history command has a default list of the 20 most recently entered commands. You can use history to re–enter a previous command, or to form a new command by editing the history reference text.

hit In GL, also called *pick hit* or *select hit*. A hit occurs whenever a drawing primitive draws within the picking or selecting region. A hit is reported back to the user only if the name stack has changed since the last hit. In other words, multiple hits may occur although only one pick/select event is reported. See also *picking* on page 2-175, *picking region* on page 2-175, *selecting region* on page 2-213, and *selecting* on page 2-213.

H&J In C.A.T terminology, the (usually automatic) process of hyphenation and justification.

hog factor In system accounting, an analysis of how many times each command was run, how much processor time and memory it used, and how intensive that use was.

hole in a file Empty space in a sparse file that is left open for future additions of data. See also *sparse file* on page 2-222.

Hollerith constant
In FORTRAN, a string of any characters capable of representation in the processor and preceded by *n* H, where *n* is the number of characters in the string.

HOME See *$HOME* on page 1-1.

home cell See *local cell* on page 2-132.

home directory
(1) A directory associated with an individual user.
(2) The user's current directory after login or after issuing the **cd** command with no argument.
(3) A parameter that supplies the full path name of the home directory for the transaction program.

home folder In CDE, a folder where you keep personal files and additional directories. By default, File Manager and Terminal Emulator windows are set to the home folder when you first open them. When discussing command–line activities, may be called the *home directory*.

home session In CDE, a choice at logout to designate a particular session, other than the one you are currently in, as the one you will automatically return to at the next login. Contrast with *current session* on page 2-58.

home–window A window that completely displays the contents of the display buffer.

homogeneous coordinates
A four–dimensional method of representing three–dimensional space. A point (*x*, *y*, *z*, *w*) in homogeneous coordinates is used to represent a point (*X*, *Y*, *Z*) in three–dimensional space by taking $X = x / w$, $Y = y / w$, and $Z = z / w$. See also *parametric bicubic surface* on page 2-169.

hook ID A unique number assigned to a specific trace point. All trace entries include the hook identifier of the originating trace point in the trace entry header. A hook ID is a 12–bit value. For user programs, the hook ID may be a value from 0x010 to 0x0FF. Hook identifiers are defined in the **/usr/include/sys/trchkid.h** file.

hooking routines
Functions that connect with the library but remain outside the library; protocol extension procedures. Synonym for *stub* on page 2-229.

hop count In the Token–Ring Network, the number of bridges through which a frame passes on the way to its destination.

hop count metric
(1) In a gateway, indicates that the next string represents the hop count to the destination host or network.
(2) The number of host–to–host connections in a route.

host (1) The primary or controlling computer in the communications network.
(2) A computer attached to a network.

host ID An identifier for a host. A host ID uniquely identifies a host within an address family on a network but does not identify the network. A host ID is not necessarily sufficient to establish communications with a host.

host name (1) A name assigned to a computer connected to a network. The use of this term can be ambiguous as it can refer to either the short form name of the computer, or the fully qualified name of the computer.
(2) The Internet address of a machine in the network. Also known as *host ID*.

hotspot The spot associated with a cursor that corresponds to the coordinates reported for the pointer. A cursor has an associated hotspot, which defines a point in the cursor that corresponds to the coordinates reported for the pointer.

HTML HyperText Markup Language (HTML) is the tagging language that a web browser uses to interpret and display documents.

Huffman coding
A character–coding technique to compress data.

hyper–integer An XDR standard that defines 64–bit (8–byte) numbers that are the extensions of integers and unsigned integers.

hyperlink In CDE, in Help text, text or a graphic you click to display another Help topic.

hypertext A way of presenting information online with connections between one piece of information and another. These connections are called hypertext links. Thousands of these hypertext links enable you to explore additional or related information throughout the online documentation. See also *hypertext link* on page 2-109.

hypertext link A connection between one piece of online information and another. See also *hypertext* on page 2-109.

Hz See *hertz* on page 2-106.

I

IAR See *instruction address register* on page 2-117.

IC See *ICU* on page 2-110.

ICCC Inter–Client Communications Conventions. A standard abbreviation for the manual that contains a detailed set of guidelines for client applications using the Enhanced X-Windows system and the AIXwindows toolkit. See also *ICCCM* on page 2-110.

ICCCM Inter–Client Communications Conventions manual. See also *ICCC* on page 2-110.

ICMP See *Internet Control Message Protocol* on page 2-120.

icon (1) A picture or graphical representation of an object on a display screen to which a user can point to with a device such as a mouse in order to select a particular operation or perform a certain action.
(2) In Common Desktop Environment, the visual representation of a file or directory, or an object used by the desktop, consisting of a picture on the desktop or in a directory window.

icon box In AIXwindows, a window used as a visual storage area for icons representing minimized windows.

Icon Editor In CDE, the software application used to create new icons and to modify existing icons.

icon layout policy In AIXwindows, a specification that determines whether icons representing minimized windows are placed on the root window or within an icon box.

iconify See *icon* on page 2-110.

ICU (IC) Instruction cache unit.

ID Identification. See *identifier* on page 2-110.

idempotent A class of operations. An operation is idempotent if its results do not affect the results of any operation. For example, a call that returns the time is idempotent.

identifier (ID) (1) A name that refers to a data object. An identifier contains some combination of letters, digits, and underscores, but its first character cannot be a digit.
(2) In programming languages, a lexical unit that names a language object, such as the name of an array, record, label, or procedure. An identifier usually begins with a letter optionally followed by letters, digits, or other characters.
(3) A sequence of bits or characters that identifies a program, device, or system to another program, device, or system.
(4) In Enhanced X–Windows, a unique value associated with a resource that a client program uses to name the resource. An identifier can be used over any connection to name the resource.

idle list A list of secondary stations on a network that are polled less often by the primary station due to their inactivity.

idle time The part of operable time during which a functional unit is not operated.

IDP Internet Datagram Protocol. A simple, unreliable datagram protocol, which is used to support the SOCK_DGRAM abstraction for the Internet Protocol family.

IEEE Institute of Electrical and Electronics Engineers.

IEEE 754 Binary Floating Point Standard.

IEEE 802.3 Ethernet LAN specification.

IEEE 802.5 Token–Ring LAN specification.

IETF Internet Engineering Task Force.

IF expression An expression in a procedure that tests for a condition. The action performed by the procedure depends on the result of the test.

if statement (1) A C language conditional statement that contains the keyword **if** followed by an expression in parentheses (the condition), a statement (the action), and an optional **else** clause (the alternative action).
(2) A conditional statement that specifies a condition to be tested and the action to be taken if the condition is satisfied.

I–field Information field.

I–field bytes Data within the information field of a transmit or receive sequenced data frame.

ignore In DOS, the option to disregard the device error and continue processing.

i–list In a Base Operating System file system, blocks 2 through *n* compose the i–list, which contains structures (i–nodes) that relate a file to the data blocks or disk. The size of the i–list depends on the size of the mounted file system. See also *i–node* on page 2-114 and *superblock* on page 2-218.

illegal A violation of an architecture rule that an implementation is required to report. See also *unpredictable* on page 2-252.

image cache In AIXwindows, a means of associating an image with a name. Once this association is in place, the appropriate AIXwindows subroutines can generate pixmaps through references to an**.Xdefaults** file (by name) and through an argument list (by pixmap). See also *pixmap* on page 2-176.

image structure
The existing image in an **XmInstallImage** function.

immediate data
(1) In Assembler language, actual data appearing in an instruction, as opposed to the symbolic name of some data. The data is immediately available from the instruction and therefore does not have to be read from memory.
(2) Data transferred during instruction run time.

immediate mode
In GL, in this mode, graphics commands are executed immediately rather than being compiled into a display list.

immediate subclass
A subclass, of a class C, having no super classes that are themselves subclasses of C.

immediate subobject
One object that is a value of an attribute of another.

immediate superclass
The superclass, of a class C, having no subclasses that are themselves superclasses of C.

immediate superobject
One object that contains another among its attribute values.

IMPL Initial microprogram load.

implicit type conversion
A type conversion generated by the compiler as the result of an association between variables of different types. For example, initializing a floating–point array to the value of a loop counter is an implicit type conversion from integer to floating–point.

implied DO In FORTRAN, an indexing specification with a list of data elements, rather than a set of statements, as its range. Similar to an FORTRAN DO statement, however, without using the word DO.

import (1) In NCS, to request the operations defined by an interface. A client imports an interface from a server.
(2) To bring data in from another system. Contrast with *export* on page 2-85.

imports In Ada language, all compilation units named in the context specification of a library unit (stated in its **with** clauses) and any supporting unit specifications introduced by the compiler, for example, the standard package SYSTEM. The imports of a secondary unit are all the units named in its context specification, any supporting unit requirements introduced by the compiler; its associated parent unit (if any), and the bodies of any generic specifications included among its other imports.

in See *inch* on page 2-112.

inaccessible In XOM, said of an object for which the client does not possess a valid designator or handle.

inactive Describes a window that does not have the *input focus* on page 2-115.

Inbox In CDE, the main or default Mailer container. The Inbox receives all incoming mail. Messages can be moved out of the Inbox and put in various containers.

inch (in) 2.54 centimeters or 25.4 millimeters.

include file A text file that contains declarations used by a group of functions, programs, or users. Synonymous with *header file* on page 2-106. See also *include statement* on page 2-112.

include statement
A computer language preprocessor statement that directs the processor to retrieve a specific file that contains instructions and data the program may need. See also *include file* on page 2-112.

incoming call In X.25 communications, a call arriving at the data terminal equipment (DTE).

incoming–call packet
In X.25 communications, a call supervision packet transmitted by a DCE to inform a DTE of a call requested by another DTE. See also *packet* on page 2-166.

incomplete class declaration
A C++ class declaration that does not define any members of a class. Until a class is fully declared, or defined, you can only use the class name where the size of the class is not required. Typically, an incomplete class declaration is used as a forward declaration.

incremental backup
The process of copying files that have been opened for reasons other than read–only access since the last backup was created and that meet the backup frequency criteria. Contrast with *full backup* on page 2-97.

index (1) A table containing the key value and location of each record in an indexed file.
(2) A computer storage position or register whose contents identify a particular element in a set of elements.
(3) A list of the contents of a file or a document, together with keys or references for locating the contents. See also *base register* on page 2-18.

index constraint
In Ada language, an index constraint for an array type specifies the lower and upper bounds for each index range of the array type.

index priority Priority of an attribute type in search queries.

indexed application widget
Any widget named in the User Interface Language (UIL) that is not the child of any other widget in the User Interface Definition (UID) hierarchy.

indexed component
In Ada language, an indexed component denotes a component in an array. It is a form of name containing expressions which specify the values of the indices of the array component. An indexed component may also denote an entry in a family of entries.

indexed fields An area in a structured data file that contains tree data paths.

indexed instruction
An instruction that uses an indexed address.

indicator (1) An internal switch that communicates a condition between parts of a program or procedure.
(2) A device that can be set to a prescribed state, usually according to the result of a previous process or on the occurrence of a specified condition in the equipment. The device usually indicates the existence of the state and may be used to determine the selection among alternative processes.

indirect block A block containing pointers to other blocks.

industry–standard benchmark
A benchmark that has been adopted by consensus or by some (presumably neutral) sponsoring organization as constituting a meaningful measure of some aspect of computer–system performance. There are many counter–examples to the assumption that an improvement in industry–standard benchmark performance corresponds to an improvement in the performance experienced by users.

inferiors In Enhanced X–Windows, all the subwindows nested below a window.

infinity (1) A name for the upper boundary of the set of numbers.
(2) In binary floating–point concepts, a value with an associated sign that is mathematically greater in magnitude than any binary floating point number.

information field overflow (I–field overflow)
Condition that occurs when the size of the information field in the receive data exceeds the primary station's buffer capacity. Some of the information field is lost.

informational message
(1) A message that provides information to the operator but does not require a response.
(2) A message that is not the result of an error condition.

inherit To copy resources or attributes from a parent to a child.

inheritance (1) In AIXwindows and Enhanced X–Windows, the passing of class resources from an object superclass downstream in the class hierarchy to an object subclass.
(2) An object–oriented programming technique that allows you to use existing classes as bases for creating other classes.

initial program load (IPL)
(1) The initialization procedure that causes an operating system to commence operation. Synonymous with *system restart* and *system startup*.
(2) The process by which a configuration image is loaded into storage at the beginning or a work day or after a system malfunction.
(3) The process of loading system programs and preparing a system to run jobs.

initial program load device (IPL device)
The input/output device, usually a fixed disk or diskette, from which the system software is loaded when the machine is turned on.

initial sequence controller
One of three control programs for the initial program load (IPL) ROM. The initial sequence controller accepts control after the hardware initialization and passes control to the core sequence controller.

initialize (1) In programming languages, to set the starting value of a data object.
(2) To set counters, switches, addresses, or contents of storage to zero or other starting values at the beginning of, or at prescribed points in, the operation of a computer routine.
(3) To prepare for use, such as initializing a diskette.
(4) To prepare the system for operation. After loading the kernel into memory, the system runs internal checks, initializes all memory and some devices, and analyzes the root file system.

initializer The assignment operator followed by an expression or multiple expressions for aggregate variables.

inline See *inline component* on page 2-114.

inline component
A component within a component that has its own properties and can contain anything a component contains. It is created from the same set of masters as a component.

inline expansion
An optimization in which the reference to a procedure is replaced with the code of the procedure itself, to eliminate calling overhead.

inline function In C++, inlining is a hint to the compiler to perform inline expansion of the body of a function member. Functions declared and defined simultaneously in a class definition are inline. You can also explicitly declare a function inline by using the keyword **inline**. Both member and nonmember functions can be inlined.

inlining See *inline expansion* on page 2-114.

i–node The internal structure that describes the individual files in the operating system; there is one i–node for each file. An i–node contains the node, type, owner, and location of a file. A table of i–nodes is stored near the beginning of a file system. Synonym for *file index* on page 2-90. See also *i–list* on page 2-111 and *i–node number* on page 2-114.

i–node number A number specifying a particular i–node file in the file system. See also *i–node* on page 2-114.

inodetab A kernel parameter that establishes a table in memory for storing copies of i–nodes for all active files.

input (1) Data to be processed.
(2) In Pascal, a predefined standard file definition.

input data type The type of data contained in the input file. Examples are ASCII and PostScript.

input device (1) A physical device that provides data to a computer.
(2) The device that is the source of the software you are installing. The input device can be a tape drive, CD–ROM drive, DVD drive, diskette drive, or a directory.

input field (1) An area in a display file into which you can type data.
(2) In computer graphics, an unprotected field on a display surface in which data can be entered, modified, or erased.

input file A file opened in order to allow records to be read.

input focus In a graphics environment, a window defining the scope for processing keyboard input. By default, keyboard events are sent to the client using the window the pointer is in. It is also possible to attach the keyboard input to a specific window. Events are then sent to the appropriate client regardless of the pointer position. Synonymous with *focus window*. See also *inactive* on page 2-112.

input list A list of variables to which values are assigned from input data.

input manager In a graphics environment, a client that controls keyboard input and is usually part of a window manager.

input mode An open mode in which records can be read from the file.

Input Only window
In a graphics environment, an invisible window that can be used to control such things as cursors, input event generation, and grabbing. This window cannot be used for graphics requests.

input redirection
The specification of an input source other than the standard one.

input semantics
The specified order and format in which user input must be entered.

input stream The sequence of operation control statements and data given to the system from an input device.

input/output (I/O)
(1) Pertaining to either input, output, or both between a computer and a device.
(2) Pertaining to a device whose parts can perform an input process and an output process at the same time.
(3) Pertaining to a functional unit or channel involved in an input process, an output process, or both, concurrently or not, and to the data involved in such a process.

input/output channel (IO channel)
(1) In a data processing system, a functional unit that handles transfer of data between internal and peripheral equipment.
(2) In a computing system, a functional unit, controlled by a processor, that handles transfer of data between processor storage and local peripheral devices.

input/output channel controller (IOCC)
A hardware component that supervises communication between the input/output bus and the processor.

input/output configuration data set (IOCDS)
A configuration definition built by the I/O configuration program (IOCP) and stored on disk files associated with the processor controller.

input/output configuration program (IOCP)
The program that defines the I/O configuration data required by the processor complex to control I/O requests.

input/output device number
A value assigned to a device driver by the guest operating system or to the virtual device by the virtual resource manager. This number uniquely identifies the device regardless of whether it is real or virtual.

input/output file
A file opened for input and output use.

input/output subsystem
That part of the operating system comprised of processes and device managers that provides the mechanisms for data transfer and I/O device management and control.

InputOutput window
In a graphics environment, a kind of opaque window used for input and output. InputOutput windows can have both InputOutput and InputOnly windows as inferiors.

inquiry (1) A request for information in storage.
(2) A request that puts a display station into inquiry mode.
(3) In data communications, a request for information from another system.

insert cursor The position of the cursor marking where new characters will be added when entering text.

insert mode (1) A keyboard operation that puts new text within existing text at the cursor position.
(2) The source entry utility operation during which source statements are typed in and added as new records in a source member.
(3) In the Token–Ring Network, to make an attaching device an active part of a ring.

insertion cursor position
The point at which text will be inserted.

insertion point In Common Desktop Environment, the point at which text is inserted when you type. It usually appears as a flashing vertical line or underline.

install (1) To add a program, program option, or software program to the system in a manner such that it is runnable and interacts properly with all affected programs in the system.
(2) To connect a piece of hardware to the processor.
(3) The process of connecting used when discussing the process of connecting something to the internal portion of the processor.

Install Icon In CDE, an item on a subpanel used to install icons into the Front Panel using drag and drop.

installation (1) The procedure of adding a program or program option to the mass storage medium of the computer, making the program runnable, and ensuring that the program interacts properly with all other affected programs in the system.
(2) The task of adding a device driver to the system and activating the driver so that it can be used.
(3) The task of connecting a piece of hardware to the processor or of adding a software program to the system.

Installation Assistant
An application used to perform system configuration tasks.

installation image
An installation image contains a copy of the software you are installing in backup format, as well as copies of other files the system needs to install the software product.

installation script
A shell script or executable file created by the developer of an application program to install the program. The script file must follow specific guidelines in order to be compatible with the program installation tools that are provided in the operating system.

instance (1) A concrete realization of an abstract object class. An instance of a widget or gadget is a specific data structure that contains detailed appearance and behavioral information that is used to generate a specific graphical object on–screen at runtime. See also *widget instance* on page 2-262 and *instance record* on page 2-116. For Ada programming, see *generic unit* on page 2-100.
(2) An object–oriented programming term synonymous with ”object”. An instance is a particular instantiation of a data type. It is simply a region of storage that contains a value or group of values. For example, if a class `box` is previously defined, two instances of a class `box` could be instantiated with the declaration: `box box1, box2;`
(3) In XOM, an object in the category represented by a class.

instance record
A particular widget record that contains the data objects pertaining to the identity of any given widget as an instance. See also *record* on page 2-196, *widget record* on page 2-262, and *instance* on page 2-116.

instant duration locks
Locks that behave as if they were obtained and then immediately released. These are commonly used when a large number of data objects require locking, but the transaction wishes to avoid obtaining some of the necessary locks while waiting an unknown amount of time for another transaction to release the other required locks. These are primarily useful for implementing schemes such as Jim Gray’s key–range locking (see also his *Transaction Processing: Concepts and Techniques*).

instantiate (1) To make an instance of; to replicate.
(2) In object–oriented programming, to represent a class abstraction with a concrete instance of the class.
(3) In AIXwindows, to create a specific concrete instance of that general class.
(4) To create or generate a particular instance (or object) of a data type. For example, an instance `box1` of class `box` could be instantiated with the declaration: `box box1;`

instruction A program statement that specifies an operation to be performed by the computer, along with the values or locations of operands. This statement represents the programmer’s request to the processor to perform a specific operation.

instruction address register (IAR)
A system control register containing the address of the next instruction to be run. The IAR can be accessed by way of a supervisor call in supervisor state, but cannot be directly addressed in problem state. Synonymous with *program counter* on page 2-187. See also *location counter* on page 2-133.

instruction cache
A cache for providing program instructions to the processor faster than they can be obtained from RAM.

instruction pointer (IP)
See *instruction address register* on page 2-117.

instruction scheduling
A compiler optimization that schedules instructions to the different processing units of the CPU so that maximum instruction overlap can occur.

int specifier In C language, one of the words **int**, **short**, **short int**, **long**, **long int**, **unsigned**, **unsigned int**, **unsigned short**, **unsigned short int**, **unsigned long**, or **unsigned long int**,which describe the type of data a variable represents.

integer A positive or negative whole number or zero.

integer constant
(1) A decimal constant containing no decimal point.
(2) An octal or hexadecimal constant.
(3) A string of decimal digits containing no decimal point.

integer expression
An arithmetic expression with only integer type values.

integer type An arithmetic data type that consists of integer values. In Ada language, an integer type is a discrete type whose values represent all integer numbers within a specific range.

integral object In the C language, a character object, an object having an enumeration type, or an object having the type **short**, **int**, **long**, **unsigned short**, **unsigned int**, or **unsigned long**.

integrity A protection level that may be specified in secure RPC communications that ensures that data transferred between two principals has not been modified in transit.

intention locks Locks used for hierarchical resources that permit an application to avoid locking a large data object, such as a file of records, when it only needs to modify a portion of the data object, such as one record in a file. Obtaining the intention lock on the file indicates that some records in the file may be changing currently, but allows access to records that are not further locked against access.

interaction A continuing cycle of reciprocal action between a user and one or more software applications through input devices (a keyboard, mouse, file server, and so on) and output devices (a display, printer, file server, and so on).

interactive Pertaining to an activity that involves requests and replies, such as between a system user and a program or between two programs.

interactive processing
A processing method in which each system user action causes response from the program or the system. Contrast with *batch processing* on page 2-19.

interface (1) A common boundary, but not of internal connections. An interface can be a hardware component to link two devices or a portion of storage or registers accessed by two or more computer programs. Synonymous with *command interpreter* on page 2-42.
(2) A shared boundary between two functional units, defined by functional characteristics, common physical interconnection characteristics, signal characteristics, and other characteristics.
(3) Hardware or software, or both, that link systems, programs, or devices.
(4) Synonymous with *shell* on page 2-218.
(5) A set of operations. The Network Computing Architecture specifies a Network Interface Definition Language for defining interfaces. See also *extended interface* on page 2-86 and *limited interface* on page 2-129.

interface, extended
See *extended interface* on page 2-86.

interface icon Any pictorial representation of a selection choice appearing within an interface.

interface, limited
See *limited interface* on page 2-129.

interior gateway
A gateway that communicates only with gateways on its own autonomous system.

interlacing A method of doubling vertical resolution by displacing odd video frames by one–half scan line.

interlanguage call
In a program written in a given language, any reference to a procedure written in a different language.

intermediate data type
Any of the basic data types in terms of which the other, substantive data types of the interface are defined.

intermediate nodes
On the widget tree, widgets with one or more children. See also *widget tree* on page 2-262.

intern The procedure of defining an atom.

internal clocking
In data communications, data clocking provided by an adapter.

internal data definition
A description of a variable appearing at the beginning of a block that directs the system to allocate storage for that variable and makes that variable accessible to the current block.

internal data structures
The format of other data stored internally by a program or function. Used to perform type conversion on argument lists.

internal routine
A routine available only within the lexical scope in which it was declared.

international character support
Synonym for *national language support* on page 2-150.

International Standards Organization (ISO)
An international body that standardizes goods and services. For Enhanced X–Windows, standards relating to character sets and fonts.

Internet (1) A wide area network connecting thousands of disparate networks in industry, education, government, and research. The Internet network uses TCP/IP as the standard for transmitting information.
(2) Any wide area network connecting more than one network.

Internet address
The numbering system used in TCP/IP Internetwork communications to specify a particular network or a particular host on that network with which to communicate. Internet addresses are commonly denoted in dotted decimal form.

Internet Control Message Protocol (ICMP)
A protocol used by a gateway to communicate with a source host, for example, to report an error in datagram. It is an integral part of Internet Protocol (IP).

Internet Protocol (IP)
The protocol that provides the interface from the higher level host–to–host protocols to the local network protocols. Addressing at this level is usually from host to host.

Internet Router Enables an IP host to act as a gateway for routing data between separate networks that use a specific adapter.

internetwork Any wide area network connecting more than one network.

interoperability
Ability of a system or a product to work with other systems or products without special effort on the part of the customer. Becomes a quality of increasing importance for information technology products as the concept that ”the network is the computer” becomes a reality.

interpreted routine
A routine that decodes instructions written as pseudocodes and immediately executes those instructions. See also *compile* on page 2-44.

interpreter A tool that allows a program to be run immediately, without recompiling or relinking.

interprocedural analysis
The process of inspecting referenced procedures for information on relationships between arguments, returned values, and global data.

inter–process communication (IPC)
(1) Used for programs to communicate data to each other and to synchronize their activities. Semaphores, signals, and internal message queues are common methods of inter–process communication.
(2) In Enhanced X–Windows, a communication path. See also *client* on page 2-38.

interrupt (1) In data communications, to take an action at a receiving station that causes the sending station to end a transmission.
(2) To stop a process temporarily.
(3) A signal sent by an I/O device to the processor when an error has occurred or when assistance is needed to complete I/O. An interrupt usually suspends the running of the program that is currently running. Contrast with *exception* on page 2-84 and *signal* on page 2-219.

interrupt–confirmation packet
In X.25 communications, a packet used to acknowledge the receipt of an interrupt packet. See also *packet* on page 2-166.

interrupt packet
In X.25 communications, an expedited packet that is allowed to overtake normal data packets (which are delivered in sequence). See also *packet* on page 2-166.

intrinsic function
In FORTRAN, a function that is supplied with the run–time environment that performs math, character, logical, or bit–manipulation operations.

intrinsics In Enhanced X–Windows, a set of management mechanisms that provides for constructing and interfacing between composite widgets, their children, and other clients. Also, intrinsics provide the ability to organize a collection of widgets into an application.

inverse A matrix that results from a mathematical operation on a matrix such that the two matrices can be multiplied together to obtain the unit matrix.

inverse transpose
The inverse of a matrix after it has been transposed

inverted index An index into a bibliography database that allows for direct access. This can be compared to the thumb cuts of a dictionary, which allow a user to move to the exact location of the word being searched.

invocation stack
A list of programs linked together as a result of programs calling other programs within the same job. Synonymous with *program stack*.

invoke To start a command, procedure, or program.

invoke ID An integer used to distinguish one (directory) operation from all other outstanding ones.

IO See *IOU* on page 2-121.

I/O See *input/output* on page 2-115.

I/O channel See *input/output channel* on page 2-115.

I/O configuration
The collection of channel paths, control units, and I/O devices that attaches to the processor unit.

I/O error An error in the processing of input or output device data.

I/O Stream Library
A C++ class library that provides the facilities to deal with many varieties of input and output.

IOCC See *input/output channel controller.* on page 2-115

IOCDS I/O configuration data set.

IOT fault A signal (SIGIOT) that abnormally ends a process.

IOU (IO) Input/output unit.

IP See *Internet Protocol* on page 2-120.

IP address An address in a network using the Internet Protocol. It is a 32–bit integer usually written in dotted decimal notation, where each successive eight bits is translated to an integer and separated from the other components using a dot.

IP socket The port concatenated with the Internet Protocol (IP) address.

IPC See *inter–process communication* on page 2-120.

IPL See *initial program load* on page 2-114.

IPL device initial program load device.

ips Inches per second, a measure of tape drive speed and performance.

ISC See *initial sequence controller* on page 2-114.

ISO See *International Standards Organization* on page 2-119.

ISO 7776 The ISO description of LAPB–compatible DTE data link procedures.

ISO 8208 See *X.25* on page 2-266.

isolation level A field in the specification of an open file descriptor (OFD) that determines the degree to which operations performed with that OFD are isolated from other operations using that OFD. Operation isolation is achieved by the use of locks obtained internally on behalf of each operation. The value of this field determines how locks are obtained and held on behalf of operations performed using that OFD.

ITE In CDE, Internal Terminal Emulator. ITE allows use of a bitmapped display as a terminal (through command–line mode from the login screen).

item The data in one line of an indexed field.

J

Japanese Industry Standard (JIS)
A standard of coding character sets.

Japanese Shift–In start delimiter
In SNA Server, an optional feature supported by X.21 Physical Link Control.

JIS See *Japanese Industry Standard* on page 2-123.

Java Language
A programming language based on the C++ language and developed by Sun Microsystems. Java was developed to include methods for Internet data manipulation. Java applications can be written once and run on any machine having a Java Virtual Machine as part of its operating system.

job (1) A unit of work defined by a user to be done by a system. The term ”job” sometimes refers to a representation of the job, such as a set of programs, files and control statements to the operating system.
(2) One or more related procedures or programs grouped into a procedure, identified by appropriate job control statements. See also *process* on page 2-186.

job number A number assigned to a job as it enters the system to distinguish the job from other jobs.

job queue A list of jobs waiting to be processed by the system.

journaled file system
The standard sequential structure of database files used in this operating system.

journaling (1) The process of recording changes made in a physical file member in a journal.
(2) The process of recording information sequentially in a database.

jump In the running of a computer program, a departure from the implicit or declared order in which instructions are being run.

jumper A connection on an adapter that changes the operating characteristics of a device. For example, it could set up the I/O address or interrupt type of a port.

justify To print a document with even right and left margins.

K

kanji A graphic character set consisting of symbols used in Japanese ideographic alphabets.

katakana A character set of symbols, used primarily to write foreign words phonetically, contained in one of the two common Japanese phonetic alphabets.

Kb Kilobit

KB Kilobyte

K–byte See *kilobyte* on page 2-125.

kernel (1) The part of an operating system that contains programs for such tasks as input/output, management and control of hardware, and the scheduling of user tasks.
(2) The part of the Base Operating System containing functions needed frequently.

kernel device driver
See *device head* on page 2-68.

kernel dump Synonym for *system dump* on page 2-220.

kernel mode The state in which a process runs kernel code. Contrast with *user mode* on page 2-254.

kernel parameters
Variables that specify how the kernel allocates certain system resources. Synonymous with *system parameters*.

kerning The placement of characters such that their bounding boxes (character boxes) overlap. This allows for a more natural–looking spacing between characters.

key (1) One or more characters used to identify a record and establish the record's order within an indexed file.
(2) A unique identifier (of type **key_t**) that names the particular interprocess communications member.
(3) Identifies the name of the shared library text image.
(4) An identifier within a set of data elements. (5.) A character string that matches a definition in a key table.

key click See *keyboard click* on page 2-124.

key grabbing In Enhanced X–Windows, keys on the keyboard can be passively grabbed by a client. Or the keyboard can be actively grabbed by the client when a key is pressed. See also *grab* on page 2-102, *button grabbing* on page 2-17, *pointer grabbing* on page 2-177, *active grab* on page 2-5 and *passive grab* on page 2-171.

key pad A physical grouping of keys on a keyboard such as the numeric key pad and the cursor key pad.

key range The two key fields signifying a range of records to be processed sequentially. The range of records is selected by specifying key values that bound the records to be selected, or by specifying an individual key value for which all matching records should be selected in a non–unique index.

keyboard An input device consisting of various keys that allows the user to input data, control cursor and pointer locations, and to control the dialog with the workstation.

keyboard click (key click)
Transient pulses or surges on a transmission line set up by the opening or closing of keying circuit contacts.

keyboard grabbing
In Enhanced X–Windows, a client can actively grab control of the keyboard and key events will be sent to that client rather than the client to which the events would normally have been sent.

keyboard mapping
A list, usually in a profile, that establishes a correspondence between each key on the keyboard and the character displayed on a display screen, or action taken by a program, when that key is pressed. See also *mapping* on page 2-141.

keyboard send–receive
See *keyboard send–receive mode* on page 2-125.

keyboard send–receive (KSR) mode
A mode in which a graphics display emulates a standard ASCII terminal during both input and output functions.

keyboard traversal
An X widget resource that allows users to move the keyboard focus and activate user interface components using a key sequence rather than a mouse.

keylock feature
(1) A security feature in which a lock and key can be used to restrict the use of the display station.
(2) A program that restricts use of the keyboard.

keysym An encoding of a symbol on a keycap on a keyboard.

keyword (1) A predefined word in a programming language. A reserved word.
(2) In programming languages, a lexical unit that characterizes some language construct. A keyword normally has the form of an identifier.
(3) A name or symbol that identifies a parameter.
(4) Part of a command operand that consists of a specific character string.

kg See *kilogram* on page 2-125.

kill An operating system command that stops a process.

kill character Character that deletes a line of characters entered after a prompt.

kilobyte (K–byte)
1024 bytes in decimal notation when referring to memory capacity; in all other cases, it is defined as 1000.

kilogram (kg) One thousand grams; 2.2046 pounds.

kprocs A kernel parameter that establishes the maximum number of processes the kernel can run simultaneously.

KSR See *keyboard send–receive mode* on page 2-125.

KTS Key Telephone System. A private telephone system requiring manual selection of outside lines.

kVA Kilovolt–ampere, or 1000volt–amperes.

L

L1 cache The first cache accessed when a storage reference occurs.

L2 cache The cache that is accessed, on certain ESCALA models, if the L1 cache lookup results in a cache miss. Normally, the L2 cache is larger and slower than the L1 cache, but faster than RAM.

label (1) A name in the disk or diskette table of contents that identifies a file.
(2) The field of an instruction that assigns a symbolic name to the location at which the instruction begins.
(3) In programming languages, a construction naming a statement and including an identifier. See also *file name* on page 2-90.
(4) An identifier followed by a colon, used to identify a statement in a program. Usually the target of a goto or switch statement. See also *statement label* on page 2-225.

labeled statement
(1) A programming language statement that contains one or more identifiers followed by a colon and a statement.
(2) A possibly empty statement immediately preceded by a label.

LAN See *local area network* on page 2-132.

landscape display
A rectangular display wider than it is high. See also *portrait display* on page 2-180.

landscape left A page orientation such that the left side of the printed image is at the trailing edge of the paper as it emerges from the printer.

landscape right
A page orientation such that the right side of the printed image is at the trailing edge of the paper as it emerges from the printer.

LAP See *link–access procedures* on page 2-130.

LAPB Link–access procedure balanced. See also *link–access procedures* on page 2-130.

last line mode A command mode in the vi editor. Enables the user to enter a command at the bottom of the screen. See also *text input mode* on page 2-240.

latency The time from the initiation of an operation until something actually starts happening (for example, data transmission begins).

layout The arrangement of printed matter on the page, including margins, line spacing, type specification, header and footer information, indents, and more. Synonymous with *geometry* on page 2-100.

lb See *pound* on page 2-181.

LC See *link control* on page 2-130.

LCN See *logical channel number* on page 2-134.

leaders An evenly spaced row of dots used in a table of contents to guide the eye from the title to the page number.

leaf A page of text. See also *page* on page 2-167.

leaf delta A set of changes to the source code saved in the Source Code Control System (SCCS) file.

leaf entry A directory entry that has no subordinates. It can be an alias entry or an object entry.

leap seconds An infrequent adjustment to UTC to account for the irregularity of the earth's rotation.

leased facility Synonym for *nonswitched line* on page 2-155.

leaves On a widget tree, widgets with no children. See also *widget tree* on page 2-262.

LED See *light–emitting diodes* on page 2-129.

left–adjust The process of aligning lines of text at the left margin or at a tab setting such that the leftmost character in the line or file is in the leftmost position.

left margin The area on a page or screen between the left edge and the leftmost character position on the page or screen.

length specification
A source language specification of the number of bytes to be occupied by a variable.

letter An uppercase or lowercase character from the set A through Z.

level (1) The version of a software application program.
(2) See also *higher layer* on page 2-107.
(3) In X.25 communications, see also *physical level* on page 2-175, *packet level* on page 2-167, and *frame level* on page 2-96.

level 1 Synonym for *physical level* on page 2-175.

level 2 Synonym for *frame level* on page 2-96.

level 3 Synonym for *packet level* on page 2-167.

lexical analyzer
A program that analyzes input and breaks it into categories, such as numbers, letters, or operators.

lexical element In Ada language, an identifier, a literal, a delimiter, or a comment.

lexical level The depth to which routines are nested within one another, which determines the scope of the identifiers declared within those routines.

lexical scope The portion of a program or segment unit in which a declaration applies. An identifier declared in a routine is known within that routine and within all nested routines. If a nested routine declares an item with the same name, the outer item is not available in the nested routine.

library (1) A collection of functions, calls, subroutines, or other data.
(2) A data file that contains copies of a number of individual files and control information that allows them to be accessed individually.
(3) In Ada–language library management, a database that stores the various intermediate code files produced by the compiler and records the dependency and order of compilation information as required by the Ada language specification. When compiling a unit that depends on other (previously compiled) units, the required dependency information (such as the package specification of a unit that is included with a **with** clause) is obtained from the library. Similarly, when an Ada program unit is to be linked, the library specifies the set of units that must be included to create an executable image.

library component
In Ada language, a package body, package specification, subprogram body, subprogram specification, object form module, or linked object module that resides in a library.

library list file In Ada language, a text file containing the name of one or more sublibraries comprising an Ada program library.

library unit In Ada language, one of five syntactic entities: a subprogram declaration, a package declaration, a generic declaration, a generic instantiation, or a subprogram body in the case where there is no corresponding subprogram declaration. As the name implies, a library unit resides in the Ada program library. The significance of library units is that they may be referenced by other independently compiled units. This reference may either be explicit (referenced via a **with** clause) or implicit (such as the reference of a package body to its specification).

license An instance of permission to use a licensed software product or service. Sometimes, a user needs more than one license to use a product.

license annotation
A special data string that modifies the use of a license in a manner defined by the vendor of the software product.

license database
The database of licenses maintained by a license server. The license database file– **lic_db** –resides in the **/usr/lib/netls/conf** directory.

license information
The information that describes licenses. This information consists of the product name, the product version, the number of licenses, the license type, the start and end dates of the licenses, the target type, the target ID, and a time stamp.

license password
A string encoded with license information for a software product.

license server daemon
A software program that administers licenses for software products, invoked with the command **netlsd.** The **netlsd** command can be found in the **/usr/lib/netls/bin** directory.

licensed product
A software product that has been enabled by a software vendor for use with the License Use Management system. Enablement allows a vendor to enforce end–user compliance to their license agreement.

iFOR/LS A run–time license–management application based on Gradient Technologies' Version 2.0.1 (Version 1.1.2a) of the Network Licensing System. The system allows software vendors to bundle compliance mechanisms with their software. In tracking license usage, iFOR/LS allows customers to easily comply with their software license agreements.

iFOR/LS **Test Product**
The product used by the **ls_tv** tool to verify that license servers are working properly.

licensed program (LP)
(1) A software program that remains the property of the manufacturer, for which customers pay a license fee.
(2) A separately priced program and its associated materials that bear a copyright and are offered to customers under the terms and conditions of a licensing agreement.

lifetime analysis
The process of inspecting references to variables to determine whether the final assignment to a variable needs to be stored or can be discarded.

ligature Two (or occasionally more) characters printed together so they are connected.

light–emitting diodes (LED)
A semiconductor chip that gives off visible or infrared light when activated. An LED is often used to display operator information.

lighted programmable function keyboard (LPFK)
An input device used primarily in graphic applications and that has lighted keys under control of an application program.

limited interface
A set of system calls that provides a limited function interface. See also *interface* on page 2-119 and *extended interface* on page 2-86.

limited subset A small part or simpler version of a larger set of data or programs.

limited type In Ada language, a *type* for which neither assignment nor the predefined comparison for equality is implicitly declared. All task types are limited. A private type can be defined to be limited. An equality operator can be explicitly declared for a limited type.

line (1) A horizontal display on a screen.
(2) The part of a data circuit that connects to data circuit–terminating equipment (DCE), or to data switching exchange (DSE), or that connects several DCEs or DSEs.
(3) A string of characters accepted by a system as a single block of input from a workstation, such as all characters entered before a carriage return.
(4) See *X.25 line* on page 2-266.

line adapter A functional unit that converts the serial–by–bit input to a station. See also *communications line adapter* on page 2-43.

line editor An editor that displays data one line at a time and that allows data to be accessed and modified only by entering commands.

line height The vertical measurement of a line of text, measured from the bottom of one line to the bottom of the next line. Line height is usually expressed in points.

line number For the Ada–language debugger, a line relative to the Ada compilation that contains the specified compilation unit.

line of memory The section of memory that corresponds to a cache line, which corresponds to a single virtual–memory address tag.

line pacing The sending of a line followed by a waiting interval before continuing transmission. See also *pacing* on page 2-166.

line printer A printer that prints output, one line of characters at a time, as a unit. Output of line printers is in constant–width characters.

line speed (1) The rate at which data is transmitted from one point to another over a telecommunication line.
(2) The number of binary digits that can be sent over a telecommunication line in one second, expressed in bits per second (bps). Synonym for *speed*.

line switching Synonym for *circuit switching* on page 2-36.

linear interpolation
A method of approximating data values by assuming that they lie along a straight line. Typically, the two end data points are known. For example, if A is the value at a, and B is the value at b, and a<t<b, then the value C at t is (from the two–point formula): B – A divided by b – a, multiplied by t – D, added to A.

linefeed An ASCII character that causes an output device to move forward one line.

linestyle The pattern used to draw a line. A linestyle might be solid or broken into a pattern of dashes.

linewidth The width of a line in pixels.

link (1) In the file system, a connection between an i–node and one or more file names associated with it.
(2) In data communications, a transmission medium and data link control component that together transmit data between adjacent nodes.
(3) In programming, the part of a program that passes control and parameters between separate portions of the computer program.
(4) To interconnect items of data or portions of one or more computer programs, such as linking object programs by a linkage editor or linking data items by pointers.
(5) See *X.25 link* on page 2-266. (6.) See *hypertext link* on page 2-109.

link–access procedures (LAP or LAPB)
In X.25 communications, the link level elements used for data interchange between a DCE and a DTE operating in user classes of service 8 to 11, as specified in CCITT Recommendation X.1. LAPB is a duplex, asynchronous, symmetric protocol, used in point–to–point communication. See also *LAPB* on page 2-126.

link address An address assigned at initialization that identifies a channel or control unit and allows it to send and receive frames and perform I/O operations. A set of computers sharing a network that does not include bridges of wide area network links.

link anchor The reference point giving the location of a particular link.

link control (LC)
See *logical link control* on page 2-135.

link descriptor In ODM, a named variable of type **link** used to define a relationship between an object in an object class and an object in another object class. See also *descriptor* on page 2-67.

link–editing To create a loadable computer program by means of a linkage editor.

link level See *frame level* on page 2-96.

link station The part of data link control that is responsible for the transfer of data on a single logical link.

link target See *target* on page 2-237.

link trace A sequential log of events that occur on the link. This log can help determine the source of a recurring error.

linkable The state of an Ada–language program when all its compilation–unit dependencies have been resolved. To produce an executable file, the compiler does not need to do any translation of Ada source; it only needs to call the linkage editor.

linkage editor A program that resolves cross–references between separately compiled object modules and then assigns final addresses to create a single relocatable load module. If a single object module is linked, the linkage editor simply makes it relocatable.

linker See *linkage editor* on page 2-130.

LIS Logical IP Subnet. An LIS is comprised of some number of ATM stations. LISs are analogous to traditional LAN segments.

LISP A programming language designed for list processing and used extensively for artificial intelligence problems.

LISP code Program instructions written in the LISP programming language.

LISP mode For text editors, a mode in which symbols used in the LISP programming language, such as ((left parenthesis) and]] (double right bracket), are treated as symbols, not as editor commands.

list (1) A data object consisting of a collection of related records.
(2) An ordered set of data.

list–directed In FORTRAN, an input/output specification that uses a data list instead of a FORMAT specification.

list–directed data
In FORTRAN, data that is transferred between main storage and an I/O device according to the length and type of variables in the I/O list. See also *formatted data* on page 2-94.

list fields See *indexed fields* on page 2-113.

listen In the X.25 API, to be prepared to receive incoming calls that satisfy criteria specified in an entry in the routing list, through a specified X.25 port.

listen identifier In the X.25 API, an identifier used to listen for and receive an incoming call.

listening Programs waiting for network communication on a given socket are listening on that socket. See also *socket* on page 2-221 and *port* on page 2-179.

literal (1) A symbol or a quantity in a source program that is itself data, rather than a reference to data.
(2) In programming languages, a unit that directly represents a value. For example, 14 represents the integer 14.
(3) In Ada language, a literal represents a value literally, that is, by means of letters and other characters. A literal is either a numeric literal, an enumeration literal, a character literal, or a string literal.

literal string A string that does not contain pattern–matching characters and can therefore be interpreted just as it is. Contrast with *regular expression* on page 2-198.

little endian An attribute of data representation that reflects how multi–octet data are stored in memory. In little endian representation, the lowest addressed octet of a multi–octet data item is the least significant. See also *endian* on page 2-80 and *big endian* on page 2-20.

little endian order
The method of storage in which integer values are stored least significant byte first. See also *big endian order* on page 2-20.

LLB See *Local Location Broker* on page 2-132.

llbd The local location broker daemon.

LLC See *logical link control* on page 2-135.

LMS Line monitoring system.

LNS See *LU network services component* on page 2-138.

load (1) To transfer programs or data from storage into an area of memory where the program can be run or where the data can be manipulated.
(2) To place a diskette into a diskette drive. (3.) To insert paper into a printer. See also *call* on page 2-29.

load level The balance of work between processing units, channels, or devices.

load module See *run file* on page 2-208.

load–store bound
Where the delay in a series of computations is caused by the amount of data that must be loaded into registers or stored back into memory.

loader A program that reads run files into main storage so that the files can be run.

local (1) Pertaining to a device, file, or system that is accessed directly from your system, without the use of a communications line. Contrast with *remote* on page 2-200.
(2) Pertaining to information that is defined and used only in one subdivision of a computer program.

local address The address specified for the current network or host. The local address is usually referred to as the local host address or the local network address to differentiate the two types.

local area network (LAN)
(1) A network in which communications are limited to a moderate–sized geographic area (1 to 10 km) such as a single office building, warehouse, or campus. A local network services a facility without the use of common carrier facilities, although they may be interconnected using common carriers. A local network depends upon a communications medium capable of moderate to high data rate (1 to 20 M bytes per second), and normally operates with a consistently low error rate.
(2) A data network in which serial transmission is used for direct data communication among data stations.

local cell The cell to which the local machine belongs. See also *foreign cell* on page 2-94.

local echo On a communications workstation, a situation in which each signal is displayed twice, once when entered at the local workstation and again when returned over the communications link.

local host In TCP/IP, the host on the network at which a particular operator is working. Synonymous with *current host*.

local listen A link station at the local node that is waiting for an incoming call from a remote station.

Local Location Broker (LLB)
(1) Part of the NCS Location Broker. A server that maintains information about objects on the local host. The LLB also provides the Location Broker forwarding facility.
(2) A service that provides an interface to the global location broker from the iFOR/LS server. The LLB daemon (**llbd**) has no information about network–wide services. It runs continuously in the background to intercept and forward information to the **glbd**. See also *Location Broker* on page 2-133 and *Location Broker Client Agent* on page 2-133.

local name A name that is meaningful and usable only from within the cell where the entry exists. The local name is a shortened form of a global name. Local names begin with the prefix /.: and do not contain a cell name.

local pacing Pacing generated by the local system in an attempt to control the output from the remote system that is input to the local system. See also *pacing* on page 2-166.

local scope A name declared in a block has local scope and can only be used in that block.

local storage A device accessed directly (without telecommunications) from the user's system, where information can be retained and later retrieved.

local variable A symbol defined in one program module or procedure that can only be used within that program module or procedure.

locale A subset of a user's environment that defines conventions for a specified culture, such as time formatting, numeric formatting, monetary formatting, and character classification, conversion, and collation

locality of reference
The degree to which a running program makes use of a compact range of addresses for instructions and/or data.

Location Broker
In NCS, a set of software including the Local Location Broker, the Global Location Broker, and the Location Broker Client Agent. The Location Broker maintains information about the locations of objects. See also *broker* on page 2-18, *Local Location Broker* on page 2-132, and *Location Broker Client Agent* on page 2-133.

Location Broker Client Agent
Part of the NCS Location Broker. Programs communicate with Global Location Brokers and with remote Local Location Brokers using the Location Broker Client Agent. See also *Local Location Broker* on page 2-132 and *Location Broker* on page 2-133.

location code A path from the adapter in the processor through the signal cables and fan out box, if there is one, to the device or workstation. The code consists of four fields of information: Drawer, Slot, Connector, and Port.

location counter
A counter in the assembler that denotes the next byte available for code allocation. The location counter assigns storage addresses to program statements. See also *instruction address register* on page 2-117.

locator In computer graphics, an input device that provides coordinate data; for example, a mouse, tablet, or thumb wheel.

locator resolution
The density of points on a locator device.

locator sample rate
The rate of input from a locator device. Synonymous with *sample rate*.

lock A mechanism with which a resource is restricted for use by the holder of the lock. See also *record lock* on page 2-197.

LOCK See *Lock Service* on page 2-133.

Lock button In CDE, a Front Panel control used to lock the screen.

lock file n multiprocess applications, a system file on disk that the sharing processes use to control their access to shared data or devices.

Lock Service (LOCK)
The component of Encina Base that enables transactions to lock resources before accessing or modifying them.

log (1) To record. For example, to record all messages on the system printer.
(2) A list of messages, such as an error log.
(3) A collection of messages or message segments placed in an auxiliary storage device for accounting or data collections purposes.

log file (1) The text file that records messages and errors from the license server, and sometimes from licensed products, which resides in the **/usr/lib/netls/conf** directory.
(2) In Ada language, a file that contains a record of your commands and comments and the debugger's responses. This log can be used for later analysis, for documenting program behavior, or for making comparisons after program modification.

log force An action that causes all pending log records to be written to permanent storage. A log force is usually associated with committing a transaction, and ensures that the log records associated with that transaction are actually present in the log. Until a log force is done, these records might only be stored in memory and might, therefore, be vulnerable to system failures.

log force groups
A logical association of the log records associated with different processes. Grouping the log records associated with all processes spawned by a specific transaction provides a convenient way to ensure that all of the log records associated with any processes acting on behalf of a specific transaction can be referred to with a single expression or operation.

log in (1) To begin a session at a display station.
(2) The act of gaining access to a computer system by entering identification and authentication information at the workstation.

log off To end a session with a computer system at a display station.

log on See *log in* on page 2-134.

log out See *log off* on page 2-134.

log record A predefined structure into which the log data is formatted. Records have a specific size and format, and contain a certain set of related information. A log record can be identified by its log sequence number (LSN).

log volume An abstract representation of disk space that is used for storage by the Encina log server. There are two types of log volumes: permanent and archival. Permanent volumes can be stored on file or disk devices. Archival volumes can only be stored on file devices. Internal log server data and log file groups must be stored on permanent volumes. Log archive groups must be stored on archival volumes. See also *volume* on page 2-259, *logical volume* on page 2-136, and *physical volume* on page 2-175.

logarithm A mathematical operation related to the base of a numbering system.

logger (1) A functional unit that records events and physical conditions, usually with respect to time.
(2) A program that enables a user entity to log in (for example, identify itself, its purpose, and time of entry) and log off with the corresponding data. This enables the appropriate accounting procedures to be carried out in accordance with the operating system.

logical channel
In X.25 communications, a means of two–way simultaneous transmissions across a data link, comprising associated send and receive channels. A logical channel can represent the path that data travels from its origin to the network or from the network to its destination. See also *channel* on page 2-33.

logical channel number (LCN)
An umber that uniquely identifies a logical channel.

logical constant
A constant with a value of true or false.

logical device (1) A file for conducting input or output with a physical device.
(2) A file for mapping user I/O between virtual and real devices.

logical expression
An expression consisting of logical operators, relational operators, or both that can be evaluated to a value of either true or false.

logical link The logical connection between an application on the S/370 and an application on the workstation.

logical link control (LLC)
In a local area network, the protocol that governs the assembling of transmission frames and their exchange between data stations, independently of the medium access control protocol. See also *medium access control* on page 2-142.

logical name A name assigned to a device that distinguishes it from all other device instances in the system. It is the name used to refer to a particular device. For example, ”tok0” can refer to a token–ring adapter. This is the same as ”device name,” which is a field in the Customized Devices Object Class. See also *device name* on page 2-68.

logical network
A subnetwork of machines set up to function as a whole and separate network. A logical network usually functions as a subnetwork of a larger physical network.

logical operation
An operation that follows the rules of Boolean logic.

logical operator
A symbol that represents an operation, such as AND, OR, or NOT, on logical expressions.

logical partition (LP)
(1) One to three physical partitions (copies). The number of logical partitions within a logical volume is variable.
(2) A fixed–size portion of a logical volume. A logical partition is the same size as the physical partitions in its volume group. Unless the logical volume of which it is a part is mirrored, each logical partition corresponds to, and its contents are stored on, a single physical partition. See also *logical volume* on page 2-136.

logical primary A primary that can have a value of true or false.

logical resource
A software construct, such as a lock or a buffer, that is required for the execution of a program and is in limited supply.

logical storage A conceptual storage layout in which an application maps first into logical addresses, which are then mapped into real addresses by control blocks.

logical type A data type that contains the values of true and false.

logical unit (LU)
(1) A type of network addressable unit that enables end users to communicate with each other and gain access to network resources.
(2) In SNA, a port through which an end user accesses the SNA network to communicate with another user, and through which the end user accesses the functions provided by system services control points (SSCPs). An LU can support at least two sessions, one with an SSCP and one with another LU, and may be capable of supporting many sessions with other LUs.

Logical Unit Type 1 (LU1)
An SNA session that supports communication between an application and multiple input/output devices. This communication could occur in an interactive or batch environment.

Logical Unit Type 2 (LU2)
An SNA session that uses a 3270 device data stream to support communication between an application and a display.

Logical Unit Type 3 (LU3)
An SNA session that uses a 3270 device data stream to support communication between an application and a printer.

Logical Unit Type 6.2 (LU6.2)
(1) An SNA session between two applications in a distributed data processing environment.
(2) The LU type used for SNA advanced program–to–program communications (APPC). See also *peer–to–peer communications* on page 2-172.

logical volume (LV)
(1) A collection of physical partitions organized into logical partitions all contained in a single volume group. Logical volumes are expandable and can span several physical volumes in a volume group.
(2) A set of logical partitions, each of which is stored on one or more physical partitions from one or more of the physical volumes of a given volume group. A logical volume has a device name (of the form /dev/hdn) and contains a single file system. See also *log volume* on page 2-134, *migration installation* on page 2-145, *logical partition* on page 2-135, and *volume group* on page 2-259.

Logical Volume Manager (LVM)
Manages disk space at a logical level. It controls fixed–disk resources by mapping data between logical and physical storage and by allowing data to span multiple disks and to be discontiguous, replicated, and dynamically expanded.

login directory The directory you access when you first log in to the system.

login name A string of characters that uniquely identifies a user to the system.

login session The period of time during which a user of a workstation can communicate with an interactive system, usually the elapsed time between log in and log off.

login shell The shell that is started when a user logs into the computer system. The login shell for a particular user is determined by the entry in the **/etc/passwd** file for that user. See also *shell* on page 2-218.

long (1) In ODM, a terminal descriptor type used to define a variable as a signed 4–byte number. See also *terminal descriptor* on page 2-239. (2) A signed 4–byte number.

long constant A 4–byte integer constant followed by the letter ”l” or ”L.”

long queue status
Synonym for *long status* on page 2-136.

long status A detailed, multiline status that contains more information about each job than the normal short status. Synonymous with *long queue status*.

loop (1) A sequence of instructions performed repeatedly until an ending condition is reached.
(2) A closed unidirectional signal path connecting input and output devices to a system.

loop collapse In nested array–processing loops, an optimization that collapses the nested loops into a single loop with an iteration count that is the product of the iteration counts of the original loops, and that adjusts array indices appropriately.

loop defactorizing
An optimization that removes an invariant factor from a loop that sums calculations into a scalar. The summation scalar can be multiplied by the factor on exit from the loop.

loop elimination
A form of loop unrolling in which the loop is completely unrolled, and references to loop index within the unrolled loop are replaced by constant values.

loop fusion An optimization that takes the bodies of loops with identical iteration counts and fuses them into a single loop.

loop nest reordering
An optimization that changes the order of loops within a loop nest, to achieve stride minimization or to eliminate data dependencies.

loop overhead The CPU time used by a loop that cannot be attributed to computations within the loop.

loop peeling An optimization that improves the performance of a loop that maps an array to a *cylindrical coordinate system*.

loop rerolling An optimization that transforms user–unrolled loops into their original, unrolled equivalents, so that other optimizations can be attempted.

loop unrolling An optimization that increases the step of a loop, and duplicates the expressions within a loop to reflect the increase in the step. This can improve instruction scheduling and memory access time.

looping statement
A statement that runs any number of times, depending on the value of a specified expression.

low–order Least significant; rightmost. For example, in a 32–bit register (0 through 31), bit 31 is the low–order bit.

LP See *licensed program* on page 2-128.

LPFK See *lighted programmable function keyboard* on page 2-129.

LPM Lines per minute. The number of lines a printer can print in one minute.

LPR Line Printer Server.

LQ Letter quality.

LRU Least recently used.

ls_admin In License Use Management, the software program used to modify a license server database, invoked with the command **ls_admin**, which is located in the **/usr/lib/netls/bin** directory.

ls_rpt In License Use Management, the software program that reports on the history of license server events, invoked with the command **ls_rpt**, which is located in the **/usr/lib/netls/bin** directory.

ls_stat In License Use Management, the software program that reports on the status of licenses, invoked with the command **ls_stat**, located in the **/usr/lib/netls/bin** directory.

ls_tv In License Use Management, the network license server daemon test and verification tool, invoked with the command **ls_tv**, located in the **/usr/lib/netls/bin** directory.

lsb Least significant bit.

LSB Least significant byte.

LU See *logical unit* on page 2-135.

LU1 See *Logical Unit Type 1* on page 2-135.

LU2 See *Logical Unit Type 2* on page 2-135.

LU3 See *Logical Unit Type 3* on page 2-135.

LU6.2 See *Logical Unit Type 6.2* on page 2-134.

LU, dependent A logical unit that cannot start a conversation but must wait for the host system to start the conversation.

LU, independent
A logical unit that can start a conversation with another logical unit.

LU–LU session
In SNA Server, a session between two logical units (LUs) of the same type that supports communication between two end users, or between an end user and an LU services component.

LU network services component (LNS)
Begins and ends LU–LU sessions in response to requests from the resource manager and from the remote LU. It also activates and deactivates CP–LU sessions.

lvalue (1) An expression that represents a data object that can be both examined and altered.
(2) The left–hand part of an expression.

LV See *logical volume* on page 2-136.

LVM See *Logical Volume Manager* on page 2-136.

M

m See meter on page 2-144.

MAC (1) See *medium access control* on page 2-142.
(2) Mandatory Access Control.

machine execution state
A state that indicates the machine is shut down, booting, or running. This state is one of two machine states.

machine instruction
(1) A binary number that directs the operation of a processor. Compilers and assembler convert source instructions to machine instructions.
(2) An instruction of a machine language. Synonym for *computer instruction* on page 2-47 and *computer language*.

machine language
A language that can be used directly by a computer without intermediate processing. The final output of the compilation process is a load module containing machine language instructions. Synonym for *computer language*.

machine object
An entry in the Network Installation Management database that represents a machine configuration.

machine state A state that identifies the machine execution state and control state for each machine.

machine word Synonym for *word* on page 2-263. See also *computer language* on page 2-47.

macro (1) A label that is declared at the start of a program or file. The label can then be used to represent the values assigned to the label in the declaration.
(2) A name or label used in place of a number of other names.
(3) The sequence of instructions or statements that a macrogenerator runs when replacing a macro instruction.
(4) A set of statements defining the name of, format of, and conditions for generating a sequence of assembler statements from a single source statement.
(5) A series of Ada–language debugger commands that execute in sequence when you call the macro. With the macro option, you can define and manipulate new debugger commands. See also *routine* on page 2-207, *statement function* on page 2-225, and *subroutine* on page 2-231.

macro call A single instruction that, when executed, causes the execution of a predefined sequence of instructions in the same source language.

macro instruction
See *macro call* on page 2-139 and *macro* on page 2-139.

macro processor
A program that converts macro instructions into specified values.

magic number A numeric or string constant in a file that indicates the file type.

mail Correspondence in the form of messages transmitted between workstations over a network. Synonymous with *electronic mail*.

mail box A storage location in a network to which messages for a user are sent.

mail drop The file into which messages are first received.

Mailer In CDE, an application that enables you to send, receive, and compose electronic mail messages.

mailer The program that does the actual delivery of mail.

mailer container
In CDE, the electronic mail box and filing system that contains all mail messages. Once a message is put in a container, you can display, modify, delete, print, include, forward, and reply to it.

Mailer control In CDE, the Front Panel control used to start the Mailer software application. Dropping a file on the control loads the file into the Mailer Compose window.

main In FORTRAN, the default name given to a main program if one was not supplied by the programmer.

main function A function that has the identifier **main**. Each C language program must have exactly one function named **main**. This function is the main program of a C language program.

Main Panel In CDE, the portion of the Front Panel excluding the subpanels.

main program (1) The first program unit to receive control when a program is run. Contrast with *subprogram* on page 2-230.
(2) A program that performs primary functions, passing control to routines and subroutines for the performance of more specific functions.

main storage (1) Program–addressable storage or memory from which instructions and other data can be loaded directly into registers for subsequent running or processing. Synonymous with *system memory* on page 2-220.
(2) The part of internal storage into which instructions and other data must be loaded for running or processing.
(3) The part of the processing unit where programs are run.

mainframe A large computer, particularly one to which other computers can be connected so that they can share facilities the mainframe provides. The term usually refers to hardware only.

maintenance analysis procedure (MAP)
Documentation used by customer engineers and by service representatives to repair equipment. A MAP contains yes/no questions and procedures that direct the user to the failing part of the equipment.

maintenance level update
The service updates (fixes and enhancements) that are necessary to upgrade the Base Operating System (BOS) or an optional software product to the current release level. See also *service update* on page 2-216.

maintenance mode
State in which a product or system can be serviced. Synonymous with *service mode*.

maintenance system
A special version of the operating system that is loaded from diskette and used to perform system management tasks.

major device number
A system identification number for each device or type of device. The major device, minor device, and channel numbers uniquely identify a hardware device. See also *minor device number* on page 2-145.

managed children
In Enhanced X–Windows, children in which the managed field has a value of **True** can have their layout (geometry) changed so that they can be repositioned and resized.

managed window
See *managed children* on page 2-140.

manager See *device manager* on page 2-68.

Manager class In AIXwindows, a metaclass that provides the resources and functionality to implement certain features, such as a keyboard interface and traversal mechanism. It is built from the **Core**, **Composite**, and **Constraint** classes.

mangling The encoding, during compilation, of C++ identifiers such as function and variable names to include type and scoping information. The linker uses these mangled names to ensure type–safe linkage.

manual call In data communications, a line type that requires the operator to place a call over a switched line. Contrast with *auto–call* on page 2-15.

manual dialing In making an ATE connection, dialing the number manually over a telephone line.

map See *mapping* on page 2-141.

MAP See *maintenance analysis procedure* on page 2-140.

mapped In Enhanced X–Windows, a window is said to be mapped if a map call has been performed on it.

mapped conversation
A temporary connection between an application program and an advanced program–to–program communication (APPC) session in which the system provides all the protocol information. It allows the two programs to exchange data records of any length and in any format specified by the transmission programs. Only LU6.2 sessions allow mapped conversation; it is used primarily for application transaction programs. Contrast with *basic conversation* on page 2-19.

mapped file (1) A file that can be accessed through direct memory operations instead of being read from disk each time it is accessed.
(2) A file on the fixed disk that is accessed as if it is in memory.

mapping (1) In Enhanced X–Windows, a window on which a map call has been performed. Mapping makes a window visible if there are no obscuring or occluding windows.
(2) A list, usually in a profile, that establishes a correspondence between items in two groups. For example, a keyboard mapping can establish what character is displayed when a certain key is pressed. See also *keyboard mapping* on page 2-125 and *profile* on page 2-186.

margin Left and right border of text on a screen or hardcopy page.

mark block In Pascal, a dynamic block header that designates a subheap within a heap.

marker (1) A visual symbol within a non–interactive pane indicating the location of the cursor when the pane was last interactive.
(2) In computer graphics, a glyph with a specified appearance that is used to identify a particular location.

marshal In NCS, to copy data into a Remote Procedure Call (RPC) packet. Stubs perform marshalling. Contrast with *unmarshal* on page 2-252. See also *stub* on page 2-229.

mask (1) A pattern of characters that controls the keeping, deleting, or testing of portions of another pattern of characters or bits, usually through an AND or OR operation.
(2) To apply a mask.

master The only machine in the NIM environment that has permission to remotely execute commands on other NIM clients.

master dump table
A structure containing dump table entries generated by kernel components. The dump program uses this table to locate data structures that should be included in a dump.

master file (1) A collection of permanent information, such as a customer address file. (2) A file that is used as an authority in a given job and that is relatively permanent, even though its contents may change.

master processor
The first processor started at boot time in a multiprocessor system.

master server In a network installation environment, the server that has permissions to execute commands on all other machines in the environment. The master server is designed to manage the network, client, and resource objects in the network installation database.

matrix (1) A rectangular array of elements arranged in rows and columns that can be manipulated based on matrix algebra rules.
(2) In computers, a logic network in the form of an array of input and output leads with logic elements joined at some of their intersections. (3.) By extension, an array of any number of dimensions.

matrix stack In GL, a stack of matrices with hardware and software support. The top matrix on the stack is the current transformation matrix, and all points passed through the graphics pipeline are multiplied by that matrix. It is a concatenation of the current modeling and viewing transformations. See also *current transformation matrix* on page 2-42.

maximum transfer unit (MTU)
(1) The maximum number of bytes that an Internet Protocol (IP) datagram can contain.
(2) The largest amount of data that can be transmitted in a single frame for a particular network interface.

Mb Megabit.

MB Megabyte.

M–bit In X.25 communications, the bit in a data packet that indicates that there is more data to follow in another data packet (when a message is too large for one packet).

mbuf A small (256–byte) buffer provided by the mbuf management facility to the various layers of communication software in the operating system.

M–byte See *megabyte* on page 2-142.

MC See *MCU* on page 2-142.

MCU (MC) Memory control unit.

medium access control
In a local area network, the protocol that governs communication on the transmission medium without concern for the physical characteristics of the medium. However, it takes into account the topological aspects of the network, to enable the exchange of data between data stations. See also *logical link control* on page 2-135.

megabyte (MB) Loosely, one million bytes. When referring to semiconductor memory capacity, two to the twentieth power; 1 048 576 in decimal notation. When referring to media device storage, a megabyte is ten to the sixth power (1 000 000).

megahertz (MHz)
A unit of measure of frequency. One megahertz equals 1 000 000 hertz.

member. (1) A data object in a structure, a union, or a library.
(2) Synonym for *element* on page 2-79.
(3) A C++ data object or function in a structure, union or class. Members can also be classes, enumerations, bit fields and type names.

member function
C++ Operators and functions that are declared as members of a class. A member function has access to the private and protected data members and member functions of an object of its class. Member functions are also called *methods*.

memory (1) Program–addressable memory from which instructions and other data can be loaded directly into registers for subsequent running or processing.
(2) Memory on electronic chips. Examples of memory are random access memory, read–only memory, or registers. See also *storage* on page 2-227.

memory deallocation
To free up memory that has been previously allocated for a specific purpose.

memory dump The means by which the computer system records its state at the time of a failure.

memory image The logical layout of the parts of a process in memory.

memory leak A software bug in which the program allocates memory, loses track of it, and then allocates some more. If the program is long–running, it can eventually tie up large amounts of real memory and paging space. System performance gradually deteriorates; the program that finally fails due to lack of resource may not be the culprit. Memory leaks in kernel extensions that allocate pinned memory may be particularly costly.

memory load control
A VMM facility that detects memory over–commitment and temporarily reduces the number of running processes, thus avoiding thrashing.

memory over–commitment
A condition in which the number of virtual–memory pages being used by the currently running programs exceeds the number of real–memory page frames available to hold them. If the over–commitment is large or sustained, system performance suffers.

menu A displayed list of items from which an operator can make a selection.

menu bar A rectangular area at the top of the client area of a window that contains the titles of the standard pull–down menus for that application.

menu cursor In AIXwindows, the cursor defined for a particular menu. Each type of cursor is identified by an ID number.

menu pane The physical window containing a pop–up menu listing a group of options to be chosen by the user. See also *pane* on page 2-168.

menu system An interactive interface that lists related software options in a manner that expedites review and selection by the user.

message (1) Information from the system that informs the user of a condition that may affect further processing of a current program.
(2) An error indication, or any brief information that a program writes to standard error or a queue.
(3) Information sent from one user in a multiuser operating system to another.
(4) A general method of communication between two processes.
(5) A group of characters and control bit sequences transferred as an entity.
(6) One or more linked blocks of data or information, with associated STREAMS control structures containing a message type. Messages are the only means of communicating within a stream.

message control block
In the X.25 API, the structure used to indicate what type of packet has arrived and to point to the structure that contains the packet information.

message queue
A linked list of messages connected to a QUEUE.

message queue ID (msqid)
An identifier assigned to a message queue for use within a particular process. It is similar in use to a file descriptor of a file.

message type A defined set of values identifying the contents of a message.

metaclass In AIXwindows and Enhanced X–Windows, an object class that does not instantiate widgets or gadgets but is capable of passing a unique set of inheritable resources to the subclasses beneath it in the class hierarchy. Each instance of a widget subclass has the features common to that widget class and exports these features to child widgets of that class. Included in this class are **Core**, **Composite**, **Constraint**, **Primitive**, **Button**, **Manager**, **MenuMgr**, and **MenuPane**.

metadata The structural data associated with the file system, such as the organization of directories, inode tables, and links. Metadata is not data supplied by a user; it is information about the structure of user data.

meter (m) A linear measurement that equals 1.0936 yards, 3.2808 feet, or 39.3696 inches.

method (1) In Enhanced X–Windows, the functions or procedures that a widget itself implements.
(2) In ODM, executable code associated with an object and defined as the value of a method descriptor for the object. The method can be a command, program, or shell script. See also *method descriptor* on page 2-144.
(3) An object–oriented programming term synonymous with *member function*.

method descriptor
In ODM, a named variable of type **method** used to define a method or operation to associate with an object. The method can be any executable code such as a command, program, or shell script. See also *method* on page 2-144 and *descriptor* on page 2-67.

MHz See *megahertz* on page 2-143.

MIB Variable A managed object that is defined in the Management Information Base (MIB). The managed object is defined by a textual name and a corresponding object identifier, a syntax, an access mode, a status, and a description of the semantics of the managed object. The MIB Variable contains pertinent management information that is accessible as defined by the access mode.

microdocument
A document within a frame that has its own properties and shares components with the main document.

migration installation
An installation method for upgrading to the current release while preserving the existing root volume group. This method preserves the **/usr**, **/tmp**, **/var**, and **/** (root) file systems, as well as the root volume group, logical volumes, and system configuration files. See also *root volume group* on page 2-206 and *logical volume* on page 2-136.

mil A measurement of thickness: 1/1000 inch.

millisecond A measurement of time: 1/1000 of a second.

minimally consistent
Said of an object that satisfies various conditions set forth in the definition of its class.

minor device number
A number that specifies various types of information about a particular device. For example, a number that distinguishes between several printers of same type. See also *major device number* on page 2-140.

mirroring The creation of a mirror image of a primitive.

mixed string A string consisting of a mixture of DBCS characters and single–byte characters.

mm (1) A package of macros for manuscript preparation that supports the **eqn** and **troff** commands or the **neqn** and **nroff** commands and features annotation, footnoting, indexing, and tables by supporting the **tbl** command.
(2) Millimeter.

mnemonic (1) A symbol chosen to help the user remember the significance of the symbol.
(2) The field of an assembler instruction that contains the acronym or abbreviation for a machine instruction.
(3) In CDE, a single, underlined character in a command. The mnemonic indicates that you can choose the command by typing that letter. Mnemonics are used most commonly in menu commands; however, other buttons may also have mnemonics.

mnemonic overstrike
A nonstandard character created by printing one character on top of another to represent a nonprinting character.

modal The state in which a secondary window receives keyboard or pointer input that it does not pass on to its associated window.

modal dialog In AIXwindows, a **Dialog** widget that interrupts the work session to solicit input from the user.

modal interaction
The communication between separate modes or functions.

modal pop–up In Enhanced X–Windows, a window that normally is not visible to the window manager and available only after the manager is turned off. This pop–up disables user–event processing except for events that occur in the dialog box.

mode (1) A method of operation.
(2) In SNA data communications, the set of rules and protocols to be used for a session.
(3) In the M–Video Capture Adapter, a method of operation such as live or overlay capture.
(4) For Ada programming, see *parameter* on page 2-169.

model number In Ada language, an exactly representable value of a real type. Operations of a real type are defined in terms of operations on the model numbers of the type. The properties of the model numbers and of their operations are the minimal properties preserved by all implementations of the real type.

modeless dialog
In AIXwindows, a **Dialog** widget that solicits input from the user but does not interrupt the work session.

modeless pop–up
In Enhanced X–Windows, a window that is normally visible and is controlled by the window manager.

modeling coordinates, modeling space
In GL, the coordinate system in which all drawing primitives do their drawing. The user can select the position and orientation of the modeling space with regard to the world space by means of translations, rotations, scales, or generalized transformations. The relation between modeling coordinates and world coordinates is determined by the modeling matrix. Modeling coordinates are a useful conceptual device when drawing complex or repetitive scenes. For instance, a paper clip can be defined once in modeling coordinates, and then drawn hundreds of times by moving the modeling coordinates around in world space. See also *eye coordinates* on page 2-88, *screen coordinates* on page 2-210, *world coordinates* on page 2-265, and *transformation* on page 2-245.

modem (modulator–demodulator)
A device that converts digital data from a computer to an analog signal that can be transmitted on a telecommunication line, and converts the analog signal received to digital data for the computer.

modem eliminator
A device that connects a workstation directly to a computer port through a wired connector with a specific pin arrangement. When two devices both function as DTEs (data terminal equipment), the cable that connects them must transmit send and receive signals using a modem eliminator. Synonymous with *null modem*.

mode name (1) The name of an entry in the login mode table.
(2) In SNA, identifies the set of rules and protocols to be used for the session.

mode word An i–node field that describes the type and state of the i–node.

modification number
The modification level of a program, which is an indicator of changes that do not affect the external interface of the program. The version, release, modification, and fix levels together comprise the *program level* on page 2-187, *fix number* on page 2-92, *release number* on page 2-199, and *version number* on page 2-257.

modifier Word or quantifier used to change an instruction causing the execution of an instruction different from the original one. Consequently, the same instruction, successively changed by a modifier, can be used repetitively to carry out a different operation each time it is used.

modifier key In CDE, a key that when pressed and held along with another key or mouse button changes the meaning of the second key or mouse click. Control, Alt, and Shift are examples.

modifier keys In Enhanced X–Windows, keys such as Shift, Shift Lock, Control, Alt, Caps Lock, and Meta.

modulation Changing the frequency or size of one signal by using the frequency or size of another signal.

modulator–demodulator
See *modem* on page 2-146.

module (1) A discrete programming unit that usually performs a specific task or set of tasks. Modules are subroutines and calling programs that are assembled separately, then linked to make a complete program.
(2) In programming languages, a language construct that consists of procedures or data declarations and that interact with other such constructs.
(3) A packaged functional hardware unit designed for use with other components.
(4) Synonym for *program unit* on page 2-187.
(5) See also *run file* on page 2-208.
(6) Software that performs functions on messages as they flow between stream head and driver. A module is the STREAMS counterpart to the commands in a shell pipeline except that a module contains a pair of functions that allow independent bidirectional (downstream and upstream) data flow and processing.

MOM See *monitor mode* on page 2-147.

mond See *monitor scheduling daemon* on page 2-147.

monitor (1) A device that observes and verifies operations of a data processing system.
(2) A functional unit that observes and records selected activities for analysis within a data processing system. Possible uses are to show significant departures from the norm or to determine levels of utilization or particular functional units.
(3) Synonym for *display* on page 2-72.

monitor mode (MOM)
A mode in which an application program can directly access the display adapter.

monitor scheduling daemon
A process that runs on an application server and provides clients with links to processing agents as needed

monochrome A special case of static gray in which there are only two color map entries. Some monochrome adapters can display shades of gray in the Gray Scale Adapter.

monochrome display
A display device that has only one color. See also *gray scale* on page 2-103.

more–data bit See *M–bit* on page 2-142.

mount To make a file system accessible.

mouse A hand–held locator that a user operates by moving it on a flat surface. It allows the user to select objects and scroll the display screen by pressing buttons.

mouse button 1
On a mouse, the leftmost button when configured for right–handed use; the rightmost button when configured for left–handed use. Mouse button 1 is primarily used for selection, and is the default button for the ”click” instruction.

mouse button 2
On a three–button mouse, the middle button. On a two–button mouse, the right button if configured for right–handed use or the left button if configured for left–handed use. Mouse button 2 has two modes: Transfer and Adjust.

mouse button 3
On a mouse, the rightmost button when configured for right–handed use; the leftmost button when configured for left–handed use. Mouse button 3 activates pop–up menus.

mouse scaling The distance the cursor moves relative to the mouse movement. The scaling factor is either 1:1 or 2:1.

mouse threshold
An operating system parameter that determines the amount of horizontal or vertical mouse movements required to move the cursor on the screen.

ms A package of macros for manuscript and thesis preparation that features automatic footnote numbering and bibliography capabilities.

msb Most significant bit.

MSB Most significant byte.

MS–DOS Microsoft Disk Operating System.

msqid See *message queue ID* on page 2-144.

MTU See *maximum transfer unit* on page 2-142.

multibyte control
One of the two types of controls valid in a character stream data. Synonym for *escape sequence* on page 2-83.

multicast address
A multicast address, for FDDI, is an address with the high order bit in the high–order byte of the address set. This type of address allows a user to have multiple stations on a ring set to the same address. If a packet is sent to a multicast address, all stations with that address receive the packet. A multicast address is sometimes referred to as a group address.

multidrop (1) Stations connected to a multipoint channel at one location.
(2) A network configuration in which there are one or more intermediate nodes on the path between a central node and an endpoint node.

multihomed machine
A machine that has more than one configured network adapter and more than one host name.

multihomed server
A server that has more than one network interface running.

multiline More than one communications line.

multimode optical fiber
A graded–index or step–index optical fiber that allows more than one bound mode to propagate. Contrast with *single–mode optical fiber* on page 2-219.

multiple inheritance
An object–oriented programming technique implemented in C++ through derivation, in which the derived class inherits members from more than one base class.

multiplex To interleave or simultaneously transmit two or more messages on a single channel.

multiplexed device
(1) A device that takes several input signals and combines them into a single output signal so that each of the input signals can be recovered. (2) A device capable of interleaving events of two or more activities or capable of distributing events of an interleaved sequence to the respective activities.

multiplexer See *multiplexed device* on page 2-149.

multipoint Pertaining to communication among more than two stations over a single telecommunications line.

multipoint link A circuit that interconnects several stations.

multitasking A mode of operation that provides for concurrent performance or interleaved processing of two or more tasks. Synonym for tasking.

multiuser mode
A mode of operation that enables two or more users to use the services of a processor within a given period of time. The usage is usually serial unless otherwise specified.

mutex Jargon for *mutual exclusion lock*. Use of this type of lock excludes all threads other than the lock holder from any access whatsoever to the locked resource.

mutual exclusion mechanism
A means for preventing two separately executing pieces of code from interfering with each other's use of a particular data object. For example, if one thread is executing a function that modifies a shared data structure, then the application may need to prevent other threads from attempting to read the data simultaneously, before the modifications are complete.

MVS Multiple Virtual Storage.

MVS/TSO A type of operating system used on a System/370 computer.

mwm AIXwindows window manager. See also *window manager* on page 2-239.

N

n See *en* on page 2-79.

NA Not applicable.

name (1) A sequence of 1 to 250 alphanumeric characters, the first of which must be alphabetic, that identifies a data object.
(2) In Ada language, a name is a construct that stands for an entity: it is said that the name denotes the entity, and that the entity is the meaning of the name. See also *declaration* on page 2-64 and *prefix* on page 2-182.
(3) In C++, a name is commonly referred to as an identifier. However, syntactically, a name can be an identifier, operator function name, conversion function name, destructor name, or qualified name. See also *symbolic name* on page 2-219.

name–lookup method resolution
Similar to the *method resolution* techniques employed by Objective–C and Smalltalk.

name resolution
The process of translating (resolving) a symbolic name into its more efficient Internet address.

name server A host that provides name resolution for a network. Name servers translate symbolic names assigned to networks and hosts into the efficient Internet addresses used by machines.

name string A character string that identifies one of a variety of objects such as an icon, a resource, or a font.

named common
In FORTRAN, a separate common block consisting of variables and arrays and given a name.

named pipe A pipe that an application opens by name in order to write data into or read data from the pipe. By convention, named pipes are placed in the **/dev** directory and are treated as special files. Using a named pipe facilitates communication between a sending process and a receiving process.

naming scope See *scope* on page 2-210.

NaN See *not–a–number* on page 2-155.

national language support
Conversion subroutines for languages other than American English that translate between various character sets and date and time string formats. Synonymous with *international character support*.

national terminal number (NTN)
In X.25 communications, the 1– to 12–digit number that follows the country code in the network user address.

natural or social science format
The style of bibliography entry favored by writers in the natural and social sciences.

navigation keys
In CDE, the keyboard keys used to move the current location of the cursor. These include the arrow keys (with or without the Control key); the Tab key (with or without the Control or Shift keys); the Begin and End keys (with or without the Control key); and the Page Up and Page Down keys.

NCA See *Network Computing Architecture* on page 2-151.

NCCF See *Network Communications Control Facility* on page 2-152.

NCK See *Network Computing Kernel* on page 2-152.

NCS See *Network Computing System* on page 2-152.

NCS cell A logical concept of grouping together one or more machines in an NCS network. Any node belonging to an alternate cell may only have their license requests satisfied by License Use Management servers in that cell. Nodes outside the cell may not make license requests to servers in another cell. Two types of cells, default and alternate, are used to provide two ways of accessing iFOR/LS servers.

NDC See *normalized device coordinates* on page 2-155.

negative response
In data communications, a reply indicating that data was not received correctly or that a command was incorrect or unacceptable.

negotiation In X.25 communications, the process by which two DTEs establish the packet size, packet window size, and throughput class to be used during a call procedure. Contrast with *validation* on page 2-255.

neighbor gateway
One of the peers acquired by an exterior gateway. All exterior gateways do not communicate with all other exterior gateways. Instead, they acquire neighbors through which they communicate.

nest (1) To incorporate a structure or structures into a structure of the same kind.
(2) To place subroutines or data in other subroutines or data at a different hierarchical level. Therefore, the subroutines can be run as recursive subroutines or so that the data can be accessed recursively.
(3) A self–contained software element that completely encompasses a similar software element is said to have that similar element ”nested” within it. Examples include software loops (the nested loop) nested within larger loops (the nesting loop) and submenus nested within menus.

nested class A C++ class defined within the scope of another class.

nested DO In FORTRAN, a DO loop or DO statement in which the range is entirely contained within the range of another DO loop.

nested transaction
A transaction begun within the scope of another transaction. These are also referred to as subtransactions.

NETASCII Eight–bit ASCII with the first bit always set high, for error checking.

NetBios Network Version of Basic Input/Output System.

NetLS See iFOR/LS on page 2-128.

netlsd The command used to invoke the network license server daemon.

network A collection of data processing products that are connected by communication lines for information exchange between locations.

network adapter
Circuitry that allows devices to communicate with other devices on the network.

network address
(1) The part of an address indicating a specific network. A complete address for a machine on a network consists of the network address and the host address.
(2) In NCS, a unique identifier (within an address family) for a specific host on a network or an internet. The network address is sufficient to identify a host, but does not identify a communication end point within the host.

network boot image
A boot image that supports standalone, diskless, and dataless machines.

Network Communications Control Facility (NCCF)
A licensed program that serves as a base for command processors that can monitor, control, and improve the operation of a network.

Network Computing Architecture (NCA)
A set of protocols and architectures that support distributed computing.

Network Computing Kernel (NCK)
The combination of the RPC runtime library and the Location Broker, which contain the necessary pieces required to run distributed applications.

Network Computing System (NCS)
A set of software tools developed by Apollo Computer Inc. that conform to the Network Computing Architecture. These tools include the Remote Procedure Call runtime library and the Location Broker. NCS is the underlying communications protocol used by iFOR/LS to transmit licensing transactions between clients and servers. Messages are broadcast from clients to the NCS–managed network.

Network File System (NFS)
A distributed file system that enables users to access files and directories located on remote computers and treat those files and directories as if they were local. NFS is independent of machine types, operating systems, and network architectures through the use of remote procedure calls (RPC).

Network Information Center (NIC)
The publication distribution center for DARPA TCP/IP information.

Network Installation Management (NIM)
An environment that provides installation and configuration of software within a network interface.

network interface
The software that formats packets at the network layer into packets that specific network adapters can understand and transmit.

Network License System
See iFOR/LS on page 2-128.

network management
The conceptual control element of a data station that interfaces with all of the layers of that data station and is responsible for the setting and resetting control parameters, obtaining reports of error conditions, and determining if the station should be connected to or disconnected from the medium.

network mask (netmask)
A 32–bit mask used to identify the most local portion of a local area network (LAN).

network object An entry in the Network Installation Management (NIM) database that represents a local area network.

network protocol
A communications protocol from the Network Layer of the OSI network architecture, such as the Internet Protocol (IP).

network provider
In X.25 communications, the organization, often a PTT, that provides a public network.

network state A state that indicates either that the network object can participate in NIM operations or an error in the definition of the network object.

network terminating unit (NTU)
In X.25 communications, the point of access to the network.

Network Time Protocol (NTP)
Internet–recommended time standard.

network user A kernel–level protocol or user–level application that accesses the services of the network layer.

network user address (NUA)
In X.25 communications, the X.121 address containing up to 15 binary code digits.

network user identification (NUI)
(1) The facility that enables the transmitting DTE to provide billing, security, or management information on a per–call basis to the DCE.
(2) The NUI can identify a network user independent of the port being used.
See also *address* on page 2-6.

new In C++, a keyword identifying a free store allocation operator. The **new** operator may be used to create class objects.

new installation
An installation method used when the fixed disk or disks you are installing BOS onto are *empty.* A hard disk is considered empty if it does not contain any data or if it contains data not in a volume group.

new–line character (NL)
A control character that causes the print or display position to move down one line. This character is represented by '\n' in the C language. Usually a carriage return is implicitly associated with an NL.

new–process image
A new program laid over the current program by the **exec** subroutine.

next The dialog management action for dependent workstations that causes the next portion of a data object to be presented.

NFS See *Network File System* on page 2-152.

NIC See *Network Information Center* on page 2-152.

nice value A number that is used to bias the priority of a process. A higher number results in a lower priority.

nickname Synonym for *alias* on page 2-7.

NIM See *Network Installation Management* on page 2-152.

NIM routing The information that defines which networks in the NIM environment can communicate with each other and which gateways they use to facilitate that communication. NIM routing is used to represent the TCP/IP routine that exists for the LANs in the overall network environment.

NIS Network Information Service. A distributed database that allows you to maintain consistent configuration files throughout your network.

NL See *new–line character* on page 2-153.

node (1) A computer connected to a network.
(2) An end point of a link, or a junction common to two or more links in a network. Nodes can be processors, controllers, or workstations, and they can vary in routing and other functional capabilities.
(3) In Systems Network Architecture the portion of a hardware component, along with its associated software components, that implements the functions of the seven architectural layers (SNA).
(4) In a tree structure, a point at which subordinate items of data originate.

node ID A unique string of characters that identifies the node on a network.

node verification
An additional level of security beyond that provided by the network addressing scheme. Node verification helps to ensure that a connection reaches the correct remote station. It is available on LU6.2 connections only. See also *BIND password* on page 2-17.

nodelock file The text file at a user node (rather than at a license server node) where nodelocked licenses are added. The **nodelock** file is located in the **/usr/lib/netls/conf** directory.

nodelocked license
A type of license locked to a specific node so that the product may only be used at that node. The license server does not administer nodelocked licenses.

nodes Systems connected in a network to form a monitor cell.

no–input zone Synonym for *dead zone* on page 2-63.

noise (1) A disturbance that affects a signal and potentially distorts the information carried by that signal.
(2) Random variations of the characteristics of any entity, such as voltage, current, or data.
(3) A random signal of known statistical properties of amplitude, distribution, and spectral density.

nolock lock A dummy lock mode that is used when a lock mode must be supplied, but when actually locking the data is not required. Functions specifying this lock mode can read data even when it is currently locked by other operations or transactions, enabling dirty reads to be performed. This type of lock is operationally consistent, which means that using it to lock and read a value will return an actual value that was correct at some moment in time, but is not transactionally consistent.

nondeterministic program
A program whose results cannot be determined by analyzing the source code. A program that does not initialize variables before their first use is nondeterministic, because the value of an uninitialized variable can change between runs of the program.

nonexecutable program unit
In FORTRAN, a block data subprogram.

nonexecutable statement
A statement that describes the characteristics of a program unit, data, editing information, or statement functions, but does not cause any action to be taken.

non–extended result
An exception notification that does not have any data defined in the **result_ext** file of the **dlc_getx_arg** structure.

non–printing character
Synonym for *control character* on page 2-52.

non–productive
Data traffic on the media that is only made up of repetitive control information and does not contain end user data.

non–return–to–zero (NRZ)
A binary code system in which a signal condition must be sustained for the full time interval and does not revert to a standby or quiescent state between signal elements. Using NRZ permits the maximum data signaling rate on the channel, which should be twice the band width, according to Nyquist's theorem.

nonspacing character
See *diacritic* on page 2-68.

nonspacing character sequence
For accented characters, a two–part sequence consisting of a valid diacritic followed by an alphabetic character or a space. The system converts the sequence into a single code point that results in the alphabetic character with the specified diacritic mark.

nonswitched line
(1) A connection between computers or devices that does not have to be established by dialing.
(2) A dedicated line. Synonym for *leased facility*. Contrast with *switched line* on page 2-218.

nonswitched network
On a network, a connection between computers or devices that does not have to be established by dialing.

nonterminal symbol
The structure that the parser recognizes.

nonvolatile random access memory (NVRAM)
Random access memory (storage) that retains its contents after the electrical power to the machine is shut off. A specific part of NVRAM is set aside for use by the system ROS for the boot device list.

nonzero A value that is not equal to zero.

normal attachment stop
See *normal stop* on page 2-155.

normal mode See *multiuser mode* on page 2-149.

normal port A port that is asynchronous and allows users to log in. No outgoing use of the port is allowed while it is enabled.

normal stop One of two ways to stop an attachment. (The other way is a forced stop.) If the attachments or any of their connections are in a pending state, SNA Services rejects the normal stop attachment action.

normalized device coordinates (NDC)
In GL, coordinates in the range from –1 to 1. All primitives that draw within the unit cube are visible on the screen (unless masked by the screen mask). See also *transformation* on page 2-245 and *unit cube* on page 2-251.

not–a–number (NaN)
In binary floating–point computations, a value, not interpreted as a mathematical value, that contains a mask state and a sequence of binary digits.

notify flag A parameter that indicates whether a reply is required at that point in an event loop.

nroff A typesetting utility originally designed to drive Teletype model 37 printing workstations. It now drives a wide variety of backspacing and non–backspacing tty–type line printers and tty–emulating printers.

NRZ See *non–return–to–zero* on page 2-155.

NS Network Systems. Also, Network Services.

NSA Next station addressing.

NSI Name Service Interface.

NTN See *national terminal number* on page 2-150.

NTP See *Network Time Protocol* on page 2-153.

NTSC A national television industry broadcasting standard as defined by the National Television Standard Committee used in the USA, Canada, Japan, and other countries. A video display and timing format that is the American broadcast standard. Most video tape recorders record and play back NTSC signals. Specialized hardware is required to convert from RGB monitor outputs to an NTSC signal. See also *PAL* on page 2-168 and *SECAM* on page 2-212.

NTSC signal A signal as defined by the National Television Standard Committee. Also called *composite video*.

NTU See *network terminating unit* on page 2-153.

NUA See *network user address* on page 2-153.

NUI See *network user identification* on page 2-153.

NUL See *null character* on page 2-156.

NUL character In XPG4 system interface, a character with all bits set to zero.

null Empty, having no value, containing nothing.

NULL In the C language, a pointer guaranteed not to point to a data object.

null character (NUL)
(1) The **hex 00** character used to represent the absence of a printed or displayed character.
(2) A control character used to accomplish media–fill or time–fill that can be inserted into or removed from a sequence of characters without affecting the meaning of the sequence.
(3) A control character used to delimit a string of characters.

null character string
Two consecutive single quotation marks that specify a character string consisting of characters.

null modem See *modem eliminator* on page 2-141.

null signal A signal parameter of 0 (zero).

null statement A statement that consists of a semicolon.

null–terminated
Having a zero byte at the end. In the C language, character strings are stored this way internally.

numeric Pertaining to any of the digits 0 through 9.

numeric character
See *digit*. on page 2-69

numeric constant
A constant that expresses an integer, real, or complex number.

numeric error A situation where the result of a real or integer expression exceeds a compiler's limits (or limits imposed by the programming language) for such expressions.

numeric literal A numeric character or string of characters whose value is implicit in the characters themselves. For example, 777 is literal as well as the value of the of the number 777. A numeric literal can contain any of the numeric digits 0 through 9, a plus or minus sign, and a decimal point. Contrast with *character literal* on page 2-34.

NURBS (Non–Uniform Rational B–spline)
A parametric surface that can be trimmed with nonuniform rational B–spline curves and piecewise linear curves. See also *trimming loops* on page 2-246.

NVRAM See *nonvolatile random access memory* on page 2-155.

O

O Organization.

object (1) In the Network Installation Management (NIM) environment, an entry in the NIM database that represents a machine, network, or resource.
(2) In Enhanced X–Windows, a software abstraction consisting of private data and private and public routines that operate on the private data. Users of the abstraction can interact with the object only through calls to the public routines of the objects.
(3) In ODM, an instance or member of an object class, conceptually similar to a structure that is a member or an array of structures. See also *object class* on page 2-158.
(4) In Pascal, synonymous with *data object* on page 2-61.
(5) In GL, synonymous with *display list* on page 2-72. A sequence of drawing commands that have been compiled into a unit. Conceptually, a display list is like a macro; it can be invoked multiple times simply by referring to its name. The object can be instantiated at different locations, sizes, and orientations by appropriate use of the transformation matrices. For instance, series of polygons arranged in the shape of a bolt can be compiled into an object. The bolt can then be drawn multiple times by invoking its display list.
(6) In NCS, an entity that is manipulated by well–defined operations. Disk files, printers, and array processors are examples of objects. Objects are accessed though interfaces. Every object has a type.
(7) In Ada language, an object contains a value. A program creates an object either by elaborating an object declaration or by evaluating an allocator. The declaration or allocator specifies a type for the object: the object can only contain values of that type. See also *constant* on page 2-51 and *variable* on page 2-251.
(8) In C++, a region of storage. An object is created in C++ when a variable is defined or **new** is invoked. An object is destroyed when it goes out of scope.
(9) In XDS, anything in some "world," generally the world of telecommunications and information processing or some part thereof, that is identifiable (can be named) and for which the DIB contains some information.
(10) In XOM, any of the complex information objects created, examined, modified, or destroyed by means of the interface.
(11) In CDE, any logical piece of data that has associated behavior. For example, in File Manager, files, folders, actions, and applications are all considered objects. Each type of object has specific associated actions. Typically, each object is represented as an icon. See also *type UUID* on page 2-249.

object class In ODM, a stored collection of objects with the same definition, conceptually similar to an array of structures. See also *object* on page 2-158, *terminal descriptor* on page 2-239, and *class* on page 2-36.

Object Class Table (OCT)
A recurring attribute of the directory schema with the description of the object classes permitted.

object code (1) Instructions that are able to be run by a machine, usually generated by a compiler from source code written in a higher–level language (such as C language). For programs that must be linked, object code consists of relocatable machine code.
(2) Output from a compiler or assembler that is itself executable machine code or is suitable for processing to produce executable machine code. Contrast with *source code* on page 2-222.

Object Data Manager (ODM)
A data manager intended for the storage of system data. The ODM is used for many system management functions. Information used in many commands and SMIT functions is stored and maintained in the ODM as objects with associated characteristics.

object definition
See *class* on page 2-36.

object file (1) A member file in an object library.
(2) The primary output of a compiler or assembler, which can be processed by the binder (**ld**) to produce an executable file. The names of object files normally end in**.o**.

object handle In graphical files, the start point of an arc or lines object.

object identifier.
A value (distinguishable from all other such values) that is associated with an information object. (X.208)

object instance
See *instance* on page 2-116 and *object* on page 2-158.

object library An area on a direct access storage device used to store object programs and routines.

object management
The creation, examination, modification, and deletion of potentially complex information objects.

object module (1) A portion of an object program suitable for input to a linkage editor.
(2) A set of instructions in machine language produced by a compiler from a source program. See also *module* on page 2-147.

object–oriented programming
A programming approach based on the concepts of data abstraction and inheritance. Unlike procedural programming techniques, object–oriented programming concentrates not on how something is accomplished but instead on what data objects comprise the problem and how they are manipulated.

object program
A fully compiled or assembled program that is ready to be loaded into the computer. Synonymous with *target program*.

object space The space in which a graphics object is defined. A convenient point is chosen as the origin and the object is defined relative to this point. When an object is rendered by a call to the **callobj** subroutine, it is rendered in modeling coordinates, and the object space becomes (for that moment) the same as the modeling space.

object type In CDE, a desktop mechanism used to associate particular data files with the appropriate applications and actions. Object typing defines the criteria for typing the file (such as name or contents), the appearance (the icon used in File Manager), and the behavior of the object (for example, what happens when you double–click it).

object UUID A UUID that identifies a particular object. Both the RPC runtime library and the Location Broker in NCS use object UUIDs to identify objects. See also *Universal Unique Identifier (UUID)* on page 2-251.

obscure (In Enhanced X-Windows**)**
(1) A state of being for a window. A window is obscured if another window is in front of it making the obscured window partially viewable. Window *B* is obscured by window *A* if both are viewable InputOutput windows and *A* is higher in the global stacking order and the rectangle defined by the outside edges of *A* intersects the rectangle defined by the outside edges of *B*. Window borders are included in the calculation, and a window can be obscured and yet still have visible regions.
(2) An action one window does to another when it partially obstructs the viewing of the other. Window *A* obscures window *B* if both are viewable InputOutput windows, *A* is higher in the global stacking order, and the rectangle defined by the outside edges of *A* intersects the rectangle defined by the outside edges of *B*. Contrast with *widget visibility* on page 2-262.

occlude (In Enhanced X-Windows**)**
(1) A state of being for a window. A window is occluded if the view of it is completely obstructed by another window. Window *B* is occluded by window *A* if both are mapped, *A* is higher in the global stacking order, and if no part of the border of *B* is viewable. A fine distinction exists between ”occlude” and ”obscure.” Window borders are included in the calculation. InputOnly windows never obscure other windows but can occlude other windows.
(2) An action one window does to another when it completely obstructs the view of the other. Contrast with *widget visibility* on page 2-262.

OCS On–Card Sequencer.

OCT See *Object Class Table* on page 2-158.

octal (1) A base–eight numbering system.
(2) Pertaining to a fixed–radix numeration having a radix of eight.

octal constant The digit 0 (zero) followed by any digits 0 through 7.

octet (1) A group of 8 bits (also known as a byte).
(2) Pertaining to a selection, condition, or choice that has eight potential values or states. See also *octet string* on page 2-160.

octet string See *Object Data Manager* on page 2-159.

ODM See *Object Data Manager* on page 2-159.

off–hook Activated, with regard to a telephone set. On a public switched system, a data set that automatically answers is said to go off–hook. Contrast with *on–hook* on page 2-160.

offline (1) Pertaining to the operation of a functional unit when it is not under the direct control of a computer.
(2) Neither controlled directly by nor communicating with the computer. Contrast with *online* on page 2-161.

offset (1) In Pascal, the selection mechanism in the SPACE data type; an element is selected by placing an integer value in brackets. The origin of SPACE is based on zero.
(2) In publications, the indentation of all lines of a block of text following the first line.

OM See *XOM* on page 2-267.

on–hook Deactivated, in regard to a telephone set. A telephone not in use is ”on–hook.” Contrast with *off–hook* on page 2-160.

On Item help In CDE, when you choose On Item from the Help menu, the cursor changes to a question mark and you can obtain information about a particular command, operation, dialog box, or control by clicking that object.

On Item Help control
In CDE, Front Panel control used to access item help information on the Front Panel.

one–way channel
In X.25 communications, a logical channel that allows incoming calls only or outgoing calls only. Contrast with *two–way channel* on page 2-248.

ones complement
The diminished radix complement in the pure binary numeration system. The ones complement is derived by replacing all of the zeros with ones and all of the ones with zeros in a binary number. In a ones complement system, the ones complement of a number *n* is – *n*. See also *twos complement* on page 2-248.

online (1) Being controlled directly by or directly communicating with the computer. Contrast with *offline* on page 2-160.
(2) Pertaining to the operation of a functional unit when under the direct control of a computer.

op code See *operation code* on page 2-162.

opaque See *opaque data type* on page 2-161.

opaque data structure
A data structure used internally by one functional unit of code but exported for limited external use in another functional unit of code. The external uses must avoid accessing the structure's components or making assumptions about its constitution.

opaque data type
In XDR, bytes of a fixed size that are not interpreted as they pass through the data streams between computers.

opaque structure
A data item or data type whose structure is hidden from the code that is handling it.

open To make a file available to a program for processing. Contrast with *close* on page 2-39.

Open Systems Interconnection (OSI)
(1) The interconnection of open systems in accordance with specific ISO standards.
(2) The use of standardized procedures to enable the interconnection of data processing systems.

operand (1) An instruction field that represents data (or the location of data) to be manipulated or operated upon. Not all instructions require an operand field.
(2) An identifier, constant, or expression that is grouped with an operator.
(3) An entity on which an operation is performed.
(4) Information entered with a command name that defines the data on which a command processor operates and that controls the running of the command processor.

operating system (OS)
A set of programs that control how the system works. Controls the running of programs and provides such services as resource allocation, scheduling, input and output control, and data management.

operation (1) A specific action (such as add, multiply, or shift) that the computer performs when requested.
(2) In NCS, a procedure through which an object is accessed or manipulated. An operation is defined syntactically by its name and its parameters but not by its implementation.
(3) In Ada language, an operation is an elementary action associated with one or more types. It is either implicitly declared by the declaration of the type, or it is a subprogram that has a parameter or result of the type.

operation code (op code)
A numeric code that tells the processor which operation to perform.

operator (1) A symbol (such as +, –, or *) that represents an operation (in this case, addition, subtraction, multiplication).
(2) A person who operates a device.
(3) In Ada language, an operator is an operation which has one or two operands. A unary operator is written before an operand; a binary operator is written between two operands. This notation is a special kind of function call. An operator can be declared as a function. Many operators are implicitly declared by the declaration of a type (for example, most type declarations imply the declaration of the equality operator for values of the type).

operator function
An overloaded C++ operator that is either a member of a class or takes at least one argument that is a class type or a reference to a class type.

optimization The process of achieving improved run–time performance or reduced code size of an application. Optimization can be performed by a compiler, by a preprocessor, or through hand–tuning of source code.

optimize To improve the speed of a program or to reduce the use of storage during processing.

optimized unit An Ada–language compilation unit that has been processed by one or more of the compiler's optimizing phases. Some source–level information is unavailable to the debugger when it examines the unit. You can use an optimized unit in the same contexts as the corresponding unoptimized unit. Optimized and unoptimized units can be mixed within an Ada program.

option (1) An item of either hardware or software that may be purchased in addition to the basic system. An option can include cables, an adapter, a warranty, and other items.
(2) A specification in a statement that can influence the running of the statement.
(3) An installable unit of a software package. Software product options are separately installable units that can operate independently from other options of that software package.

option button In Common Desktop Environment, a diamond–shaped button that allows you to select from a number of options.

optional facilities
In X.25 communications, facilities that may or may not be offered by the network provider to which customers choose whether or not to subscribe. See also *closed user group* on page 2-39, *fast select* on page 2-89, *reverse charging* on page 2-204, and *throughput–class negotiation* on page 2-241.

optional software
Also referred to as *optional software products*. Software that is *not* automatically installed on your system when you install the Base Operating System (BOS). Optional software can be products packaged and sold with BOS. Optional software can also be separately purchased software products that are specially ordered and not sold as part of BOS. In either case, BOS must be installed on your system before you can install optional software.

ordinal number
One of the counting numbers, used to indicate position.

ordinal type A type in which members can be counted to indicate position.

ORed Having the logical **OR** operation performed.

organization Data that associates a named set of users who can be granted common access rights that are usually associated with administrative policy. Also, the third field of a subject identifier.

orphaned files Files that cannot be reached by the **fsck** command.

orthographic projection
A representation in which the lines of a projection are parallel. Orthographic projections lack *perspective foreshortening* and its accompanying sense of depth realism. Because they are simple to draw, orthographic projections are often used by draftsmen. See also *perspective projection* on page 2-174.

OS See *operating system* on page 2-161.

OS/2 Operating System/2.

OSF Open Software Foundation.

OSI See *Open Systems Interconnection* on page 2-161.

OSPF Open Shortest Path First routing protocol.

OSS OSI Session Service.

OU Organizational Unit.

outgoing call In X.25 communications, a call being made to another data terminal equipment (DTE).

output (1) The result of processing data.
(2) Pertaining to a functional unit or channel involved in an output process, or to the data or involved in such a process.
(3) Data transferred from storage to an output device.
(4) In Pascal, a predefined standard file definition.

output buffer In Enhanced X–Windows, an area used by the **Xlib** library to store requests.

output device A physical device that a computer uses to present data to a user. Synonymous with *output unit*.

output file (1) A file that a program opens so that it can write to that file.
(2) A file that contains the results of processing.

output handler The program module responsible for distributing data generated by a process or subprocess.

output list A list of variables from which values are written to a file or device.

output mode An open mode in which records can be written to a file.

output redirection
The specification of an output destination other than the standard one.

output stream Messages and other output data that an operating system or a processing program displays on output devices.

output unit Synonym for *output device* on page 2-163.

overflow (1) That portion of an operation's result that exceeds the capacity of the intended unit of storage
(2) In a register, the loss of one or more of the leftmost whole–number digits because the result of an operation exceeded the size of the register.

overflow condition
(1) A condition that occurs when a portion of an operation's result exceeds the capacity of the intended unit of storage.
(2) A condition that occurs when the overflow line on a page has been printed or passed.

overflow line The line specified as the last line to be printed on a page.

overlay (1) To write over (and therefore destroy) an existing file.
(2) A program segment that is loaded into main storage, replacing all or part of a previously loaded program segment.
(3) Repeatedly using the same areas of internal storage during different states of a program.
(4) In DPS, a collection of predefined data such as lines, shading, text, boxes, or logos that can be merged with variable data while printing on a page.
(5) In the M–Video Capture Adapter, the replacement of specified pixels of one source with pixels from another source such as memory pixels overlaid on live images.

overlay planes One or more bit planes in a display buffer that are used to create visual data (text, graphics, and so on) that overlays the visual data in the frame buffer in a nondestructive manner. Rather than a background color, overlay planes are transparent.

overloading (1) In Ada language, an identifier can have several alternative meanings at a given point in the program text: this property is called overloading. For example, an overloaded enumeration literal can be an identifier that appears in the definitions of two or more enumeration types. The effective meaning of an overloaded identifier is determined by the context. Subprograms, aggregates, allocators, and string literals can also be overloaded.
(2) In C++, a capability that allows you to redefine functions and most standard C++ operators when the functions and operators are used with class types.

override (1) A parameter or value that replaces a previous parameter or value.
(2) To replace a parameter or value.

overscan A characteristic of display monitors where a number of lines and horizontal pixels delivered to the monitor exceed the visible display space of the screen.

overstriking A method of generating special characters by typing one character and then, without moving the print head to the next character position, typing the second character on top of the first.

overwrite To record into an area of storage so that the data that was previously stored there is destroyed.

overwrite mode
A form of system operation that replaces existing characters with characters typed at the keyboard.

owner The user who has the highest level of access authority to a data object or action, as defined by the object or action; usually the creator of the object.

ownership The creator or namer of an entity.

P

pacing (1) A technique used by a receiving component to control the rate of transmission by sending a component to prevent overrun.
(2) A file transfer protocol required by some systems. It controls data transmission by waiting for a specified character, or waiting a specified number of seconds between lines. This protocol prevents the loss of data when the block size is too large or data is sent too quickly for the system to process. See also *line pacing* on page 2-129, *local pacing* on page 2-132, *receive pacing* on page 2-196, and *remote pacing* on page 2-200.

pacing response
In SNA Server, an indicator that signifies the readiness of a receiving component to accept another pacing group. The indicator is carried in a response header (RH) for session–level pacing, and in a transmission header (TH) for virtual–route pacing.

pack loop A loop that packs active cells of a sparse array into successive cells in another array, so that the resulting array contains no empty cells between its first and last active cells.

package (1) An installable unit of a software product. Software product packages are separately installable units that can operate independently from other packages of that software product.
(2) In Ada language, specifies a group of logically related entities, such as types, objects of those types, and subprograms with parameters of those types. It is written as a package declaration and a package body. The package declaration has a visible part, containing the declarations of all entities that can be explicitly used outside the package. It may also have a private part containing structural details that complete the specification of the visible entities, but which are irrelevant to the user of the package. The package body contains implementations of subprograms (and possibly tasks as other packages) that have been specified in the package declaration. A package is one of the kinds of program unit. See also *private part* on page 2-185.

package closure
The set of classes that need to be supported to be able to create all possible instances of all classes defined in the package.

packet In data communications, a sequence of binary digits, including data and control signals, that is transmitted and switched as a composite whole. The data, call control signals, and error control information are arranged in a specific format. See also *call–accepted packet* on page 2-29, *call–connected packet* on page 2-29, *call–request packet* on page 2-29, *clear–confirmation packet* on page 2-37, *clear–indication packet* on page 2-37, *clear–request packet* on page 2-37, *data packet* on page 2-61, *incoming–call packet* on page 2-112, *interrupt packet* on page 2-121, *interrupt–confirmation packet* on page 2-120, *reset–request packet* on page 2-202, *reset–confirmation packet* on page 2-202, *address field* on page 2-6, and *restart–confirmation packet* on page 2-204.

packet assembler/disassembler (PAD)
In X.25 communications, equipment used for connecting asynchronous (start/stop) devices to an X.25 network.

packet header In X.25 communications, control information at the start of the packet; the contents of the packet depend on the packet type.

packet level In X.25 communications, the packet format and control procedures for the exchange of packets containing control information and user data between the data terminal equipment (DTE) and the data circuit–terminating equipment (DCE). Synonymous with *level 3*. See also *level* on page 2-127, *frame level* on page 2-96, *datalink level* on page 2-61, and *physical level* on page 2-175.

packet–level interface
In X.25 communications, the level of the DTE/DCE interface in packet mode operation relating to the exchange of data and signaling, where this information is contained in packets. See also *frame–level interface* on page 2-96.

packet mode operation
Synonym for *packet switching* on page 2-167.

packet size In X.25 communications, in the context of data packets, refers to the length of the user data.

packet switching
Routing and transferring data by addressing packets so that a channel is occupied only during packet transmission. On completion of the transmission, the channel is available for transfer of other packets. Synonymous with *packet mode operation*. See also *circuit switching* on page 2-36.

packet window In X.25 communications, the number of packets that can be outstanding without acknowledgment. See also *frame window* on page 2-96 and *window* on page 2-263.

packing In AIXwindows and Enhanced X–Windows, the grouping of children objects within a parent container object. If the children are closely packed, the common distance between their borders is minimal; if they are loosely packed, the common distance border–to–border is maximized.

pad (1) To fill unused positions in a field with dummy data, usually zeros or blanks.
(2) A device used to introduce transmission loss into a circuit. It can be inserted to introduce loss or match impedances.

padding Bytes inserted in the data stream to maintain alignment of the protocol requests on natural boundaries. Padding increases the ease of portability to some machine architectures.

page (1) A block of instructions, data, or both.
(2) The number of lines that can fit into a window.
(3) In a virtual storage system, a fixed–length block that has a virtual address and is transferred as a unit between real storage and auxiliary storage.
(4) A contiguous 4096–byte portion of a virtual–memory segment. The offset of each page from the beginning of the segment is an integral multiple of 4096. See also *leaf* on page 2-126.

page cluster A type of memory buffer that is constructed from a full memory page (normally 4096 bytes).

page fault (1) A program interruption that occurs when an active page refers to a page that is not in memory.
(2) An interrupt that occurs when the processor attempts to access a virtual–memory page that is not in real memory.

page frame (1) In real storage, a storage location having the size of a page.
(2) An area of main storage that contains a page.
(3) A 4096–contiguous–byte portion of real memory that is used to hold a virtual–memory page.

page frame table
A table, contained in real memory, that contains the real memory locations of all currently defined pages.

pagination (1) The process of adjusting text to fit within page margins.
(2) In word processing, the automatic arrangement of text according to a preset number of page layout parameters.

paging (1) The action of transferring instructions, data, or both between real storage and external page storage.
(2) Moving data between memory and a mass storage device as the data is needed.
(3) In System/370 virtual storage systems, the process of transferring pages between real storage and external page storage.

paging device A disk device used to store pages of memory that are not currently in real memory.

paging space Disk storage for information that is resident in virtual memory but is not currently being accessed.

paint In computer graphics, to shade an area of a display image.

PAL (1) Programmable array logic.
(2) A national television industry broadcasting standard used in Europe and some other countries. See also *NTSC* on page 2-156 and *SECAM* on page 2-212.

PAL signal A phase analog lock signal, also called composite video. The European standard for composite video.

palette (1) Location for building customized components and parenting them with other components. Subsequently, components can be reused by copying and moving them to other interfaces.
(2) In CDE, a range of graphically displayed choices, such as colors or collections of tools, that you can select in an application.

pane On a display screen, the inner portion of a window used to present information to the user. A window can consist of one or more panes. See also *menu pane* on page 2-143.

panel (1) A set of logically related information displayed on the screen for the purpose of communicating information to or from a computer user.
(2) A group of one or more panes that are treated as a unit. The panes of a panel are displayed together, erased together, and usually represent a unit of information to a person using the application. A panel is represented on the display as a rectangular area tiled (completely filled) with panes.

panning (1) In computer graphics, the viewing of an image that is too large to fit on a single screen by moving from one part of the image to another.
(2) Progressively translating an entire display image to give the visual impression of lateral movement of the image.

paragraph (1) Text that is separated from other text by blank lines.
(2) In word processing, one or more sentences that maybe preceded by or followed by an appropriate indicator.

parallel channel
Communications protocol between controller and mainframe processors.

parallel device A device that can perform two or more concurrent activities. Contrast with *serial device* on page 2-215.

parallel processing
The condition in which multiple tasks are being performed simultaneously within the same activity. Contrast with *serial processing* on page 2-215.

parallel transmission
(1) Transmitting all bits of a character simultaneously.
(2) In data communication, the simultaneous transmission of a number of signal elements that constitute the same telegraph or data signal.

parameter (1) Information that the user supplies to a panel, command, or function.
(2) A variable that is given a constant value for a specified application.
(3) Data passed between programs or procedures.
(4) In Ada language, a parameter is one of the named entities associated with a subprogram, entry, or generic unit, and used to communicate with the corresponding subprogram body, accept statement or generic body. A formal parameter is an identifier used to denote the named entity within the body. An actual parameter is the particular entity associated with the corresponding formal parameter by a subprogram call, entry call, or generic instantiation. The mode of a formal parameter specifies whether the associated actual parameter supplies a value for the formal parameter, or the formal supplies a value for the actual parameter, or both. The association of actual parameters with formal parameters can be specified by named associations, by positional associations, or by a combination of these. See also *formal parameter* on page 2-94 and *mode* on page 2-146.

parameter block
A block of memory that contains specific parameters for an ioctl operation.

parameter declaration
Description of a value that a function receives. A parameter declaration determines the storage class and the data type of the value.

parametric bicubic surface
A surface defined by three equations. The *x* equation is: x(u,v) = a11u3v3 + a12u3v3 + a13u3v + a14u3 + a21u2v3 + a22u2v2 + a23u2v + a24u2 + a31uv3 + a32uv2 + a33uv + a34u + a41v3 + a42v2 + a43v + a44. The equations for *y* and *z* are similar. The points on a bicubic patch are defined by varying the parameters *u* and *v* from 0 to 1. If one parameter is held constant and the other is varied from 0 to 1, the result is a cubic curve. If *w* (*u*, *v*)=1 for all *u*, *v*, the bicubic surface is called ”ordinary,” but if *w* (*u*, *v*) varies as a function of *u*, *v*, then the surface is called ”rational.” See also *homogeneous coordinates* on page 2-108.

parametric component
In AIXwindows, a simple mechanism that delivers all the functions necessary for most applications, yet is easier and less time consuming to build.

parametric cubic curve
A curve defined by the equation: x(t) = axt3 + bxt2 + cxt + dx; y(t) = ayt3 + byt2 + cyt + dy; z(t) = azt3 + bzt2 + czt + dz; w(t) = awt3 + bwt2 + cwt + dw. Where *x, y, z,* and *w* are cubic polynomials. The parameter *t* typically varies between 0 and 1. Such a curve is considered rational only if a(w), b(w), or c(w) is not equal to 0; otherwise, it is simply an ordinary parametric curve. See also *B–spline cubic curve* on page 2-20, *Bezier cubic curve* on page 2-20, and *cardinal spline cubic curve* on page 2-31.

parent (1) A process that has spawned a child process using the fork primitive.
(2) Pertaining to a secured resource, either a file or library, whose user list is shared with one or more files or libraries. Contrast with *child* on page 2-35.
(3) In AIXwindows and Enhanced X–Windows, a graphical object that controls one or more smaller graphical objects attached to it. The smaller graphical objects are called children, and they are automatically deleted when their parent is deleted.
(4) In Ada language, the associated specification of a package body or subprogram body. The parent of a subunit is the body in which it was declared.

parent device A hierarchical location term. It indicates what device the device you are concerned with connects to. For example, the parent device of an SCSI disk might be an SCSI adapter.

parent directory The directory one level above the current directory. See also *parent folder* on page 2-170.

parent folder In CDE, a folder that contains subfolders and files. When discussing command–line activities, this may be called the *parent directory* on page 2-170. See also *subfolder* on page 2-230.

parent ID The character sequence identifying the graphical object that controls smaller graphical objects, called *children*.

parent type For Ada programming, see *derived type* on page 2-66.

parent window In Enhanced X–Windows, the window that controls the size and location of its children. If a window has children, it is a parent window.

parity bit A binary digit (bit) appended to a group of binary digits to make the sum of all digits in the group either always odd (odd parity) or always even (even parity).

parity check A test to determine whether the number of ones (or zeros) in an array of binary digits is odd or even.

parity error A transmission error that occurs when the received data does not have the parity expected by the receiving system. This error is usually caused by the sending and receiving systems having different parity settings.

parse (1) In systems with time sharing, to analyze the operands entered with a command and create a parameter list for the command processor from the information.
(2) Before a command line interpreter can convert an operating–system command into an executable form of machine code, the command must be broken down into easily coded elements, or *"*parsed*"*, by the interpreter.

parser A program that interprets user input and determines what to do with the input. See also *grammar rules* on page 2-102.

participant An application is a participant in a transaction when it either initiates the transaction or receives a request on behalf of that transaction.

partition (1) A logical division of storage on a fixed disk.
(2) A fixed–size division of storage.

partner In data communications, the remote application program or the remote computer.

Pascal A high–level, general–purpose programming language. Programs written in Pascal are block–structured, consisting of independent routines.

pass–by–CONST
In Pascal, the parameter–passing mechanism by which the address of a variable is passed to the called routine. The called routine is not permitted to modify the formal parameter. Synonymous with *pass–by–read–only–reference*.

pass–by–read–only–reference
Synonym for *pass–by–CONST* on page 2-171.

pass–by–read/write–reference
Synonym for *pass–by–VAR* on page 2-171.

pass–by–value In Pascal, the parameter–passing mechanism by which a copy of the value of the actual parameter is passed to the called routine. If the called routine modifies the formal parameter, the corresponding actual parameter is not affected.

pass–by–VAR In Pascal, the parameter–passing mechanism by which the address of a variable is passed to the called routine. If the called routine modifies the formal parameter, the corresponding actual parameter is also changed. Synonymous with *pass–by–read/write–reference*.

pass–through function
The ability to pass data through a program transparently, without alteration.

pass–through mode
The mode of use provided by the VM/Pass–ThroughFacility, which allows VM display station users to interactively access a VM system, including the one to which the terminal is attached. It also allows users to access non–PVM systems that support Remote 3270 Binary Synchronous Communication (BSC) display stations and 4300 processors having the Remote Operator Console Facility (ROCF). A user can access, log on to, and use another system in a defined network as though the user's local terminal were directly connected to that system. PVM activities become transparent to the user once logged on to the target system.

passive gateway
A gateway that does not exchange routing information. Its routing information is contained indefinitely in the routing tables and is included in any routing information that is transmitted. Contrast with *active gateway* on page 2-5.

passive grab In Enhanced X–Windows, grabbing a key or button is a passive grab. The grab becomes an active grab when the key or button is actually pressed. Contrast with *active grab* on page 2-5. See also *grab* on page 2-102, *button grabbing* on page 2-17, *pointer grabbing* on page 2-177, and *key grabbing* on page 2-124.

password (1) A string of characters known only to the user and the system. The user must specify the correct password to gain access to a system and the data stored with it.
(2) A string encoded with information about a software vendor (vendor password) or about a software product (product password).

password security
The process of requiring a user to enter a password to log in to a system.

patch (1) In SNA, the series of path control network components traversed by the information exchanged between two network addressable units (NAUs). A path consists of a series of path control elements, data link control elements, and links.
(2) In a network, any route between any two nodes.
(3) In a database, a sequence of segment occurrences from the root segment to an individual segment.
(4) In CDE, a text string that specifies the hierarchical location of a folder (directory).

path list The structure, or the corresponding parameter, containing the full path name for a file.

path name A file name specifying all directories leading to the file. See also *full path name* on page 2-97 and *relative path name* on page 2-199.

pattern (1) A regular expression or series of regular expressions that define the search pattern.
(2) In GL, a 16x16, 32x32, or 64x64 array of bits defining the texturing of polygons on the system display.

pattern–action When the **awk** command finds a pattern in an input data file that matches a line in the program file, it performs the associated action on that line.

pattern address
Reference to a line by a string contained within the line, rather than by a numerical or symbolic address. A pattern address can be a character string or a *regular expression*. See also *symbolic address* on page 2-233.

pattern matching
Specifying a pattern of characters that the system should find.

pattern–matching character
Special characters such as * (asterisk) or ? (question mark) that can be used in a file specification to match one or more characters. For example, placing a ? in a file specification means that any character can be in that position. Synonymous with *wildcard*.

pattern strings Strings of regular expressions composed of special pattern–matching characters. The pattern strings can be used in addresses to specify lines and, in some subcommands, portions of a line.

PBX Private Branch Exchange. A private telephone system that performs automatic selection of outside lines.

PC Personal computer.

PCI Programmed Control Interrupt.

PCM Physical Connection Management.

PCS See *programmable character set* on page 2-188.

PDN See *public data network* on page 2-175.

PE Phrase Encoded, a magnetic tape recording format with a density of 1600 bpi.

peak rate The maximum speed at which a device could operate under ideal conditions, if its designer were choosing the workload.

peer–to–peer communications
Pertaining to data communications between two nodes that have equal status in the interchange. Either node can begin the conversation. See also *Logical Unit Type 6.2* on page 2-134.

peer trust A type of trust relationship established between two cells by means of a secret key shared by mutual authentication surrogates maintained by the two cells. A peer trust relationship enables principals in the one cell to communicate securely with principals in the other.

pel See *picture element* on page 2-175.

pending Waiting, as in an operation that is pending.

pending state A condition of a server program in which it has received a request for an action (start, stop, or suspend) but has not yet performed that action.

PEP Packet Exchange Protocol. A datagram service that is implemented by a user–level library, using IDP datagram sockets.

per–process data area
In kernel mode, a portion of the user process stack segment. This area is paged with the process and it contains process information such as the current directory of files opened by the process or input in I/O mode. This information occupies the top of the stack segment. See also *user block* on page 2-253 and *user structure* on page 2-254.

peripheral device
With respect to a particular processing unit, any equipment that can communicate directly with that unit.

peripheral unit See *peripheral device* on page 2-173.

permanence A basic property of transaction processing systems. This term means that once a transaction has committed, the modifications made to data by that transaction must be permanent. Subsequent transactions requesting the data modified by a previous transaction must always see the new data. These changes must be preservable even in the event of a system failure.

permanent error
An error that cannot be eliminated by retrying an operation.

permanent link A connection below a multiplexer that can exist without having an open controlling stream associated with it.

permanent storage
A storage device whose contents cannot be modified.

permanent virtual circuit (PVC)
In X.25 communications, a virtual circuit that has a logical channel permanently assigned to it at each DTE. Call–establishment protocols are not required. Contrast with *switched virtual circuit* on page 2-218. See also *virtual circuit* on page 2-258.

permission The modes of access to a protected object.

permission code
A three–digit octal code or a nine–letter alphabetic code that indicates access permissions. The access permissions are read, write, and run. See also *access permission* on page 2-3.

permission field
One of the three–character fields within the permissions column of a directory list. The permission field indicates the read, write, and run permissions for the file or directory owner, for the group, and for all others.

permissions Codes that determine how the file can be used by any users who work on the system. In Common Desktop Environment, a set of flags that determine a user's access to files and directories, which you can see using the **Properties...** command on the File menu.

persistence In Display PostScript (DPS), a specified character set that is used for all subsequent text segments in a compound string until a new character set is encountered.

persistent data Data which retains its value across multiple runs of transactional applications, regardless of system failures or restarts

persistent segment
A segment whose pages have permanent locations on disk, rather than temporary slots in the paging space.

perspective projection
A technique used to achieve realism when drawing primitives. In a perspective projection, the lines of projection meet at the viewpoint; thus, the size of a primitive varies inversely with its distance from the source projection. The farther a primitive or part of a primitive is from the viewer, the smaller it will be drawn. This effect, known as *perspective foreshortening*, is similar to the effect achieved by photography and by the human visual system. See also *orthographic projection* on page 2-163.

peta Two to the fiftieth power.

PEX A protocol for supporting three–dimensional graphics.

PEXlib A programmer's interface to the PEX protocol.

PFM See *program fault management* on page 2-187.

phase (1) One of several stages of file system checking and repair performed by the **fsck** command.
(2) A distinct part of a process in which related operations are performed.
(3) A part of a sort and merge program, such as sort phase and merge phase.
(4) A part of a data call.

phase modulation
Altering the phase of a carrier signal to convey data signals.

PHIGS See *Programmers' Hierarchical Interactive Graphics System* on page 2-188.

phototypesetter
A typesetting machine that operates by projecting light through film matrices of the type characters upon light–sensitive paper or film.

physical block See *block* on page 2-19.

physical data block
See *block* on page 2-19.

physical device
See *device* on page 2-67.

physical file (1) An indexed file containing data for which one or more alternative indexes have been created.
(2) A database file that describes how data are to be presented or received from a program and how data are actually stored in the database. A physical file contains one record format and one or more members.

physical layer The lowest layer of network design as specified by the ISO Open System Interconnection (OSI) reference model. This layer is responsible for interfacing with the medium, detecting and generating signals on the medium, and converting and processing signals received from the medium and from the data link layer. See also *physical level* on page 2-175.

physical level In X.25 communications, the mechanical, electrical, functional, and procedural media used to activate, maintain, and deactivate the physical link between the data terminal equipment (DTE) and the data circuit–terminating equipment (DCE). Synonymous with *level 1*. See also *level* on page 2-127, *frame level* on page 2-96, *data–link level* on page 2-61, *packet level* on page 2-167, and *physical layer* on page 2-174.

physical network
A network of machines linked by physical network cabling, modems, or other hardware. A physical network can contain one or several logical networks.

physical partition (PP)
The smallest unit of disk–space allocation for a logical volume. The physical partition is contiguous space on a physical volume. A fixed–size portion of a physical volume. One or more physical partitions constitute the underlying physical storage medium for a logical partition.

physical unit (PU)
In SNA, a set of programs that control the actual physical hardware associated with a node.

physical volume (PV)
(1) The portion of a single unit of storage accessible to a single read/write mechanism; for example, a drum, a disk pack, or part of a disk storage module.
(2) A read–write fixed disk physically attached to a computer. The actual storage space provided by a single fixed–disk drive. See also *log volume* on page 2-134.

picking In computer graphics, a method for finding out what primitives are being drawn near the cursor on the display screen. See also *hit* on page 2-108, *picking region* on page 2-175, *selecting* on page 2-213, and *selecting region* on page 2-213.

picking region A rectangular volume around the cursor that is sensitive to picking events. If a drawing primitive draws within this volume, a pick event is reported. The width and height of the region can be set by the user. If the z–buffer is enabled, the depth of the region is the entire z–buffer. See also *hit* on page 2-108, *selecting* on page 2-213, *picking* on page 2-175, and *selecting region* on page 2-213.

picture A pixmap used for displaying Common Desktop Environment icons, background patterns, and controls.

picture element (pel)
(1) In computer graphics, the smallest element of a display space that can be assigned color and intensity independently.
(2) A point in the frame buffer or on the display. See also *pixel* on page 2-176.

PID See *process ID* on page 2-186.

piecewise linear curve
A list of coordinate pairs in the parameter space for the Non–Uniform Rational B–Spline (NURBS) surface. These points are connected with straight lines to form a path.

pin (1) An area of memory reserved for certain functions.
(2) One of the connectors in an adapter plug.

PIO See *programmable input/output operation* on page 2-188.

PIP See *Program Initialization Parameters* on page 2-187.

pipe (1) To direct the data so that the output from one process becomes the input to another process. The standard output of one command can be connected to the standard input of another with the pipe operator (|). Two commands connected in this way constitute a pipeline.
(2) A one–way communication path between a sending process and a receiving process. See also *pipeline* on page 2-176.

pipeline (1) A direct, one–way connection between two or more processes.
(2) A serial arrangement of processors or a serial arrangement of registers within a processor. Each processor or register performs part of a task and passes results to the next processor. Several parts of different tasks can be performed at the same time.
(3) To perform processes in a series.
(4) For increased processing speed, to start the running of an instruction sequence before the previous instruction sequence is completed. See also *pipe* on page 2-176.

pipeline options In GL, variables that control the flow of processing in the graphics pipeline. For instance, lighting is a pipeline option. If lighting is turned on, the color of a primitive is obtained by evaluating the lighting equations. If lighting is turned off, the last color specified is used. Other pipeline options are the back–facing flag, the shade–model flag, the depth–cueing flag, the picking flag, the color–mode (color index or RGB) flag, the z–buffer flag (enables or disables drawing to the z–buffer), and so on. See also *attribute* on page 2-14.

pitch A unit of width of typewriter type, based on the number of times a letter can be set in a linear inch. For example, 10–pitch type has 10 characters per inch.

pixel A rectangular picture element. The smallest element used to compose an image, a single dot. A display screen is composed of an array of pixels. In a black–and–white system, pixels are turned on and off to form images. In a color system, each pixel has three components: red, green, and blue. The intensity of each component can be controlled. See also *picture element* on page 2-175.

pixel map A three–dimensional array of bits. A pixel map can be thought of as a two–dimensional array of pixels, with each pixel being a value from zero to 2 to the N –1, with *N* as the depth of the pixel map. Synonym for *pixmap* on page 2-176.

pixel value In Enhanced X–Windows, the number of bit planes used in a particular window or pixmap. For a window, a *pixel* value indexes a color map and derives an actual color to be displayed. A pixel is an *N* –bit value, where *N* is the number of bit planes (the depth) used in a particular window or pixmap.

pixmap (1) Synonym for *pixel map* on page 2-176.
(2) In AIXwindows and Enhanced X–Windows, a data type to which icons (originally created as bitmaps) are converted. Once this conversion takes place, the appropriate AIXwindows subroutines can generate pixmaps through references to an**.Xdefaults** file (by name) and through an argument list (by pixmap). See also *image cache* on page 2-111.

placeholder An object, component or file that only exists to mark the position of an intended entity.

plaintext The input to an encryption function or the output of a decryption function. Decryption transforms ciphertext into plaintext.

plane When a pixmap or window is thought of as a stack of bitmaps, each bitmap is called a plane or bit plane.

plane mask (1) Determines which of the display adapter storage places are modified by the output functions.
(2) In Enhanced X–Windows, a bit mask restricting graphics operations to affect a subset of bit planes. It is stored in a graphics context. Graphics operations can be restricted to affect only a subset of bit planes of a destination.

platen The support mechanism for paper on a printer, commonly cylindrical, against which printing mechanisms strike to produce an impression.

playing back In Encina, the actions of the Recovery Service when a TP system using recoverable data is restarted. When started, the Recovery Service plays back log records for transactions that have prepared but which were not actually committed, guaranteeing that the state of that recoverable data reflects the records maintained by the TP system.

PList An array of pointers with a suite of operations for adding and removing elements in various ways.

plotter A hard copy device, attached to the system with cables, that prints two–dimensional graphs and charts.

plug A device that connects the wires of an electrical circuit to an electrical source. The plug is designed to be inserted into a jack.

PMF Parameter management frame.

PMP See *Preventive Maintenance Package* on page 2-183.

PMR See *Problem Management Record* on page 2-185.

point (1) A unit of typesetting measure equal to 0.01384 inch (0.3505 mm), or about 1/72 of an inch. There are 12 points per pica.
(2) In CDE, to move the mouse until the pointer rests on a particular screen element or area.

point–handle A point within a graphic object.

point–to–point link
A switched or nonswitched link that connects a single remote link station to a node or to another station.

pointer (1) A variable that holds the address of a data object.
(2) A physical or symbolic identifier of a unique target.
(3) In computer graphics, the device attached to the cursor and tracked on the screen.

pointer grabbing
In Enhanced X–Windows, a client can actively grab control of the pointer so that button and motion events will be sent to that client rather than the client to which the events normally would have been sent. See also *grab* on page 2-102, *button grabbing* on page 2-17, and *key grabbing* on page 2-124.

pointer to member
Used to access the address of nonstatic members of a C++ class.

pointer type A data type that defines variables containing addresses and, sometimes, other information about variables.

pointing Positioning the pointing cursor on a displayed object. The action of lining up the mouse pointer so that the pointer lies on top of something.

pointing device
In Enhanced X–Windows, a device with effective dimensional motion, usually a mouse. One visible cursor is defined by the Core protocol, and it tracks whatever pointing device is attached as the pointer.

polar coordinates
A coordinate system in which positions are measured as a distance from the origin and an angle from some reference direction (usually, counterclockwise from the *x* –axis).

poll (1) In data communications, an interrogation that determines whether a station is ready to transmit information.
(2) To run a polling sequence.

polled I/O devices
Devices (keyboard, mouse, button, dials) whose current values are read by the user process.

polling (1) On a multipoint connection or a point–to–point connection, the process whereby data stations are invited, one at a time, to transmit data.
(2) Interrogation of devices so as to avoid contention, determine operational status, or determine readiness to send or receive data.

polyline In computer graphics, a sequence of adjoining lines.

polymarker In computer graphics, a sequence of markers. The definition of the marker includes specific attributes such as color, style, width, height, pattern, and origin.

polymorphic functions
Functions that can be applied to objects of more than one data type. C++ implements polymorphic functions in two ways: overloaded functions (calls are resolved at compile time); and virtual functions (calls are resolved at run–time).

polymorphism An object–oriented programming feature that may take on different meanings in different systems. Under various definitions of polymorphism, (a) a *method* or *procedure* call can be executed using arguments of a variety of types, or (b) the same variable can assume values of different types at different times, or (c) a method name can denote more than one *method procedure*.

pop A term used when a module that is immediately below the stream head is removed.

pop–down In Enhanced X–Windows, an action referring to a type of widget that closes when a pointer button is released.

pop up (1) In Enhanced X–Windows, a box on the display screen that displays information or asks you to make choices.
(2) In Enhanced X–Windows, an action referring to a type of widget that opens automatically when a pointer button is held down within certain windows.
(3) To use a widget to create a window outside the window hierarchy defined by the widget tree.

pop–up cascade
In Enhanced X–Windows, several spring–loaded pop–ups emanating in succession from one modal pop–up.

pop–up child In Enhanced X–Windows, a child on the pop–up list.

pop–up list A list of pop–up children stored in a widget.

pop–up menu (1) Synonym for *popup* on page 2-178.
(2) In AIXwindows, a type of **MenuPane** widget that appears as the result of some user action (usually clicking a mouse button) and then disappears when the action is completed.
(3) The interface definition for translation actions.
(4) In CDE, a menu that, when requested, is displayed next to the object with which it is associated. Pop–up menus are usually displayed by clicking mouse button 3 or pressing Shift+F10.

pop–up widget In Enhanced X–Windows, a window child of the root that is attached to its widget parent differently than the normal widget; a pop–up widget is not geometrically constrained by its parent widget.

pop–up window
Any window that opens automatically when activated. See also *pop–up* on page 2-178.

popdown In AIXwindows, the manner in which a type of **MenuPane** widget disappears suddenly (pops down) in the display when some user action (usually clicking a mouse button) is completed.

popup In AIXwindows, the manner in which a type of **MenuPane** widget appears suddenly (pops up) in the display as the result of some user action (usually clicking a mouse button). Synonym for *pop–up menu* on page 2-179. See also *pop–up window* on page 2-179.

POR See *power–on reset* on page 2-181.

port (1) A part of the system unit or remote controller to which cables for external devices (display stations, terminals, or printers) are attached. The port is an access point for data entry (input) to or exit from (data output) a computer system.
(2) An entrance to or exit from a network.
(3) To make the programming changes necessary to allow a program that runs on one type of computer to run on another type of computer.
(4) In NCS, a specific communications end point within a host. A port is identified by a port number. See also *socket* on page 2-221 and *listening* on page 2-131.

portability The characteristic that determines whether a source program can be compiled and run on computers of different architectures without requiring recoding.

portable character set
In the XPG4 system interface, the collection of characters present in all locales supported by XSI–conformant systems: Or
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz
0123456789!#%^&*()_+–={}[] :”~;’‘?,.|\/@$

Also included are the alert, backspace, tab, newline, vertical–tab, form–feed, carriage–return, space characters, and the null character, NUL.

portable file name character set
In the XPG4 system interface, the set of characters from which portable file names are constructed. For a file name to be portable across implementations of the XPG4 and ISO POSIX–1 standard, it must consist only of the following characters:
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz
0123456789._–

The last three characters are the period, underscore, and hyphen characters, respectively. The hyphen must not be used as the first character of a portable file name. Uppercase and lowercase letters retain their unique identities between conforming implementations. In the case of a portable path name, the slash character can also be used.

Portable Operating System Interface For Computer Environments (POSIX
An IEEE standard for computer operating systems.

portrait display
A rectangular display that is taller than it is wide. See also *landscape display* on page 2-126.

portrait upside–down
A page orientation such that the top of the printed image is at the trailing edge of the paper as it emerges from the printer.

position (1) Any location in a string that may be occupied by an element and that is identified by a serial number.
(2) The location of a character in a series, as in a record, a displayed message, or a computer printout.

position (within an attribute)
The ordinal position of one value relative to another.

position (within a string)
The ordinal position of one element of a string relative to another.

positional association
In Ada language, specifies the association of an item with a position in a list, by using the same position in the text to specify the item.

positional parameter
(1) A shell facility that assigns values from the command line to variables in a program.
(2) A parameter that must appear in a specified location relative to other positional parameters.

POSIX See *Portable Operating System Interface For Computer Environments* on page 2-180.

post The action required to make a pop–up or pull–down menu appear. This action is normally a click or button press on one of the mouse buttons.

POST See *power–on self–test* on page 2-181.

post processor
A computer program that effects some final computation or organization. In text formatting, a postprocessor command translates the output of the **nroff and troff** commands for use on certain printers, typesetters, or phototypesetters.

posted event A notification sent to the DLC by its attached device handler by way of the **e_post** system call.

PostScript A graphics language used to drive output of text and graphics. Trademark of Adobe Systems, Inc.

pound (lb) Unit of measurement for weight equal to 16 ounces or 454 grams.

power factor The ratio of power consumed to the volt amps (apparent power).

power–on light The light on the operator panel that indicates that the DC power in the system unit is functioning.

power–on reset (POR)
A key sequence that restarts the operating system (or other program) without turning off the electrical power of the system.

power–on self–test (POST)
A series of internal diagnostic tests activated each time the system power is turned on.

power requirement
The actual power consumed by a computer system, measured in watts.

power source The minimum acceptable rating of the electrical circuit providing power to a computer system, measured in volt amps (kVA).

PPA (physical point of attachment)
The point at which a system attaches itself to a physical communications medium.

PPA identifier An identifier of a particular physical medium over which communication occurs.

P(R) In X.25 communications, the packet receive sequence number.

pragma In Ada language, conveys information to the compiler.

precedence (1) The priority system for grouping different types of operators with their operands.
(2) In programming languages, an order relation defining the sequence of the application of operators within an expression.

precision (1) A measure of the ability to distinguish between nearly equal values. See also *single precision* on page 2-220 and *double precision* on page 2-75.
(2) The degree of discrimination with which a quantity is stated. For example, a three–digit numeral discriminates among 1000 possibilities.
(3) In GL, the number of digits that are printed or displayed. (4.) The number of straight line segments used to approximate one segment of a spline.

preconnected file
A unit or file that was defined at installation time. For example, standard input and standard output are preconnected files.

Predefined Connection Object Class
Specifies the kind of connections that can be made to a device and where.

predefined convention
In FORTRAN, the implied type and length specification of a data item based on the initial character of its name, when no explicit specification is given. The initial characters I through N imply type integer of length 4; the initial characters A through H, O through Z, $, and _ imply type real of length 4.

predefined database
Contains configuration data for all possible devices supported by the system. See also *Device Configuration Database* on page 2-67 and *Customized Database* on page 2-29.

Predefined Devices Object Class
Represents each device type, as determined by class, subclass, and type. The Predefined Devices Object Class contains basic information about the devices, such as device method names and how to access the information contained in the other object classes.

predicate Boolean logic term denoting a logical expression that determines the state of some variables. For example, a predicate can be an expression stating that ”variable A must have the value 3.” The control expression used in conjunction with condition variables is based upon a predicate. Use a condition variable to wait for some predicate to become true, for example, to wait for something to be in a queue.

preferential CUG
In X.25 communications, the default closed user group.

prefix In Ada language, used as the first part of certain kinds of name. A prefix is either a function call or a name. See also *name* on page 2-150.

preinstalled Software that is installed by the manufacturer and ready to use.

premultiplication
In GL, matrix multiplication on the left. If a matrix M is premultiplied by a matrix T, the result is TM.

preprocessor (1) A functional unit that effects preparatory computation or organization.
(2) In emulation, a program that converts data from the format of an emulated system to the format accepted by an emulator.
(3) A program that examines the source program for preprocessor statements, which are then run, resulting in the alteration of the source program.
(4) A program that modifies, and possibly optimizes, source programs before they are processed by a compiler.

preprocessor statement
In C language, a statement that begins with the # (pound sign) and contains instructions that the preprocessor interprets.

prerequisite A software product or a service update that must be installed *before* another software product or service update is installed. If you attempt to install software products or service updates without the required prerequisite software, a system message displays the names of required prerequisite software. Contrast with *dependent* on page 2-66.

presentation address
An unambiguous name that is used to identify a set of presentation service access points. Loosely, it is the network address of an OSI service. See also *address* on page 2-6.

Presentation Service Access Point (PSAP)
Address of an OSI communications partner. It addresses an application in a computer.

presentation space
An array that contains the data and attributes associated with a window.

preservation installation
An installation method used when a previous version of BOS is installed on your system and you want to preserve the user data in the root volume group. However, this method overwrites the **/usr**, **/tmp**, **/var**,and **/** (root) file systems, so any user data in these directories is lost. System configuration must be done after doing a preservation installation.

Preventive Maintenance Package (PMP)
A maintenance level update for your system. A PMP includes updates for the Base Operating System (BOS) and for each optional software product that is installed on your system.

primary An irreducible unit of data. For example, a single constant, variable, or array element.

Primary Enterprise Systems Connection Manager
In multiple ESCM environments, the source of ESCM commands.

primary expression
An identifier, parenthesized expression, function call, array element specification, structure member specification, or union member specification.

primary group In concurrent groups, the group that is assigned to the files that you create.

primary language
The primary locale you want your system to use for screen information.

primary navigation article
The general, top–level unit of software documentation.

primary representation
The form in which the service supplies an attribute value to the client.

primary selected text
A text group selected as a primary target or destination, especially text selected within a text field that can be passed to a function. Also, the first block of text specified in a function or statement. See also *primary selection* on page 2-183.

primary selection
In AIXwindows, the text selected in a widget. The primary selection has a value retrieved by the **XmTextGetSelection** function. See also *primary selected text* on page 2-183.

primary slow poll
A technique used by primary link stations to reduce nonproductive polling of a secondary link station.

primary station
(1) On a point–to–point channel, the station that gains control of the channel first. On a multipoint channel, the station controlling communications.
(2) In high level data link control (HLDC), the part of a data station that supports the primary control functions of the data link, generates commands for transmission, and interprets received responses.
(3) In SNA, the station on an SDLC data link that is responsible for control of the data link. There can be only one primary station on a data link. All traffic over the data link is between the primary station and a secondary station.

prime file In Pascal, a file containing precompiled declarations in the internal table format of the Pascal compiler. Prime files are used to initialize the internal tables of the compiler before compilation begins.

primitive A drawing command, such as **arc**, **line**, **circle**, **polygon**, or **charstr**. Such commands are called primitives because they are not made up of smaller parts, and because they are the basic pieces out of which more complex scenes can be composed. Also used to describe the figures created by drawing commands.

Primitive In Enhanced X–Windows, the Primitive class provides the resources and functionality for the low–level widgets that are managed by the manager class. Primitive class widgets cannot have normal child widgets but they can have pop–up child widgets.

primitive coordinates
The space in which a primitive is defined. A convenient point is chosen as the origin and the primitive is defined relative to this point. Synonym for *primitive space*. See also *eye coordinates* on page 2-88, *screen coordinates* on page 2-210, and *world coordinates* on page 2-265.

primitive font A font in which characters are defined as primitives. Like all other primitives, primitive font characters can be scaled and rotated. See also *raster font* on page 2-194 and *font* on page 2-94.

primitive space
Synonym for *primitive coordinates* on page 2-184.

primitive widget
In Enhanced X–Windows, a widget that instantiates its own children of a known class and does not expect external clients to do so. Primitive widgets do not have general geometry management methods. Primitive widgets that instantiate children are responsible for all operations requiring downward traversal below themselves. See also *widget* on page 2-262.

principal identifier
The name used to identify a principal uniquely.

Print Manager In CDE, a software application that shows all the printers on your system.

print queue A file containing a list of the names of files waiting to be printed.

print server In CDE, a host computer to which one or more printers are connected, or the UNIX process that manages those printers.

printer A device externally attached to the system unit, used to print system output on paper.

Printer control In CDE, the Front Panel control used to start the Printer software application. Dropping a file on the control displays a dialog box you can use to print the file to the default printer.

Printer Jobs In CDE, a software application that provides information about jobs on a single printer.

printer session A 3270 Host Connection Program 2.1 and 1.3.3 (HCON) mode of operation during interaction with a host computer that emulates a 3286/87 printer.

printing device Any printer or other device that prints, such as a typewriter–like device or a plotter.

printout Information from the computer produced by a printer.

priority (1) A rank assigned to a task that determines its precedence in receiving system resources, the CPU in particular.
(2) The relative significance of one job to other jobs in competing for allocation of resources. The importance or urgency of a process.

priority number
A number that establishes the relative priority of printer requests.

priority value A number maintained by the scheduler for each process that indicates the priority of that process. The smaller the *priority value* of the process, the higher its *priority*.

privacy A protection level that may be specified in secure RPC communications and that encrypts RPC argument values.

private A private member of a C++ class is only accessible to member functions and friends of that class.

private object (1) In XDS, an OM object created in a workspace using the object management functions.
(2) In XOM, an object that is represented in an unspecified fashion.

private part For Ada programming, see *package* on page 2-166.

private type In Ada language, a type whose structure and set of values are clearly defined, but not directly available to the user of the type. A private type is known only by its discriminants (if any) and by the set of operations defined for it. A private type and its applicable operations are defined in the visible part of a package, or in a generic formal part. Assignment, equality, and inequality are also defined for private types, unless the private type is limited.

privileged instructions
System control instructions that can only run in the processor's privileged, or supervisor, state. Privileged instructions generally manipulate virtual machines or the memory manager and are not used ordinarily by application programmers. See also *privileged state* on page 2-185.

privileged state
A hardware protection state in which the processor can run privileged instructions. Contrast with *unprivileged state* on page 2-252. See also *privileged instructions* on page 2-185.

privileged user A user logged into an account with root user authority.

problem determination
The process of identifying the source of a problem. Often this process identifies programs, equipment, data communications facilities, or user errors as the source of the problem.

problem determination procedure
A prescribed sequence of steps aimed at recovery from, or circumvention of, problem conditions.

Problem Management Record (PMR)
A number assigned by a support center to a reported problem.

problem state (1) One of two virtual machine protection states that run in the unprivileged state of the processor. User–written application programs typically run in the problem state.
(2) A state during which the processing unit cannot run input/output and other privileged instructions.

procedure (1) See *shell procedure* on page 2-218.
(2) In a programming language, a block, with or without formal parameters, that is initiated by means of a procedure call.
(3) The description of the actions taken to solve a problem.
(4) A set of related control statements that cause one or more programs to be performed.
(5) Synonym for *function* on page 2-97.
(6) For Ada programming, see *subprogram* on page 2-230.

procedure address
The location of a particular program procedure in the AIXwindows Toolkit.

process (1) A sequence of actions required to produce a desired result.
(2) An entity receiving a portion of the processor's time for running a program.
(3) An activity within the system that is started by a command, a shell program, or another process. When a program is running, it is called a process.
(4) In a computer system, a unique, finite course of events defined by its purpose or by its effect, achieved under given conditions
(5) Any operation or combination of operations on data.
(6) In the operating system, the current state of a program that is running. This includes a memory image, the program data, variables used, general register values, the status of opened files used, and the current directory. Programs running in a process must be either operating system programs or user programs. See also *job* on page 2-123.

process accounting
An analysis of how each process uses the processing unit, memory, and I/O resources.

process attribute value
In Workload Management, process attribute values include user ID, group ID, and application pathname.

process concurrency
The degree to which a given process has multiple dispatchable threads at all times.

process group Each process in the system is a member of a process group that is identified by a process group ID. This grouping permits the signaling of related groups of processes. A newly created process joins the process group of its creator.

process ID (PID)
A unique number assigned to a process that is running.

process image See *new–process image* on page 2-153.

process lock Allows the calling process to lock or unlock both its text and data segments into memory.

process pacing
See *pacing* on page 2-166.

process table A kernel data structure that contains relevant information about all processes in the system.

processing agent
A thread within an application server that handles remote procedure calls from clients.

processing unit
A functional unit within a computer that is responsible for a certain aspect of processing.

processor affinity
The degree to which a thread is likely to be dispatched to the same physical processor on which it last ran.

product A software product is made up of software packages that are separately installable.

product ID An integer that identifies a vendor's licensed software product; by means of product IDs, the license server distinguishes among products of the same vendor.

product password
A string encoded with information about licenses for a software product. Product passwords are of two types: license passwords and compound passwords.

profile (1) A file containing customized settings for a system or user.
(2) Data describing the significant features of a user, program, or device.
(3) In security, a description of the characteristics of an entity to which access is controlled.
(4) A description of the control available to a particular network operator.
See also *customization profile* on page 2-59 and *mapping* on page 2-141.

program (1) A file containing a set of instructions that conform to a particular programming language syntax.
(2) A sequence of instructions suitable for processing by a computer. Processing can include the use of an assembler, compiler, interpreter, or translator to prepare the program for running, and to run it.
(3) In programming languages, a logical assembly of one or more interrelated modules. In Ada language, a program is composed of a number of compilation units, one of which is a subprogram called the main program. Execution of the program consists of execution of the main program, which may invoke subprograms declared in the other compilation units of the program.
(4) To design, write, and test computer programs.

program assertion
A mathematical statement used in attempts to verify program corrections. In the graphics operating system, the **assert** subroutine tests program assertions.

program counter
A register in the processing unit that guides the computer through the program. Synonym for *instruction address register* on page 2-117.

program fault management (PFM)
A subsystem of NCS that allows a user to set up cleanup routines when an application does not successfully complete.

Program Initialization Parameters (PIP)
Data passed to a program when it starts running. This data modifies the actions taken by that program or the environment in which that program runs.

program level The version, release, modification, and fix levels of a program. See also *fix number* on page 2-92, *modification number* on page 2-141, *release number* on page 2-199, *version* on page 2-256, *background* on page 2-17, and *version number* on page 2-257

program stack Synonym for *invocation stack* on page 2-121.

program temporary fix (PTF)
A temporary solution to, or bypass of, a defect in a current release of a licensed program.

program text The part of a program that is able to be run. See *text* on page 2-239.

program–text segment
A virtual–memory segment that contains the executable instructions of an application program. A program–text segment is identified by the occurrence of an instruction–cache miss in that segment.

program unit A main program or a subprogram. In Ada programming, a program unit is any one of a generic unit, package, subprogram, or task unit. Synonymous with *module* on page 2-147. See also *segment unit* on page 2-213.

programmable character set (PCS)
A geometric text font. Synonymous with *stroke text*. See also *geometric text* on page 2-100.

programmable input/output operation
The transfer of data between the processor and an I/O device or memory address space as part of an I/O instruction. The I/O instruction designates the address of the control logic, the command to be performed and the processor register location into or from which the data is transferred.

programmable terminal
(1) A user workstation that has computational capabilities.
(2) A workstation that can be programmed to performed user–determined functions.

programmatic interface
In AIXwindows, an application created in such a manner that it will operate in a multiclient environment with other applications running concurrently. Clients communicate with the window manager through Xlib calls or libraries built upon Xlib.

Programmers' Hierarchical Interactive Graphics System (PHIGS)
An ANSI and ISO standard. PHIGS defines an application programming interface designed for interactive two–dimensional and three–dimensional graphics applications using retained data structures.

PROM Programmable read–only memory.

prompt A displayed symbol or message that requests information or operator action.

propagation time
The time necessary for a signal to travel from one point to another on a communications line.

property (1) In Enhanced X–Windows, the name, type, data format, and data associated with a window. By using properties, clients and a window manager share information, such as resize hints, program names, and icon formats. It is a general–purpose naming mechanism for clients. The protocol does not interpret properties.
(2) Public information (that is, information available to any client) that is associated with a window.

property list In Enhanced X–Windows, the list of properties that are defined for a particular window

protected A protected member of a C++ class is accessible to member functions and friends of that class, or member functions and friends of classes derived from that class

protected field A displayed field in which a user cannot enter, modify, or erase data.

protection An arrangement for restricting access to or use of all or part of a computer system.

protection level
The degree to which secure network communications are protected.

protocol (1) In SNA and SNA Server, the meaning of, and the sequencing rules for, requests and responses used for managing a network, transferring data, and synchronizing the states of network components.
(2) A set of semantic and syntactic rules that determines the behavior of functional units in achieving communication.
(3) A mutually agreed–upon mechanism for communicating between clients to accomplish certain actions.

protocol boundaries
The set of SNA verbs supported by SNA Services LU6.2.

protocol family A set of related communications protocols; for example, the Department of Defense Internet Protocols. All members of a protocol family use a common addressing mechanism to identify end points. Synonymous with *address family.* See also *socket address* on page 2-221.

protocol port A unique host identifier used by transport protocols to specify a destination within a host.

prototype file The first file in a new file system that contains tokens. These include the name of the bootstrap program, the size of the created file system, and the specifications of the root file.

pruning In GL, eliminating the drawing of parts of the display list because a bounding box test shows that they are not visible. See also *culling* on page 2-57.

P(S) In X.25 communications, the packet send sequence number.

PSAP See *Presentation Service Access Point* on page 2-182.

PSDN (packet–switching data network)
A PSDN is an interconnecting set of switching nodes that enables subscribers to exchange data using a standard protocol and packet–switching technology. Such a network carries messages divided into packets over circuits that are shared by many network users. A single physical line into an office can handle many concurrent connections.

pseudo device A software–based device; for example, a pty device.

Pseudo–PostScript
A graphics language, similar to PostScript, used to drive output of text and graphics.

pseudo terminal (PTY)
A special file in the **/dev** directory that effectively functions as a keyboard and display device to software that uses the Berkeley line discipline. A pseudo terminal consists of a pair of character devices, referred to as the ”master” and ”slave.” The slave device (**/dev/pts**) is manipulated by another process through the master half (**/dev/ptc**) of the pseudo terminal.

pseudocolor In Enhanced X–Windows, a class of color map in which a pixel value indexes the color map entry to produce independent red, green, and blue values. That is, the color map is viewed as an array of triples (RGB values). The RGB values can be changed dynamically. This is mutually exclusive to the direct color color map class.

PSN See *public switched network* on page 2-175.

PSTN See *public switched telephone network* on page 2-190.

PTF See *program temporary fix* on page 2-187.

PTN See *public telephone network* on page 2-190.

PTT Post, Telegraph, and Telephone authority.

PTY See *pseudo terminal* on page 2-189.

PU See *physical unit* on page 2-175.

public A public member of a C++ class is accessible to all functions.

public data network (PDN)
A communications common carrier network providing data communications services over switched or nonswitched lines.

public directory
In BNU, the directory (**/var/spool/uucppublic**) that is open to all BNU users. The public directory is used to transfer files and programs among systems linked by BNU or other versions of the UNIX–to–UNIX Copy Program (UUCP).

public node Any node that does not run Monitor system components or servers. For example, client applications run on public nodes.

public switched network (PSN)
A communications service through which users can be connected by dialing specific service address numbers.

public switched telephone network (PSTN)
A communications common carrier network that provides voice and data communications services over switched lines.

public telephone network (PTN)
A communications common carrier network that provides voice and data communications services over switched or nonswitched lines.

puck A device used to select a particular location on a tablet.

pull installation
In the Network Installation Management environment, an installation that is initiated from a target.

pulldown The manner in which a **MenuPane** widget gives the appearance of being ”pulled down” from a **MenuBar** widget as the result of some user action (usually clicking a mouse button).

pulldown menu
A type of **MenuPane** widget that gives the appearance of being ”pulled down” from a **MenuBar** widget as the result of some user action (usually clicking a mouse button).

pure virtual function
A virtual function is declared pure by replacing the function definition with ’=0;’.

purported name
A construct that is syntactically a name, but that has not yet been shown to be a valid name.

push A term used when a module is inserted in a stream immediately below the stream head.

push button A rounded–corner rectangle with text inside. Push buttons are used in dialog boxes for actions that occur immediately when the push button is selected.

push installation
In the Network Installation Management environment, an installation that is initiated from a machine other that the target.

push permissions
Permissions that enable remote execution of commands.

pushable module
A module between the stream head and the driver. A driver is a non–pushable module and a stream head includes a non–pushable module.

PUT Program update tape.

PUT 2.0 or PUT 2.1

In SNA, a peripheral node that has limited addressing and path control routing capabilities. A PUT 2.0 node depends on subarea nodes (PUT 4 and PUT 5) to translate between its local addressing and network addressing. PUT 2.0 does not support the full capabilities of LU6.2; PUT 2.1 does. SNA Services operates only as either a PUT 2.0 or a PUT 2.1 peripheral node.

PUT 4 or PUT 5

A subarea node that provides network–wide addressing and control data flow within a subarea (the subarea node and all peripheral nodes connected to it). PUT 4 does not contain an SSCP component; PUT 5 does. SNA Services cannot perform the functions of a PUT 4 or a PUT 5 subarea node.

PVC See *permanent virtual circuit* on page 2-173.

Q

Q–bit In X.25 communications, the bit in a data packet that can be set by the sending DTE to qualify the user data in some way that is meaningful to the receiving DTE.

qdaemon The daemon process that maintains a list of outstanding jobs and sends them to the specified device at the appropriate time.

QIC Quarter–inch cartridge.

QID Queue identifier.

QLLC See *qualified logical link control* on page 2-192.

qualified class name
Any class name or class name qualified with one or more :: (scope) operators.

qualified expression
In Ada language, an expression preceded by an indication of its type or subtype. Such qualification is used when, in its absence, the expression might be ambiguous (for example as a consequence of overloading).

qualified logical link control (QLLC)
A data link control protocol that enables SNA–to–SNA communications over an X.25 network.

qualified name (1) A name made unique by the addition of one or more qualifiers.
(2) A data name explicitly accompanied by a specification of the class to which it belongs in a specified classification system.
(3) In C++, used to qualify a nonclass type name such as a member by its class name.

qualified type name
Used to reduce complex class name syntax by using typedefs to represent qualified class names.

qualifier (1) A unique name used to identify another name.
(2) A modifier that makes a name unique.
(3) All names in a qualified name other than the rightmost, which is called the simple name.

qualifier bit See *Q–bit* on page 2-192.

quality of service negotiation
An optional CCITT–specified facility.

quantization The subdivision of the range of values of a variable into a finite number of nonoverlapping, but not necessarily equal intervals. Each interval is represented by an assigned value.

quantum A subrange in quantization.

quark In Enhanced X–Windows, synonym for *string* on page 2-228.

query (1) The action of searching data for desired information.
(2) In data communications, the process by which a master station asks a slave station to identify itself and to give its status.
(3) In interactive systems, an operation at a workstation that elicits a response from the system.
(4) A request for information from a file based on specific conditions.

queue (1) A line or list formed by items waiting to be processed.
(2) To form or arrange in a queue.

queue device A logical device defining characteristics of a physical device attached to a queue.

queue element A block of data or an item in a queue.

queue stanza Defines a queue for one or more devices to which jobs can be queued.

queued I/O devices
Devices (keyboard, mouse, button, dials) whose changes are recorded in the event queue.

quiet system A system on which no processes are running other than the process whose performance is being timed, and from which all communications devices have been disconnected.

quit A key, command, or action that tells the system to return to a previous state or stop a process.

quote To mask the special meaning of certain characters, causing the characters to be taken literally.

QWERTY keyboard
A keyboard having the standard keyboard layout. The term is derived from the sequence of the first six keys in the first row of alphabetic keys. See also *AZERTY keyboard* on page 2-16.

R

race condition For the **signal** subroutine, the condition in which the signal occurs while the signal action is set to the **SIG_DFL** value, and the signal–catching function has not yet established itself as the catcher for this signal.

radio box Special configuration of toggle buttons within a row column manager widget.

radio button Indicates a fixed set of choices. Only one of the buttons in the set can be selected at a time. A circle with text alongside, the circle is partially filled when a choice is selected.

radio frequency interference
A signal (noise) that is radiated or conducted from one electronic circuit or device the may cause incorrect operation interference in another electronic circuit or device. An electronic circuit or device can be an emitter of electromagnetic interference (EMI) or susceptible to electromagnetic interference.

raise To make the stacking order of a window higher.

raising an exception
For Ada programming, see *exception* on page 2-84.

RAM Random access memory. Specifically, the memory used for system memory. Sometimes this memory is referred to as main storage.

random access
An access mode in which records can be read from, written to, or removed from a file in any order.

random file access
Location of a record that matches a specific index key value. Random access of the records in a file requires that the file have a unique index and that each record have a unique index key value.

range In Ada language, a contiguous set of values of a scalar type. A range is specified by giving the lower and upper bounds for the values. A value in the range is said to belong to the range.

range constraint
In Ada language, a range constraint of a type specifies a range, and thereby determines the subset of the values of the type that belong to the range.

range of a DO loop
Those statements that physically follow a DO statement, up to and including the final statement specified by the DO statement.

range of records
Multiple records to be processed sequentially. A range of records is selected by specifying key values that bound the records to be selected, or by specifying an individual key value for which all matching records should be selected in a nonunique index.

raster The area of the video display that is covered by sweeping the electron beam of the display horizontally and vertically. Normally the electronics of the display sweep each line horizontally from top to bottom and return to the top during the vertical retrace interval. See also *frame buffer* on page 2-95.

raster font A font in which the characters are defined directly by the raster bit map. See also *font* on page 2-94 and *primitive font* on page 2-184.

raster graphics
Computer graphics in which a display image is composed of an array of pixels arranged in rows and columns.

raster mosaics An area in annotated text fonts that contains a definition for each character in the font.

raster subsystem
That part of the system concerned with an image after it has been transformed and scaled to screen coordinates. It includes scan conversion and display.

raw device A device that treats data I/O as a continuous stream, without consideration for the data's logical structure. For example, I/O for fixed disks and streaming tapes occurs in units of bytes that have no relationship to characters.

raw I/O Character–oriented access to a block device not utilizing in–core buffers.

raw tape device
See *raw device* on page 2-195.

RC Routing control.

RCM (1) See *real–time control microcode* on page 2-196.
(2) Rendering Context Manager.

RCS See *Revision Control System* on page 2-205.

RD Receive Data used with EIA–232 protocol.

read access An access right that grants the ability to view CDS data.

read–only file Pertaining to file system mounting, a condition that allows data to be read but not copied, printed, or modified.

Read Only Storage (ROS) menus
The menus displayed by system ROS on a target. Information used in booting the target is provided by the user to the ROS menus.

read queue A message queue in a module or driver containing messages moving upstream. Associated with input from a driver.

real constant A string of decimal digits that expresses a real number. A real constant must contain either a decimal point, a decimal exponent, or both.

real device The actual device hardware.

real group ID For each user, the group ID defined in the password file.

real memory The active physical memory on any system. Contrast with *virtual memory* on page 2-258.

real number (1) A number that contains a decimal point and is stored in fixed–point or floating–point format.
(2) A number that can be represented by a finite or infinite numeral in a fixed–radix numeration system.

real operation Any operation on Ada floating–point or fixed–point values.

real page number
This field in the translation control word (TWC) contains the real page address that the bus address is mapped to in system memory.

real storage Storage directly accessible to the processor from which instructions can be run and from which instructions can fetch data.

real time (1) The time elapsed between the loading of a program and its completion. (2) Pertaining to the processing of data by a computer in connection with another process outside the computer, according to time requirements imposed by the outside process. (3) Used to describe systems that operate in conversational mode and processes that can be influenced by human intervention while in progress. (4) Pertaining to an application, such as a process control system or a computer–assisted instruction system, in which response to input is fast enough to affect subsequent input.

real–time control microcode (RCM)
In X.25 communications, the microcode that runs on the X.25 Interface Co-Processor/2 to provide control functions.

real–time system
A system that receives and processes data so the data or result is available for immediate use.

real type (1) An arithmetic data type that can approximate the values of real numbers.
(2) In Ada language, a real type is a type whose values represent approximations to the real numbers. There are two kinds of real type: fixed point types are specified by absolute error bound; floating point types are specified by a relative error bound expressed as a number of significant decimal digits.

real user ID (1) Identifies the user who was authenticated the last time the **login** command or **su** command was used.
(2) For each user, the user ID that is specified in the **/etc/passwd** file.

realm A cell, considered exclusively from the point of view of security; this term is used in Kerberos specifications.

rebinding Reestablishing a communications channel for making remote procedure calls after that channel has been closed.

reboot To reinitialize the execution of a program by repeating the initial program load (IPL) operation.

REC See *Lock Service* on page 2-133.

receive In X.25 communications, to take an incoming packet (such as an incoming–call packet or a data packet) from the buffer.

receive pacing In SNA, the pacing of message units that a component is receiving. Contrast with *send pacing* on page 2-214. See also *pacing* on page 2-166.

receive time out
In data communications, a condition that occurs when no data is received in a given period of time.

recognized private operating agency (RPOA)
A private X.25 network that can optionally be selected by the user at call setup time to carry the X.25 traffic.

Recommendation X.25
See *X.25* on page 2-266.

record (1) In programming languages, an aggregate that consists of data objects, possibly with different attributes, that usually have identifiers attached to them.
(2) A set of data treated as a unit. (3.) A collection of fields treated as a unit. See also *class record* on page 2-37, *widget record* on page 2-262, and *instance record* on page 2-116.

record lock A lock that prevents some or all of a file from being written to or read. See also *lock* on page 2-133.

record name A data name for a record described in a record description entry.

record type (1) The classification of records in a file.
(2) In Ada language, a value of a record type consists of components are usually of different types or subtypes. For each component of a record value or record object, the definition of the record type specifies an identifier that uniquely determines the component within the record.

recording instrument
In Performance Toolbox, an instrument with the ability to show the statistics for a system resource over a period of time. Recording instruments typically have a time scale with the current time to the right. The values plotted are moved to the left as new readings are received. Types of graphs used to plot these recordings include line, area, skyline, and bar graphs. Contrast with *state instrument* on page 2-225.

recoverable data
Data whose values persist across system shutdowns and failures. Changes made to recoverable data are permanent regardless of system problems. Logging changes to recoverable data is the most common method used to ensure permanence. The changes to that data recorded in the log can always be replayed to bring that data to a valid state.

recovery procedure
(1) An action performed by the operator when an error message appears on the display screen. This action usually permits the program to continue or permits the operator to run the next job.
(2) The method of returning the system to the point where a major system error occurred and running the recent critical jobs again.
(3) A process in which a specified data station attempts to resolve conflicting erroneous conditions arising during the transfer of data.

rectangle A rectangle specified by `[x,y,w,h]` has an infinitely thin outline path with corners at `[x,y][x+w,y][x+w,y+h]` and `[x,y+h]`. In XGSL, when a rectangle is filled, the lower–right edges are not drawn. For example, if *w* = *h* = *0*, nothing would be drawn; if *w* = *h* = *1*, a single pixel would be drawn.

recurring attribute
An attribute with several attribute values.

recursion (1) Using a function to define itself.
(2) Performing an operation in several steps, with each step using the output of the preceding step.

redirect To divert data from a process to a file or device to which it would not normally go.

redirecting control
Transferring an operation to a specified client. Used when window managers or client programs enforce window layout policy to prevent attempts to change the size or position of a window.

Reduced Instruction Set Computer (RISC)
A class of computer designs that uses a relatively small set of frequently used instructions that execute in one cycle.

reentrant service
A service that is safe to call from multiple threads in parallel. If a service is reentrant, there is no burden placed on calling routines to serialize their access or take other explicit precautions. See also *thread–serial service* on page 2-240, and *thread–synchronous service* on page 2-240.

refer A bibliographic preprocessor for the **nroff** and **troff** commands. ”Refer” is designed for literature citations, and it supports data entry, indexing, sorting, retrieval, printing, citations, and either footnote or endnote numbering.

reference bit A bit in each page frame table entry that denotes that the corresponding page has been accessed (either read from or written to) since the last time the operating system cleared the page.

reference count
In an i–node, a record of the total number of directory entries that refer to the i–node.

reference monitor
Code that controls access to an object.

reference widget
The parent widget that searched for children satisfied the search criteria.

refresh rate The rate at which the monitor is refreshed. A 60 Hz monitor is redrawn 60 times per second. Synonymous with *vertical retrace*.

region An area within a bitmap, a pixmap, a screen, or a window.

Region An arbitrary set of pixel locations. The **Xlib** library provides subroutines for manipulating regions. The opaque type **Region** is defined in the **<X11/Xutil.h>** header file.

register (1) A storage device having a specified storage capacity such as a bit, byte, or computer word, and that is usually intended for a special purpose. See also *general purpose register* on page 2-99.
(2) In NCS, to make an interface known to the RPC run time library and, thereby, available to clients through the RPC mechanism. The **rpc_$register** call registers an interface.
(3) In NCS, to enter an object and its location in the Location Broker database. The **lb_$register** call registers an object with the Location Broker. A program can use Location Broker lookup calls to determine the location of a registered object.

registration The creation of an object in the network installation database that uniquely identifies a client, network, or resource in the network installation environment to the master server.

Registry database
A database of information about persons, groups, organizations, and accounts.

regular expression
(1) A set of characters, meta characters, and operators that define a string or group of strings in a search pattern.
(2) A string containing wildcard characters and operations that define a set of one or more possible strings. Contrast with *literal string* on page 2-131. See also *collating element* on page 2-40 and *subpattern* on page 2-230.

reject. To cause portions of applied updates from becoming permanent parts of the product, based on the results of a test period. When you reject an applied service update, the update’s files are deleted and the software vital product data (SWVPD) information is changed to indicate that the update is no longer on the system. The previous version of the software, if there is one, is restored and becomes the active version of the software. Contrast with *apply* on page 2-10 and *commit* on page 2-43.

relational expression
(1) A logical statement that describes the relationship (such as greater than or equal) of two arithmetic expressions or data items.
(2) An expression that consists of an arithmetic expression followed by a relational operator, followed by another arithmetic expression, and that can be reduced to a value that is true or false.

relational operator
(1) The reserved words or symbols used to express a relational condition or a relational expression.
(2) An operator that compares two operands and yields a Boolean value.

relational transformation
A transformation that changes an arithmetic relational expression into the negation of its logical complement. For example, a relational transformation of A>B is NOT(A<=B).

relative address
(1) An address specified relative to a base address.
(2) An address counted relative to a symbol. When a program is relocated, the addresses themselves change, but the relative addresses remain the same.

relative directory
A directory whose name begins with a ./ (dot and a slash).

Relative Distinguished Name (RDN)
A set of Attribute Value Assertions (AVAs), each of which is true, concerning the distinguished values of a particular entry.

relative drawing commands
In computer graphics, commands that draw relative to the current graphics position as opposed to being drawn at absolute locations.

relative file A file organized as an array of fixed–length slots. Records can be inserted in the first free slot found from the beginning or end of the file, or can be explicitly positioned in a certain slot in the file, identified by its relative slot number (RSN).

relative path name
The name of a directory or file expressed as a sequence of directories followed by a file name, beginning from the current directory. Relative path names do not begin with a / (slash) but are relative to the current directory. See also *path name* on page 2-172 and *full path name* on page 2-97.

relative record number
A number that expresses the location of a record in relation to a base position in the file containing it.

relative slot number (RSN)
The number of the slot occupied by a record in a relative file. The RSN is part of the user's data record. Each record in a relative file contains a field that holds its RSN.

relative time A discrete time interval that is usually added to or subtracted from an absolute time.

release number
The release level of a program, which is an indicator of changes to the external programming interface of the program. The version, release, modification, and fix levels together comprise the program level or version of a program. See also *fix number* on page 2-92, *modification number* on page 2-141, *version number* on page 2-257, *program level* on page 2-187 and *version* on page 2-256.

release update package
A set of selective enhancements, filesets, and new versions of optional software products available since the last release of the operating system. Also included is a maintenance level package for each software package.

reliable stream delivery
A type of packet delivery that allows an application program on one machine to connect to an application program on another machine. The stream actually contains many packets of data that are sent one at a time to the receiving machine.

relocatable (1) A value, expression, or address that does not have to be changed when the program is relocated.
(2) Attribute of a set of codes whose address constants can be altered to make up for a change in origin.

remote Pertaining to a system or device that is accessed through a communications line. Contrast with *local* on page 2-132.

remote access data processing
Synonym for *teleprocessing* on page 2-238.

remote connection
A communications link between the local system and a remote system or device.

remote host Any host on the network except the one at which a particular operator is working. Synonymous with *foreign host*.

remote job entry (RJE)
Submission of a job through an input unit that has access to a computer through a data link.

remote login Initiating a session on a system that is accessed through a communications line.

remote pacing Pacing generated by the remote system attempting to control the output of the local system. See also *pacing* on page 2-166.

remote print Issuing print jobs to one machine (client) to print on another machine (server) on a network.

remote procedure call (RPC)
The calling of a remote operation between processes on different hosts or on the same host.

Remote Procedure Call run–time library
The set of **rpc_$** system calls that NCS provides to implement a remote procedure call mechanism.

Remote Statistics Interface (RSi)
In Performance Toolbox, the Manager API which allows an application program to access statistics from remote nodes (or the local host) through a network interface.

remote system A system that is connected to your system through a communication line.

remote transaction program name (RTPN)
The name of a transaction program at the other (remote) end of a conversation.

removable storage device
Any storage device defined during system configuration to be an optional part of the system DASD. A removable storage device can be removed from the system anytime during normal operation.

remove For a software option, the deletion of the option and all of its applied or committed updates from the system. The software vital product data (SWVPD) information is changed to indicate that the option has been removed from the system. Depending on the option, system configuration information is also cleaned up, although this is not always complete. If a previous version, release, or level of the option is on the system, the system will not restore the previous version. Only an option with its updates can be removed. Updates cannot be removed by themselves. Contrast with *commit* on page 2-43.

renaming declaration
In Ada language, declares another name for an entity.

render To create an image on a visual display from data that describes the scene.

rendezvous In Ada language, the interaction that occurs between two parallel tasks when one task has called an entry of the other task, and a corresponding accept statement is being executed by the other task on behalf of the calling task.

repage fault A page fault on a virtual–memory page that is known to have been read from disk ”recently.”

reparent When a widget is moved between interface windows, it is reassigned to the parent widget in the new location, and automatically acquires the inheritable attributes of the new parent.

repeat factor In GL, the magnification with which the linestyle pattern is used.

repetitive tiling operation
In XGSL, an operation that consists of repeatedly copying a 16x16–pixel tile rectangle (pointed to by the tile pixel map data address) to fill a rectangle area of a size specified by the H and W parameters of this call. The format defined in the flags field of the tile pixel map structure defines the format of the tile data.

reply (1) A response to an inquiry.
(2) In SNA, a request unit sent only in reaction to a received request unit. For example, Quiesce Complete is the reply sent after receipt of Quiesce At End of Chain.
(3) In Enhanced X–Windows, the way information requested by a client program is sent back to the client. Both events and replies are multiplexed on the same connection. Most requests do not generate replies; some generate multiple replies.

representation clause
In Ada language, a clause that directs the compiler in the selection of the mapping of a type, an object, or a task onto features of the underlying machine that executes a program. In some cases, representation clauses completely specify the mapping; in other cases, they provide criteria for choosing a mapping.

request (1) A directive, by means of a basic transmission unit, from an access method that causes the network control program to perform a data–transfer operation or auxiliary operation.
(2) In SNA, a message unit that signals initiation of an action or protocol.
(3) In Enhanced X–Windows, a command to the server to send a single block of data over a connection.
(4) In text formatting, a request is a built–in command recognized by the formatters.

requester A display station or interactive communications session that requests a program to be run.

required list In Enhanced X–Windows, an ordered list containing a subset of the installed color maps.

required parameter
A parameter having no value automatically supplied. The user must provide a value.

required value Synonym for *required parameter* on page 2-202.

requisite A software product or a service update that must be installed with another software product or service update. If you attempt to install software products or service updates without the required requisite software, a system message displays the names of required requisite software.

reserved character
A character or symbol that has a special (non–literal) meaning unless quoted.

reserved word A word that is defined in a programming language for a special purpose, and that must not appear as a user–declared identifier.

reset (1) To cause a counter to take the state corresponding to a specified initial number.
(2) To put all or part of a data processing device back to a prescribed state.
(3) On a virtual circuit, reinitialization of data flow control.
(4) To return a device or circuit to a clear state.
(5) In X.25 communications, to reinitialize the flow of control on a virtual circuit, which eliminates all data that may be in transit for the virtual circuit at the time of resetting.

reset cause See *cause code* on page 2-32.

reset collision See *collision* on page 2-40.

reset–confirmation packet
In X.25 communications, a packet transmitted by the data terminal equipment (DTE) to inform the data circuit–terminating equipment (DCE) that a reset operation has been processed. See also *packet* on page 2-166.

reset diagnostic
See *diagnostic code* on page 2-68.

reset packet A packet used to reset a virtual circuit at the interface between the data terminal equipment (DTE) and the data circuit–terminating equipment.

reset–request packet
In X.25 communications, a packet used for the resetting of a virtual circuit at the DTE/DCE interface. See also *packet* on page 2-166.

resize border In CDE, a control used to change the size of a window or a pane in a window.

resolution The process or capability of making the individual parts of objects on screen distinguishable, especially regarding font generation. The number of dots that can be displayed on the screen. Used to determine the clarity of a graphic image.

resolver routine
A kernel process used to resolve symbolic host names into Internet addresses. The method the routine uses to resolve names depends on whether the local network is organized as a flat network or as a hierarchical network.

resource (1) In NIM, any file, directory, file system, or device that is required to perform a NIM operation.
(2) In Enhanced X–Windows, items such as windows, pixmaps, cursors, fonts, graphics contexts, and color maps are known as resources. Each has a unique identifier associated with it for naming purposes. The lifetime of a resource is bounded by the lifetime of the connection over which the resource was created.
(3) In Enhanced X–Windows, a named piece of data in a widget that can be set by a client, by an application, or by user defaults.
(4) In AIXwindows and Enhanced X–Windows, a unique characteristic of appearance or behavior that is associated with one specific class of graphical object. Resources can be passed downward (inherited) by each subclass that is downstream from a higher–level class in the class hierarchy.

resource database
The database assembled for a particular combination of display, host, and application. It can contain information from several sources.

resource identifier (rid)
In Enhanced X–Windows, an integer returned to an application program that identifies a resource that has been allocated for the program's use.

resource limitation values
A set of values that Workload Management attempts to maintain for a set of resource utilization values. The resource limitation values are independent of the resource limits specified with **setrlimit()**.

resource manager
A component that manages application data. Resource managers communicate with application servers.

resource object
A resource used in the Network Installation Management environment that represents a file or directory.

resource outage
The inability to obtain the necessary resources, such as links, buffers, or control block storage.

resource pair See *resource value* on page 2-203.

resource state A state that indicates that the resource is either available or unavailable for use.

resource string
A parameter, such as a part of the program name, that identifies an application's resources.

resource target share
The shares of a resource that should be available to a Workload Management class. These shares are used with other class shares to determine the desired distribution of the resources between classes.

resource utilization value
The amount of a resource that a process is currently using in a system. See also *scope of resource collection* on page 2-210.

resource value Resources determine a widget or window's operation and attributes, such as color and behavior. The resource is associated with a value list that gives the specific values for each resource. See also *attribute* on page 2-14.

resource value string
A parameter containing the default value to be used if a specific resource has not been set.

response (1) An answer to an inquiry.
(2) In SNA, a message unit that acknowledges receipt of a request; a response consists of a response header (RH), a response unit (RU), or both.

response time (1) The time it takes for a data communications system to respond to a request. For example, if you enter a customer number on a workstation keyboard, response time begins when you press the last key and ends when the first character of your answer is displayed at the workstation.
(2) The time from the initiation of an operation until its initiator has enough information to proceed.

restart–confirmation packet
In X.25 communications, a call supervision packet transmitted by a DCE to confirm that the link has been restarted. See also *packet* on page 2-166.

restart–indication packet
In X.25 communications, a call supervision packet transmitted by a DCE to indicate to a DTE that a restart–request has been received.

restart–request packet
In X.25 communications, a call supervision packet transmitted by a DTE to request that a link be restarted.

restore To return to an original value or image; for example, to restore a library from diskette.

restricted response
In X.25 communications, when restricted response is specified for a fast–select call, the call must be cleared; it may *not* be accepted.

restricted shell Shell providing controlled, limited access to specified users.

result An entity produced by an operation.

retransmit To repeat the transmission of a message or segment of a message.

retrieval time The time interval required to locate data in storage and read it for subsequent processing.

retry To try the operation that caused the device error message again.

return code (1) A value that is returned to a program to indicate the results of an operation issued by that program.
(2) A code used to influence the running of succeeding instructions. Synonymous with *return value*.

return statement
A programming language control statement that contains the word return followed by an optional expression and a semicolon.

return value Synonym for *return code* on page 2-204.

reverse charging
In X.25 communications, an optional facility that allows a DTE to request that the cost of a call it makes be charged to the called DTE. See also *optional facilities* on page 2-162.

reverse video A form of highlighting a character, field, or cursor by reversing the color of the character, field, or cursor with its background. For example, changing a red character on a black background to a black character on a red background.

Revision Control System (RCS)
Manages multiple revisions of text files. Designed to control frequently revised text, such as programs, form letters, and papers. It features automatic identification, storage, logging, retrieval, and merging of file revisions. See also *Source Code Control System* on page 2-222.

revision text See *version identifier* on page 2-257.

rewind To select an earlier item, rather than the next or current item, from an argument list.

RFC Request for comments.

RFI See *radio frequency interference* on page 2-194.

RGB color A color value scale that is composed of the primary values (red, green, and blue).

RGB mode A configuration of the hardware that allows values stored in the frame buffer to be interpreted as packed RGB values. The values found in the frame buffer are passed directly to the red, green, and blue guns of the display monitor. The values are not passed through the color map first. (However, each color is sent individually through the gamma ramp to make a final correction to its intensity.) See also *color map mode* on page 2-41.

RGB signal Red/green/blue signal.

RGB value The set of red, green, and blue intensities that compose a color is that color's *RGB* value..

RGBA value In GL, the set of red, green, blue, and alpha intensities that compose a color. Alpha values are available only on machines having alpha bitplanes.

RIC Realtime interface coprocessor.

rid See *resource identifier* on page 2-203.

right–hand rule
If the right hand is wrapped around the axis of rotation, the fingers curl in the same direction as positive rotation, and the thumb points in the same direction as the axis of rotation. A right–handed rotation is counter–clockwise.

right margin The area between the rightmost text character and the right edge of the display or paper.

ring (1) A method used to distribute data in a LAN.
(2) In FDDI, two or more stations connected by a physical medium wherein information is passed sequentially between active stations, each station is turn examining or copying and repeating the information, finally returning it to the originating station. See also *ring network* on page 2-205.

ring buffer An application–defined buffer in which monitor mode input is placed. An application places data from input devices in the buffer. The ring buffer mechanism dramatically shortens the input data path from the input device to the application.

ring network (1) A network in which every node has two branches connected to it.
(2) A network configuration in which devices are connected by unidirectional transmission links to form a closed path. See also *ring* on page 2-205.

RIP See *Routing Information Protocol* on page 2-207.

RISC See *Reduced Instruction Set Computer* on page 2-197.

RJE See *remote job entry* on page 2-200.

RMT Ring management.

RNR frame In X.25 communications, a receive–not–ready frame. Contrast with *RR frame* on page 2-195.

RNR packet In X.25 communications, a receive–not–ready packet. Contrast with *RR packet* on page 2-195.

rolled back Undoing any modifications performed on behalf of a transaction which does not complete (is *aborted*). Any changes made by a transaction which is aborted, for whatever reason, must be undone. Once a transaction is rolled back, no evidence that the transaction was ever attempted remains outside of records in the transaction processing system's log. See also *abort* on page 2-3.

roller ball The sphere inside a mechanical mouse that contacts a desktop or other hard surface.

ROM Read–only memory.

Roman font The ordinary type style. In many typefaces, this is the default font, governing most text. It most often is used to turn off italics or boldface.

Roman numerals
Numbers formed from traditional combinations of Roman letters, either uppercase (I, II, III, IV, and so on) or lowercase (i, ii, iii, iv, and so on). See also *Arabic numerals* on page 2-11.

root (1) The user name for the system user with the most authority.
(2) In Enhanced X–Windows, (a.) The screen on which the window is created. The root of a pixmap or GContext is the same as the root of the drawable used when the pixmap or GContext was created. The root of a pixmap or graphics context is the same as the root of whatever drawable was used when the pixmap or graphics context was created. The root of a window is the root window under which the window was created. (b.) On the widget tree, the Shell widget returned by the **XtInitialize** or **XtCreateApplicationShell** subroutine.
(3) In AIXwindows, the **Shell** widget is the root of the widget tree hierarchy that is maintained within any given application interface. See also *widget tree* on page 2-262.

root device The device on which the root file system resides.

root directory The directory (*/*) that contains all other directories in the system.

root file system
The basic file system, onto which all other file systems can be mounted. The root file system contains the operating system files that get the rest of the system running.

root segment (1) In an overlay operation, the part of a program that must remain in main storage when other overlay segments are run; the first segment of a program with overlays. The root segment remains in main storage at all times while the program is being run.
(2) In a hierarchical database, the highest segment in the tree structure.

root user The person who has unrestricted ability to access and modify any part of the operating system, usually the user who manages the system. See also *root user authority* on page 2-206.

root user authority
The unrestricted ability to access and modify any part of the operating system, usually associated with the user who manages the system. See also *root user* on page 2-206.

root volume group (rootvg)
A volume group containing the Base Operating System (BOS). See also *migration installation* on page 2-145.

root window In a graphics environment, each screen has a root window covering it. This window cannot be reconfigured or unmapped, but otherwise performs like any other window. A root window has no parent.

ROS Remote Operation Service.

ROSE Remote Operation Service Elements.

rotation In computer graphics, the transformation of a primitive by rotating it about an axis. See also *transformation* on page 2-245.

route A path defined for sending data across a network.

routine A set of statements in a program causing the system to perform an operation or a series of related operations. See also *macro* on page 2-139 and *subroutine* on page 2-231.

routing (1) The assignment of the path by which a message will reach its destination.
(2) In SNA, the forwarding of a message unit along a particular path through a network as determined by parameters carried in the message unit, such as the destination network address in a transmission header.
(3) In X.25 communications, the process by which a packet gets to the intended user.

Routing Information Protocol (RIP)
A variant of the XeroxNS Routing Information Protocol, used to maintain current kernel routing table entries.

routing list The list that associates user names with network user addresses and other information, for the purpose of directing incoming X.25 calls.

routing table The table holding a list of valid paths through which hosts can communicate with other hosts. The routing table can hold static routes and dynamic routes.

row A horizontal arrangement of characters or other expressions.

row column manager
A widget that contains toggle buttons or push buttons. It updates its property that accounts for menu history, which allows it to determine which toggle button was selected last.

row–major order
A way of storing array elements such that the rightmost subscript varies most rapidly as memory–adjacent elements are accessed.

RPC See *remote procedure call* on page 2-200.

rpm Revolutions per minute.

RPN See *real page number* on page 2-195.

RPOA See *recognized private operating agency* on page 2-196.

RPQ Request for Price Quotation.

RR (1) Resource Record (DNS).
(2) Round Robin (scheduling).

RR frame In X.25 communications, a receive–ready frame. Contrast with *RNR frame* on page 2-206.

RR packet n X.25 communications, a packet used by a DTE or by a DCE to indicate that it is ready to receive data packets within the window. Contrast with *RNR packet* on page 2-206.

RS–232C See *EIA–232D* on page 2-78.

RS–422A See *EIA–422A* on page 2-79.

RSi See *Remote Statistics Interface* on page 2-200.

RTI Response type indicator.

RTPN See *remote transaction program name* on page 2-200.

RTS Ready to Send used with EIA–232 Protocol.

rubber–band outline
A window with a moveable outline.

rule file A file containing rules determining the appearance and behavior of Common Desktop Environment.

run (1) A performance of one or more jobs or programs.
(2) To cause a program, utility, or other machine function to be performed.

run file The output of the linkage editor. A program file in a format that is suitable for being loaded into main storage and run. See also *module* on page 2-147.

run–time algorithm selection
An optimization technique in which the parameters to an operation are evaluated to determine which of several equivalent algorithms will perform the operation most efficiently.

run–time checks
Error checking that occurs while an Ada program runs. If an error is detected, the program raises an exception.

run–time environment
A collection of subroutines and shell variables that provide commonly used functions and information for system components.

run–time monitor
The part of the Monitor that manages transactional client and server applications that take input from users and service requests.

running state The condition of a machine when users can login and use the machine.

RW lock Abbreviation for *read shared/write exclusive lock*. Any number of threads can hold the lock simultaneously for reading, but if a thread holds the lock for writing, all other threads are excluded from reading or writing the locked resource.

S

S–connector The type of connector used with super–video. Similar to the Y/C connector. It is a 4–pin–min din–type connector.

S–video Super–video. Used to improve the quality of a video image, a technique that maintains separate Y/C signals before recording on magnetic tape or displaying on a monitor.

SABM See *set asynchronous balanced mode* on page 2-217

SAK See *secure attention key* on page 2-212.

sample rate Synonym for *locator sample rate* on page 2-133.

SAP See *service access point* on page 2-215.

SAS Single–attachment station. A station that connects only to the primary ring by way of a wiring concentrator or connects to one other SAS in a back–to–back wiring configuration.

sash In CDE, a box on a separator or split bar that enables you to increase or decrease the size of a window pane using the mouse. You can navigate to the sash using the keyboard.

satisfy For Ada programming, see *constraint* on page 2-51 and *subtype* on page 2-231.

saved user ID The user ID that is acquired when running a setuid program. The saved user ID is the same as the owner of the file that ran. If the file that ran was not setuid, the saved user ID is set to the effective user ID of the parent.

saveset A list of window clients that should not be destroyed when a connection is closed and should be remapped or unmapped. Usually used by window managers to avoid lost windows if the manager is ended abnormally.

scalability The ability of a workload to benefit from a multiprocessor environment.

scalar An arithmetic object or enumerated object.

scalar type A type that defines a variable containing a single value at run time. Contrast with *structured type* on page 2-229. In Ada programming, an object or value of a scalar type does not have components. A scalar type is either a discrete type or a real type. The values of a scalar type are ordered.

scale Increments of measure used by the **nroff** and **troff** commands. All supported scales are converted for the typesetter into a scale called machine units (u).

scale factor (1) A number indicating the position of the decimal point in a real number.
(2) A number used as a multiplier in scaling.

scaling (1) In computer graphics, enlarging or reducing all or part of a display image by multiplying the coordinates of the image by a constant value.
(2) In programming, indicating the number of digit positions in object code to be occupied by the fractional portion of a fixed–point or floating–point constant.
(3) In GL, uniform stretching of a primitive along an axis.

scaling factor The throughput of a workload on a multiprocessor divided by the throughput of that workload on a comparable uniprocessor (*not* on a single–processor SMP system).

scan (1) To examine sequentially, part by part.
(2) To search records for a specified character string or syntax error.

scan code Raw input from the keyboard.

scan conversion
The process of generating pixel information into the frame buffer from an application program.

scanline (1) A list of pixel or bit values viewed as a horizontal row (all values have the same *y* coordinate). The values are ordered by increasing the *x* coordinate. As part of an image, the next scanline is ordered by increasing the *y* coordinate.
(2) A visible line produced on a display by one horizontal sweep of the electron beam of a cathode ray tube.

scanline order An image represented by scanlines ordered by increasing the *y* coordinate.

scanned image
An image that is examined sequentially, part by part.

scatter For input and output operations, to read data from a device and locate it in noncontiguous memory addresses. Contrast with *gather* on page 2-99.

SCCS See *Source Code Control System* on page 2-222 and *Revision Control System* on page 2-205.

SCCS delta A set of changes made to an SCCS file. Creating a new delta saves only the changes made.

SCCS identification (SID)
In SCCS, a number assigned to each version of a program.

scheduling policy
The set of rules that govern when a thread will lose control of the CPU and which thread will get control next.

scope (1) That part of a source program in which a variable can communicate its value.
(2) The portion of a program within which a declaration applies. For Ada programming, see *declaration* on page 2-64.

scope of resource collection
In Workload Management, the scope of resource collection specified for a class determines how resource limits are applied to the processes in that class. The scope levels can be set to *class*, *process*, *user*, or *group*.

scope operator (::)
Defines the scope for the right argument in C++. If the left argument is blank, the scope is global. If the left argument is a class name, then the scope is within that class.

scratch file A file, usually used as a work file, that exists temporarily, until the end of the program that uses it.

screen (1) See *display screen* on page 2-72.
(2) In the extended curses library, a special type of window that is as large as the workstation screen.
(3) In Enhanced X–Windows, a server can provide several independent screens that typically have physically independent monitors (display screens). This is the expected configuration when there is only a single keyboard and pointer shared among the screens. A screen structure contains the information about that screen and is linked to the display structure.

screen capture The storage of a screen display as a text or graphics file on disk.

screen coordinates
The coordinate system that defines the display screen. In GL, distances are measured in units of pixels, and the origin is in the lower left–hand corner. On most systems the screen size is 1024 pixels high by 1280 pixels wide. The viewport defines the mapping from normalized device coordinates to screen coordinates. Synonymous with *screen space*. See also *eye coordinates* on page 2-88, *primitive coordinates* on page 2-184, *modeling coordinates* on page 2-146, *world coordinates* on page 2-265, and *transformation* on page 2-245.

screen lock In CDE, a function that locks the workstation screen, barring further input until the valid user password is entered.

Screen Saver In CDE, a choice that, after a specified time period, switches off the workstation display or varies the images that are displayed, thereby prolonging the life of the screen.

screen space Synonym for *screen coordinates* on page 2-210.

screenmask In GL, a rectangular area of the screen to which all drawing operations are clipped. It is normally set equal to the viewport and to the window. A screenmask is useful for character clipping. See also *clipping* on page 2-38.

script file In the Ada debugger, a file that contains a series of commands that can be used to drive the debugger. Script files are useful for debugging large, complex programs when you may not be able to complete a debugging session in one sitting.

scroll To move text vertically or horizontally in order to view information that is outside the display or pane boundaries.

scroll bar The horizontal and vertical bars in the border of a directory window that allow you to scroll the files to see what is beyond the border of the window. A graphical device consisting of a slider, scroll area, and scroll arrows. A user changes the view by sliding the slider up or down in the scroll area or by pressing one of the scroll arrows. This causes the view to scroll up or down in the window adjacent to the scroll bar.

scroll region In AIXwindows, the rectangular portion of a **ScrollBar** widget that contains two arrows and a slider.

scrolled list A list that is scrolled. See also *scroll* on page 2-211.

scrolled text Text that is scrolled. See also *scroll* on page 2-211.

scrolling The horizontal or vertical movement of graphic or text information presented on a display screen.

SCSI Adapter See *Small Computer Systems Interface Adapter* on page 2-220.

sdb See *symbolic debugger* on page 2-233.

SDLC See *synchronous data link control* on page 2-219.

SDLC primary station
A station that has responsibility for the data link. It issues commands to secondary stations.

SDLC secondary station
A station that responds to requests from another station (the primary station) and has little control over data link operations.

SDT See *static debugger trap* on page 2-226.

seal To encrypt a record containing several fields in such a way that the fields cannot be modified without either knowledge of the encryption key or leaving evidence of tampering.

search The action of scanning a set of data elements to locate all instances of a particular item, such as a text string or a file name.

search loop An array–processing loop used to perform a table lookup or to find exceptional values within an array.

SECAM A national television industry broadcasting standard used in France, USSR, and some other countries. See also *NTSC* on page 2-156 and *PAL* on page 2-168.

second–level interrupt handler (SLIH)
A device–dependent routine that handles the processing of an interrupt from a specific adapter. An SLIH is called by the first–level interrupt handler (FLIH) associated with that interrupt level.

secondary key A key field of a record that defines a secondary index.

secondary representation
A second form, an alternative to the primary representation, in which the client may supply an attribute value to the service.

secondary station
A data station that runs data link control functions as instructed by the primary station. It interprets received commands and generates responses for transmission.

secondary unit In Ada language, the body of a library unit (such as a subprogram body, package body, generic body, or subprogram body) or a subunit. All compilation units that are not library units are secondary units. Secondary units are not subject to reference by other independently compiled units and can be thought of as the **hidden** implementation of a library unit or **separate** declaration.

secondary window
A window of short duration such as a dialog box. The window is only displayed for a short time, usually just long enough to convey some information or get some operational directions.

section In the vi editor, text that follows a section heading as defined by the sect= option.

sector (1) The smallest amount of information that can be written to or read from a disk or diskette during a single read or write operation.
(2) On disk or diskette storage, an addressable subdivision of a track used to record one block of a program or data.

secure attention key (SAK)
A key sequence that ends all processes associated with a terminal to provide a trusted path for secure communication with the TCB. The SAK sequence is Ctrl–x followed by Ctrl–r.

secure node A node that is capable of running one or more application servers and one or more Monitor system components, normally a highly trusted machine.

security The protection of data, system operations, and devices from accidental or intentional ruin, damage, or exposure.

seek pointer A data structure that contains the offset of the current location in a character file or device.

segment (1) A contiguous area of virtual storage allocated to a job or system task. A program segment can be run by itself, even if the whole program is not in main storage.
(2) Virtual memory is divided into segments that are linearly–addressable spaces of one or more 4KB–byte pages up to a maximum size of 2 to the 28th power bytes.
(3) The information that can be addressed via a single, unique segment–register value (256MB).
(4) A portion of a computer program that can be run as an entity without the entire program being maintained in system memory.
(5) A group of display elements.
(6) In Enhanced X–Windows, one or more lines that are drawn but not necessarily connected at the end points.

segment flag The *segflag* parameter of the **fp_open** kernel service that indicates whether the path parameter is located in user space or in kernel space.

segment registers
Registers in the system that hold the actual addresses of the memory segments currently in use.

segment unit In Pascal, an independently compilable piece of code containing routines linked with the program unit. See also *program unit* on page 2-187.

segmentation fault
A memory addressing exception. Occurs when a memory location is referenced that has not been allocated to the referencing process. An out–of–bounds array reference or incorrect use of a pointer can cause a segmentation fault.

segmenting of BIUs
An optional function of path control that divides a basic information unit (BIU) received from transmission control into two or more path information units (PIUs). The first PIU contains the request header of the BIU and usually part of the response unit (RU). The remaining PIU or PIUs contain the remaining parts of the RU.

select (1) To choose a button on the display screen.
(2) To place the cursor on an object (name or command) and press the Select (left) button on the mouse or the Select key on the keyboard.
(3) To indicate the item or items the next command you choose will affect. The item may be highlighted to confirm your selection. Selecting does not actually carry out the command.

selected component
In Ada language, a name consisting of a prefix and of an identifier called the selector. Selected components are used to denote record components, entries, and objects designated by access values; they are also used as expanded names.

selecting In GL, a method for finding what primitives are being drawn in a given volume in three–dimensional space. See also *hit* on page 2-108, *selecting region* on page 2-213, *picking* on page 2-175, and *picking region* on page 2-175.

selecting region
In GL, a rhomboid–shaped volume in world coordinates that is sensitive to selecting events. If a drawing primitive draws within this region, a select event is reported. See also *hit* on page 2-108, *selecting* on page 2-213, *picking* on page 2-175, *transformation* on page 2-245, and *picking region* on page 2-175.

selection (1) Addressing a workstation or a component on a selective calling circuit. (2) The process by which a computer requests a station to send it a message.
(3) See also *addressing* on page 2-6.
(4) In Enhanced X–Windows, an indirect property of a dynamic type maintained by the client (the owner) but belonging to the user. It is not private to a particular window subhierarchy or a particular set of clients. When a client asks for the contents of a selection, it specifies a target type. This target type can be used to control the transmitted representation of the contents.

selection area In AIXwindows, a portion of a RowColumn widget over which the mouse pointer can be placed to select other widgets.

selection criteria
In the **select** subroutine, the **readlist**, **writelist**, and **exceptlist** parameter values that specify what to check for reading, writing, and exceptions.

selection range
See *key range* on page 2-124.

selector (1) In Pascal, the term in a CASE statement that, once evaluated, determines which of the possible branches of the CASE statement are processed.
(2) For Ada programming, see *selected component* on page 2-213.

semantic The relationships of characters or groups of characters to their meanings, independent of the manner of their interpretation and use.

semantic error A compile–time error caused by incorrect definition of constants and identifiers. See also *syntax error* on page 2-219.

semaphore (1) Entity used to control access to system resources. Processes can be locked to a resource with semaphores if the processes follow certain programming conventions.
(2) Provides a general method of communication between two processes that is an extension of the features of signals.

semaphore ID (semid)
An integer that points to a set of semaphores and a data structure that contains information about the semaphores.

semid See *semaphore ID* on page 2-214.

send pacing In SNA, the pacing of message units that a component is sending. Contrast with *receive pacing* on page 2-196.

sense code A value sent or received, or a negative response to indicate what error occurred.

sentence In the vi editor, text that is separated from other text by a . (period), ! (exclamation point), or ? (question mark) followed by two spaces.

separator A punctuation character that separates parts of a command or file, or that delimits character strings.

sequential access
(1) An access method in which records are read from, written to, or removed from a file based on the logical order of the records in the file.
(2) The facility to obtain data from or enter data into a storage device so that the process depends on the location of the data and on a reference to data previously accessed.

sequential file access
The location of a range of records through key values and the subsequent processing of them in some order related to those key values. The index of the file need not be unique.

sequential I/O model
A model of the operating system for all accesses to system network resources. When SNA supports this model, it simplifies access to the network, allows programs to be designed for portability, and allows programs to use network resources through redirection.

serial device A device that performs functions sequentially, such as a serial printer that prints one character at a time. Contrast with *parallel device* on page 2-168.

serial port A port used for a serial device. See also *serial device* on page 2-215.

serial processing
Pertaining to the sequential or consecutive running of two or more processes in a single device, such as a channel or processing unit. Contrast with *parallel processing* on page 2-168.

serial transmission
Transmitting each bit of a data character separately over the same electrical path.

serializability A basic property of transaction processing systems, this refers to the idea that the exchange and modification of information by transactions must be able to be synchronized and appear as though multiple, simultaneous transactions are actually a series of sequential requests. Data being changed by a transaction, or upon which a transaction depends, must be shielded from other transactions until the first transaction completes.

serialize (1) To change from parallel–by–byte to serial–by–bit.
(2) In XDR, to convert a particular machine representation to XDR format.

server (1) An application program that usually runs in the background (daemon) and is controlled by the System Program Controller.
(2) On a network, the computer that contains the data or provides the facilities to be accessed by other computers on the network.
(3) A program that handles protocol, queuing, routing, and other tasks necessary for data transfer between devices in a computer system.
(4) In Enhanced X–Windows, provides the basic windowing mechanism. It handles IPC connections from clients, de–multiplexes graphics requests onto screens, and multiplexes input back to clients.
(5) In NCS, a process that exports one or more interfaces to one or more objects, and whose procedures can be invoked from remote hosts.

server grabbing
When a client seizes the server for exclusive use to prevent processing requests from other client connections until the grab is complete. This is typically a transient state for such things as rubber–banding and pop–up menus or to run requests indivisibly.

server reporting
A protocol for servers to report to the cell manager contact by previously unknown clients, for the purpose of registration.

service access point (SAP)
In the Ethernet logical link profile, the address for the transaction program on the local system. This address is a hexadecimal value.

service controls
A group of parameters, applied to all directory operations, that direct or constrain the provision of the service.

service mode Synonym for *maintenance mode* on page 2-140.

service request number (SRN)
A group of numbers used by service technicians to determine the failing area of the system.

service transaction program
(1) A program that provides a function internal to SNA Server.
(2) A transaction program implemented by a transaction processing system. Service transaction programs perform such functions as providing access to remote data bases and remote queues. See also *application transaction program* on page 2-10 and *transaction program* on page 2-244.

service update Software that corrects a defect in or adds new function to the Base Operating System (BOS) or to an optional software product. See also *maintenance level update* on page 2-140.

session (1) The period of time during which programs or devices can communicate with each other.
(2) A name for a type of resource that controls local LUs, remote LUs, modes, and attachments.
(3) In network architecture, an association of facilities that establish, maintain, and release connections for communication between stations.
(4) The period of time during which the user of a workstation can communicate with an interactive system, usually elapsed time between login and logoff.
(5) In SNA, a logical connection between two network addressable units (NAUs) that can be activated, tailored to provide various protocols, and deactivated as requested.
(6) In remote communications, a period of communication with a remote system or host system.

session date The date associated with a session. See also *creation date* on page 2-56 and *system date* on page 2-220.

session key Used in Kerberos specifications. See also *conversation key* on page 2-54.

session–level pacing
In SNA, a flow control technique in which a receiving half–session controls the data transfer rate (the rate at which it receives request units). It is used to prevent overloading a receiver with unprocessed requests, when the sender can generate requests faster than the receiver can process them.

Session Manager
In CDE, a software application that controls saving sessions, restoring sessions, screen locking and unlocking, and the use of screen savers. When a session is saved, the state of the desktop environment (location of icons, size and location of open windows, open/closed status of applications, current color palette, and so on) is preserved so that it can be restored at the next login.

session profile For the 3270 Host Connection Program 2.1 and 1.3.3, a profile describing the characteristics of a session between a client system and a System/370 host computer. See also *3270 Host Connection Program* on page 1-1 and *profile* on page 2-186.

session records
In the accounting system, a record (produced from log in and log off records) of time connected and line usage for connected display stations.

session server In CDE, a system that provides networked sessions. Session files reside on the session server and are used whenever you log in to a system on the network.

set In NCS, to associate an allocated Remote Procedure Call (RPC) handle with a specific socket address. See also *bind* on page 2-18.

set–associative cache
A cache in which two or four (or more) lines correspond to each possible value of the virtual–address field that identifies the line to be interrogated during cache lookup.

set associativity
An aspect of cache design that determines how many cache lines can be associated with a given memory location. A cache that is four–way set associative can contain a given memory location in one of four cache lines. See also *cache line* on page 2-28.

set asynchronous balance mode (SABM)
A link control frame.

set flags Flags that can be put into effect with the shell **set** command.

set–group–ID mode bit
In setting file access permissions, sets the effective and saved group IDs of the process to the group ID of the file on execution.

set–user–ID mode bit
In setting file access permissions, sets the effective and saved user IDs of the process to the owner ID of the file on execution.

setgid See *set–group–ID mode bit* on page 2-217.

setuid See *set–user–ID mode bit* on page 2-217.

severity code A code that indicates how serious an error condition is.

shadow A darkened area below a window and to its right, or above it and to its left, which represents the shadow a window might cast.

shadow color The shaded area around or behind a dialog box.

shadow widget An opaque pointer to a structure created each time a widget is created; it identifies the widgets in the interface. Also called *swidget*.

shared library A library created by the **ld** command that contains at least one subroutine that can be used by multiple processes. Programs and subroutines are linked as before, but the code common to different subroutines is combined in one library file that can be loaded at run time and shared by many programs. A key to identify the shared library file is left in the header of each subroutine.

shared locks Shared locks are a type of lock in which multiple transactions can simultaneously lock a data item for reading. See also *exclusive lock* on page 2-84.

shared memory
An area of memory simultaneously accessible to more than one cooperating process.

shared memory ID (shmid)
An identifier assigned to the shared segment for use within a particular process. See also *file descriptor* on page 2-90.

shared port A port used by communications applications (for example, UUCP) to ensure exclusive access to a port.

Shared Product Object Tree (SPOT)
(1) A version of the **/usr** file system that diskless clients mount as their own **/usr** directory.
(2) For NIM, a **/usr** file system or an equivalent file system that is exported by servers in the NIM environment for remote client use.

shell (1) A software interface between a user and the operating system of a computer. Shell programs interpret commands and user interactions on devices such as keyboards, pointing devices, and touch–sensitive screens and communicate them to the operating system.
(2) Software that allows a kernel program to run under different operating system environments.
(3) The command interpreter that provides a user interface to the kernel. See also *shell program* on page 2-218 and *command interpreter* on page 2-42. Synonymous with *interface* on page 2-119.
(4) A shell is a command interpreter that acts as an interface between users and the operating system. A shell can contain another shell nested inside it, in which case, the outer shell is the parent shell and the inner shell is the child.
(5) In AIXwindows, **Shell** widgets are top–level widgets that are internal and cannot be instantiated, but they provide the necessary interface with the window manager. See also *shell widget* on page 2-218.

shell box A geometry management technique where a type of bounding box can have only one child that is exactly the same size as the shell.

shell command names
Operating–system commands.

shell control command
A command that enables the user to pass control to various parts of a shell procedure, or to control how a procedure ends.

shell procedure
A series of commands, combined in a file, that carry out a particular function when the file is run or when the file is specified as a value to the **sh** command. Synonymous with *shell script*.

shell program A program that accepts and interprets commands for the operating system. Synonym for *shell*.

shell prompt The character string on the command line indicating that the system can accept a command (typically the $ character).

shell script Synonym for *shell procedure* on page 2-218.

shell variables Facilities of the shell program for assigning variable values to constant names.

shell widget In Enhanced X–Windows, holds the top–level widgets that communicate directly with the window manager. These widgets do not have parents. Synonymous with *shell* on page 2-218. See also *widget* on page 2-262.

shielded twisted pair
A transmission medium of two twisted conductors with a foil or braid shield.

Shift–Japanese Industrial Standard (SJIS)
An encoding scheme consisting of single bytes and double bytes used for character encoding. Because of the large number of characters in Japanese and other Asian languages, the 8–bit byte is not sufficient for character encoding.

shmid See *shared memory ID* on page 2-217.

short In ODM, a terminal descriptor type used to define a variable as a signed 2–byte number. See also *terminal descriptor* on page 2-239.

short circuiting
The evaluation of Boolean expressions with AND and OR such that the right operand is not evaluated if the result of the operation can be determined by evaluating the left operand. The evaluation of the expression is always from left to right.

short status Status output in abbreviated form (short form) from the spooling subsystem.

shortest–job–next (SJN)
A method of queueing jobs where the shortest jobs are printed first. Contrast with *first–come–first–served* on page 2-91. See also *discipline* on page 2-71.

sibling Children of the same parent window.

SiCounter In Performance Toolbox, a value that is incremented continuously. Instruments show the delta (change) in the value between observations, divided by the elapsed time, representing a rate per second.

SID SCCS identification. The name assigned to a delta.

side effect An undesirable result caused by altering the values of nonlocal variables by a procedure or function.

sign–off To end a session at a display station.

sign–on To begin a session at a display station.

signal (1) A simple method of communication between two processes. One process can inform the other process when an event occurs.
(2) In operating system operations, a method of inter–process communication that simulates software interrupts. Contrast with *exception* on page 2-84 and *interrupt* on page 2-120.

signal handler A subroutine called when a signal occurs.

signal mask Defines the set of signals currently blocked from delivery to a process.

signal stack An alternate stack on which signals are to be processed.

signed Information is digitally signed by appending to it an enciphered summary of the information. This is used to ensure the integrity of the data, the authenticity of the originator, and the unambiguous relationship between the originator and the data.

Simple Mail Transfer Protocol (SMTP)
A protocol, typically used over a network, in which the objective is to transfer mail. SMTP is used by the **sendmail** command to accept and receive mail.

simple name For Ada programming, see *declaration* on page 2-64 and *name* on page 2-150.

simultaneous peripheral operation online
See *spooling* on page 2-223.

single buffer mode
In GL, a mode in which the frame buffer bitplanes are organized into a single large frame buffer. This frame buffer is the one currently displayed and is also the one in which all drawing occurs. See also *double buffer mode* on page 2-74.

single–byte control codes
ASCII codes 0 through 31 (0x00 through 0x1f) and delete (0x7f).

single–mode optical fiber
An optical fiber in which only the lowest–order bound mode (which can consist of a pair of orthogonally polarized fields) can propagate at the wavelength of interest. Contrast with *multimode optical fiber* on page 2-148.

single–precision
(1) The use of one computer word to represent a number, in accordance with the required precision.
(2) The specification that causes a floating–point value to be stored in the short format. See also *precision* on page 2-167.

single–processor SMP
A system designed to handle two or more processors, running the SMP version of the operating system, which has been configured with a single processor. Contrast with uniprocessor on page 2-251.

single–shift control
In codepage switching, control codes that shift to another page for a single character; nonlocking shifts.

SiQuantity value
In Performance Toolbox, represents a level, such as memory used or available disk space. The actual observation value is shown by instruments.

size The screen management action that changes the size of a window.

size field In an i–node, a field that indicates the size, in bytes, of the file associated with the i–node.

SJIS See *Shift–Japanese Industrial Standard* on page 2-218.

SJN See *shortest–job–next* on page 2-219.

skew The time difference between two clocks or clock values.

SLA Serial Link Adapter. See also *SOCC* on page 2-221.

sleeping process
A process that is waiting for input or output to complete, time slices, an event to occur, or signals from other processes. When a process is sleeping, it can be paged out of memory.

slider (1) In AIXwindows, a small interactive graphical object connected to an **XmScrollBar** bar widget. The slider controls the vertical or horizontal movement of text information or graphics across the display screen.
(2) A control that uses a track and arm to set a value from among the available values. The position of the arm (or a separate indicator) gives the currently set value.

SLIH See *second–level interrupt handler* on page 2-212.

SLIP Serial Line Interface Protocol. The protocol that TCP/IP uses when operating through a serial connection.

slot A long electrical socket inside the system unit into which an electronic circuit board (card) is installed.

slow list A list of secondary stations on a multidrop network that, due to their inactivity, are polled less often by the primary station.

small caps See *caps* on page 2-31.

Small Computer Systems Interface Adapter (SCSI Adapter)
An adapter that supports the attachment of various direct–access storage devices and tape drives to the system unit.

small word In the vi editor, a contiguous set of alphanumeric characters bounded on at least one end with a character that is not a blank, a tab, or a new–line indicator. For example, in the word *isn't,* the two sets of characters *isn* and *t* are small words. Contrast with *big word* on page 2-20.

SMIT System Management Interface Tool.

SMP See *symmetrical multiprocessor system* on page 2-233.

SMP efficient Avoidance in a program of any action that would cause functional or performance problems in an SMP environment. A program that is described as SMP efficient is generally assumed to be SMP safe as well. An SMP–efficient program has usually undergone additional changes to minimize incipient bottlenecks.

SMP exploiting Adding features to a program that are specifically intended to make effective use of an SMP environment. A program that is described as SMP exploiting is generally assumed to be SMP safe and SMP efficient as well.

SMP safe Avoidance in a program of any action, such as unserialized access to shared data, that would cause functional problems in an SMP environment. This term, when used alone, usually refers to a program that has undergone only the minimum changes necessary for correct functioning in an SMP environment.

SMT Station management.

SMTP See *Simple Mail Transfer Protocol* on page 2-219.

SNA See System Network Architecture on page 2-221.

SNA network The part of a user–application network that conforms to the formats and protocols of System Network Architecture (SNA). It enables reliable transfer of data among end users and provides protocols for controlling the resources of various network configurations. The SNA network consists of network addressable units (NAUs), boundary–function components, and the path control network.

SNBU See *switched network backup* on page 2-218.

SNMP Simple Network Management Protocol. A protocol used by network hosts to exchange information in the management of networks. SNMP network management is based on the client–server model that is widely used in TCP/IP–based network applications.

SNOBOL A programming language designed for string processing and pattern matching.

SOCC Serial Optical Channel Converter. A 220–Mbit/sec optical point–to–point link.

social science format
See *natural or social science format* on page 2-150.

socket (1) A unique host identifier created by the concatenation of a port identifier with a TCP/IP address.
(2) A port identifier.
(3) A 16–bit port number.
(4) In NCS, a port on a specific host; a communications end point that is accessible through a protocol family's addressing mechanism. A socket is identified by a socket address. See also *socket address* on page 2-221, *port* on page 2-179, and *listening* on page 2-131.

socket address
A data structure that uniquely identifies a specific communications end point. A socket address consists of a port number and a network address. It also specifies the protocol family. See also *protocol family* on page 2-189.

software Programs, procedures, rules, and any associated documentation pertaining to the operation of a system. Contrast with *hardware* on page 2-105.

software configuration
The processing required to make installed software ready to use.

software installation
The process of restoring software from external media to a local file system. The software can require further processing, or configuration, before it is ready to use.

software keyboard
A table mapping a raw keystroke to a display symbol, predefined function or string. Software keyboards that are shipped with the operating system are associated with languages (U.S. English, U.K. English, Danish, Japanese, and so on).

software keyboard map
A table that maps a keystroke to a character or to a predefined function such as a tab.

Software Vital Product Data (SWVPD)
Information that uniquely defines system, hardware, software, and microcode elements of a processing system.

sort To rearrange some or all of a group of items, based upon the contents or characteristics of those items.

source (1) A system, a program within a system, or a device that makes a request to a target. Contrast with *target* on page 2-237.
(2) In advanced program–to–program communications, the system or program that starts jobs on another system.

source code The input to a compiler or assembler, written in a source language. Contrast with *object code* on page 2-159.

Source Code Control System (SCCS)
A program for maintaining version control for the source files of a developing program. It stores the changes made to a file instead of the changed file, thus allowing several versions of the same file to exist in the system. See also *Revision Control System* on page 2-205.

source documents
Verbal information produced concurrently with the original software, by the original development company.

source file A file that contains source statements for such items as high–level language programs and data description specifications. A file containing input data or commands.

source module See *source program* on page 2-222.

source program
A computer program expressed in a source language.

source statement
A statement written in a programming language.

space (1) A site intended for storage of data, such as a location in a storage medium.
(2) A basic unit of area, usually the size of a single character.
(3) One or more space characters.
(4) In a neutral circuit, an impulse that causes the loop to open or causes absence of signal. In a polar circuit, it causes the loop current to flow in a direction opposite to that for a mark impulse. A space impulse is equal to a binary zero.

sparse array An array in which few of the defined cells are used.

sparse file A file that is created with a length greater than the data it contains, leaving empty spaces for future addition of data. See also *hole in a file* on page 2-108.

SPC See *System Program Controller* on page 2-235.

special character
A character other than a letter or number. For example, *, +, and % are special characters.

special file Used in the operating system to provide an interface to input/output devices. There is at least one special file for each device connected to the computer. Contrast with *directory* on page 2-70 and *file* on page 2-90. See also *block file* on page 2-20 and *character special file* on page 2-34.

specific The attribute types that may appear in an instance of a given class, but not in an instance of its superclasses.

specification statement
In FORTRAN, one of the set of statements that provide the compiler with information about the data used in the source program and how to allocate storage.

specifiers Used in C++ declarations to indicate storage class, fundamental data type, and other properties of the object or function being declared.

speed The baud rate. Synonym for *line speed* on page 2-129.

SPI Stub programming interface. A private RPC runtime interface whose routines are unavailable to application code.

spill area A storage area used to save the contents of registers.

Spmi See *System Performance Measurement Interface* on page 2-235.

spool file (1) A disk file containing output that has been saved for later printing.
(2) Files used in the transmission of data among devices.

spooler A synonym for the *queueing system* that pertains to its use for queueing print jobs.

spooling (simultaneous peripheral operation online)
(1) The use of auxiliary storage as a buffer storage. This reduces processing delays when transferring data between peripheral equipment and the processors of a computer.
(2) Reading and writing input and output streams on an intermediate device in a format convenient for later processing.
(3) Performing a peripheral operation such as printing while the computer is busy with other work.

SPOT See *Shared Product Object Tree* on page 2-217.

SPP Sequence packet protocol. The primary transport–layer protocol in the Xerox Network Systems. It provides reliable, flow–controlled, two–way transmission of data for an application program. It is a byte–stream protocol used to support the SOCK_STREAM abstraction. The SPP protocol uses the standard Network System (NS) address formats.

spring–loaded pop–up
A kind of widget, such as a menu, that is not visible to the window manager. The spring–loaded pop–up disables user–event processing except for events that occur in the menu.

SRC System Resource Controller.

SRF *Standard record format*.

SRN See *Service request number* on page 2-216.

SRT See *Structure Rule Table* on page 2-228.

SSCP See *system services control point* on page 2-221.

stack (1) An area in storage that stores temporary register information and return addresses of subroutines.
(2) A list constructed and maintained so that the last data element stored is the first data element retrieved.
(3) In kernel mode, an area that is paged with the user process. The kernel maintains a stack for each process. It saves the process information such as the call chain and local variables used by the kernel for the user process.

stack buffer A storage area that stores retrievable data in sequence. The last text stored is the first text removed.

stack overflow An error condition in DOS that results from an insufficient number of stack frames, which are used by DOS to handle hardware interrupts.

stack pointer A register providing the current location of the stack.

stack traceback
The calling sequence that indicates the path taken by a process to get to its current location.

stacked tape A bootable tape with multiple software images.

stacking order The relationship between sibling windows that stack on top of each other.

stage One of a series of steps to enter a **ged** subcommand that typically ends with < cr > (carriage return). Each subcommand consists of a subset of stages, including command line, text, points, pivot, and destination.

standalone A machine in the network installation environment that accesses all required resources locally.

standalone shell
A limited version of the shell program used for system maintenance.

standalone system
See *standalone workstation* on page 2-224.

standalone workstation
A workstation that can perform tasks without being connected to other resources such as servers or host systems.

standard error (STDERR)
The place where many programs place error messages.

standard input (STDIN)
The primary source of data going into a command. Standard input comes from the keyboard unless redirection or piping is used, in which case standard input can be from a file or the output from another command.

Standard I/O Board
The Standard I/O Board provides a group of I/O functions that are basic to most system units. Common standard I/O functions are keyboard, tablet, speaker, mouse, serial port, parallel port and diskette adapter.

standard output (STDOUT)
The primary destination of data coming from a command. Standard output goes to the display unless redirection or piping is used, in which case standard output can be to a file or another command.

standard screen
In the extended curses library, a memory image of the screen to which the routines make changes.

standout mode The general–purpose highlighting mechanism used by the **terminfo** structure.

stanza A group of lines in a file that together have a common function or define a part of the system. Stanzas are usually separated by blank lines or colons, and each stanza has a name.

start method Takes the device from the stopped state to the available state. The start method applies only to devices that support the optional stopped state.

start–stop Asynchronous transmission in which a group of signals representing a character is preceded by a start element and followed by a stop element. See also *asynchronous transmission* on page 2-13.

startup set A grouping of application servers that can be thought of as a single unit for administration purposes.

state (1) A state in which the circuit remains until application of a suitable pulse. (2) One of the separate, restartable portions into which the **runacct** command (the main daily accounting shell procedure) breaks its processing.

state information
Information about the current state of the appearance and behavior of a widget or gadget. This information is recorded within each individual widget and gadget and updated as necessary.

state instrument
In Performance Toolbox, a state instrument shows the latest statistics for a system resource, optionally as a weighted average. While it does not show the statistics over time, some state instruments collect this data in case you want to change the instrument to a recording instrument. Types of graphs used to plot these recordings include state bar, state light, pie chart, and speedometer. Contrast with *recording instrument* on page 2-197.

statement (1) An instruction in a program or procedure.
(2) In programming languages, a language construct that represents a step in a sequence of actions or a set of declarations. See also *block statement* on page 2-20. In Ada language, a statement specifies one or more actions to be performed during the execution of a program.

statement function
In FORTRAN, a name, followed by a list of dummy arguments, that is equated to an arithmetic, logical, or character expression, and that can be substituted for the expression throughout the program. See also *macro* on page 2-139.

statement function definition
In FORTRAN, a statement that defines a statement function. Its form is a statement function followed by = (equal sign) followed by an arithmetic, logical, or character expression.

statement label
In FORTRAN, a number containing one to five decimal digits that is used to identify a statement. A statement label is usually used to transfer control, define the range of a DO loop, or refer to a FORMAT statement. See also *label* on page 2-126.

statement number
See *statement label* on page 2-225.

static (1) A style of creating pop–ups.
(2) In C++, a keyword used for defining the scope and linkage of variables and functions. For internal variables, the variable has block scope and retains its value between function calls. For external values, the variable has file scope and retains its value within the source file. For class variables, the variable is shared by all objects of the class and retains its value within the entire program.

static binding Binding that occurs at compilation time based on the resolution of overloaded functions.

static debugger trap (SDT)
A trap instruction placed in a predefined point in code that calls the debug program. The trap instruction causes a program check when run and, as a result of the program check, the debug program is activated.

static display In text formatting, when the **nroff** command finds a block of text in the input file that has been specified as a static display, it places the text on the current page only if there is room for the entire block. If there is not enough room, the **nroff** command starts a new page and places the block of text there. See also *floating display* on page 2-93.

static linking Linking of a program in which library procedures are incorporated into the load module, instead of being dynamically loaded from their library each time the program is run.

static memory Allocated memory of fixed size.

static routing A method of setting paths between hosts, networks, or both by manually entering routes into the routing table. Static routes are not affected by routing daemons and must be updated manually.

static variable A variable that is allocated as soon as a program starts running and that remains allocated until the program stops. Normal scoping rules apply to the variable. Contrast with *automatic variable* on page 2-15.

station (1) A computer or device that can send or receive data.
(2) An input or output point of a system that uses telecommunication facilities, such as one or more systems, computers, workstations, devices, and associated programs at a particular location that can send or receive data over a telecommunication line.
(3) A location on a device at which an operation is performed.
(4) In FDDI, an addressable logical and physical attachment in a ring capable of transmitting, receiving, and repeating information.
(5) In SNA, a link station.

statistic line In Performance Toolbox, the lines in a list that represent a specific value. *Contrast with context line* on page 2-52.

status (1) The current condition or state of a program or device. For example, the status of a printer.
(2) The condition of the hardware or software, usually represented in a status code.
(3) In Enhanced X–Windows, many **Xlib** subroutines return a success status. If the subroutine does not succeed, however, its values are not disturbed.

STDERR See *standard error* on page 2-224.

STDIN See *standard input* on page 2-224.

STDOUT See *standard output* on page 2-224.

steal (a page frame)
The act (by the Virtual Memory Manager) of reallocating a real–memory page frame that contains a virtual–memory page that is being used by a currently executing program.

sticky bit An access permission bit that causes an executable program to remain on the swap area of the disk. Only someone with root authority can set the sticky bit. This bit is also used on directories to indicate that only file owners can link or unlink files in that directory.

stipple A bitmap used to tile a region. A stipple pattern serves as an additional clip mask for a fill operation with the foreground color.

stop bit (1) In start–stop transmission, a signal at the end of a character that prepares the receiving device for reception of a subsequent character.
(2) A signal to a receiving mechanism to wait for the next signal.

stop method Takes the device from the available state to the stopped state. The stop method applies only to devices that support the optional stopped state.

stop record In Performance Toolbox, a special type of value record which signals that recording was stopped for a set of statistics and gives the time it happened. This allows programs using the recording file to distinguish between gaps in the recording and variances in the recording interval.

stopped state Allows a device to be made unavailable but still have its device driver loaded and bound in the kernel and still be known by the device driver.

storage (1) The location of saved information.
(2) In contrast to memory, the saving of information on physical devices such as disk or tape. See also *memory* on page 2-143.
(3) A unit into which recorded text can be entered, retained, and processed, and from which it can be retrieved. (4.) The action of placing data into a storage device.

storage class specifier
A storage class keyword. One of the following C++ keywords: **auto**, **register**, **static**, or **extern**.

storage device (1) A functional unit for storing and retrieving data.
(2) A facility into which data can be retained.

store To place information in a storage device (in memory or onto a diskette, fixed disk, or tape), so that it is available for retrieval and updating.

stream (1) Sequential input or output from an open file descriptor.
(2) A continuous stream of data elements being transmitted, or intended for transmission, using a defined format.
(3) All data transmitted through a data channel in a single read or write operation. Synonym for *data stream* on page 2-62.
(4) The kernel aggregate created by connecting STREAMS components, resulting from an application of the STREAM mechanism. The primary components are a stream head, a driver, and zero or more pushable modules between the stream head and driver. A stream forms a full duplex processing and data transfer path in the kernel, between a user process and a driver. A stream is analogous to a shell pipeline except that data flow and processing are bidirectional.

stream buffer A C++ stream buffer is a buffer between the ultimate consumer and the I/O Stream Library functions that format data. It is implemented in the I/O Stream Library by the streambuf class and the classes derived from streambuf.

stream collection
A method of collecting auditing data that writes audit records to a circular buffer within the kernel. The data can be displayed, or printed to provide a paper audit trail, or converted into bin records.

stream editor The **sed** command, which modifies lines from a specified file, according to an edit script, and writes them to a standard output.

stream end The end of the stream furthest from the user process. The stream end contains the driver.

stream head The end of the stream closest to the user process. The stream head provides the interface between the stream and the user process. The principal functions of the stream head are processing STREAMS–related system calls, and bidirectional transfer of data and information between a user process and messages in STREAMS' kernel space.

streaming tape device
See *streaming tape drive* on page 2-228.

streaming tape drive
A magnetic tape unit that stores large amounts of data and is designed to make a nonstop dump or restore of magnetic disks without using interblock gaps.

STREAMS A kernel mechanism that supports development of network services and data communication drivers. It defines interface standards for character input and output within the kernel, and between the kernel and user level. The STREAMS mechanism comprises integral functions, utility routines, kernel facilities, and a set of structures.

strength reduction
An optimization that replaces an arithmetic operation with a functionally equivalent arithmetic optimization of lesser strength. For example, 4*2 can be transformed into 4+4.

strict type checking
Checking data types for compliance with the rules of C language more strictly than the C compiler, such as with the **lint** program.

stride The relationship between the layout of an array's elements in memory and the order in which those elements are accessed. A stride of 1 means that memory–adjacent array elements are accessed on successive iterations of an array–processing loop. A stride of N means that for each array element accessed, N–1 memory–adjacent elements are skipped over before the next accessed element.

string (1) A linear sequence of entities such as characters or physical elements. Examples of strings are alphabetic string, binary element string, bit string, character string, search string, and symbol string.
(2) In Pascal, an object of the predefined type STRING.
(3) The form of data used in programming languages for storing and manipulating text. In C language code, a string is treated as a one–dimensional array of type **char**.

string constant
Characters enclosed in double quotation marks.

string register A register that holds a defined string value to be called by a token. See also *token* on page 2-242.

string value Value of specified string. In AIXwindows, the value of a string that identifies a Text widget.

stroke text Synonym for *programmable character set* on page 2-188 and *geometric text* on page 2-100.

structure A variable that contains an ordered group of data objects. Unlike an array, the data objects within a structure can have varied data types.

Structure Rule Table (SRT)
A recurring attribute of the directory schema with the description of the permitted structures of distinguished names.

structure tag The identifier that names a structure data type.

structured field
A mechanism that permits variable length data to be encoded for transmission in the data stream. See also *field* on page 2-89.

structured file (1) A special type of INed file that contains specialized data, such as information about the structure of the data in the file, and history information about changes that have been made to the file. Structured files can contain hierarchical data that is displayed and edited by using forms.
(2) In Encina, a file with data organized into a specific format that is usually record–oriented.

structured file system
The collection of data managed by a single structured file server (SFS). All access to a structured file system is through a single server, using a special type of file descriptor (OFD) that identifies the file system and its organization.

structured programming
A technique for organizing computer programs in hierarchical modules, making programs easier to debug, modify, and replace. Typically, all modules have a single entry point and a single exit point. Control is passed downward through the structure without unconditional branches to higher levels of the structure.

structured type
Any of several data types that define variables having multiple values; for example, records and arrays. Each value is a component of the structured type. Contrast with *scalar type* on page 2-209.

stub (1) In NCS, a program module that transfers remote procedure calls and responses between a client and a server. Stubs perform marshalling, unmarshalling, and data format conversion. Both clients and servers have stubs. The compiler generates client and server stub code from an interface definition. See also *marshal* on page 2-141.
(2) Hooking functions used as extensions to the protocol to generate protocol requests for Enhanced X–Windows. Synonym for *hooking routines* on page 2-108.
(3) The RPC calls produced by the compiler when an interface is defined. Two sets of stubs are produced, client stubs and server stubs. The application code calls the stub, and the RPC mechanism translates this into a call to the appropriate function on the remote machine.

Style Manager In CDE, the software application used to customize some of the visual elements and system device behaviors of the workspace environment, including colors and fonts, and keyboard, mouse, window, and session start–up behaviors.

stylus A device used to select a particular location on a tablet.

subaddress In X.25 communications, the unallocated digits at the end of the national terminal number (NTN). If the network provider allocates all digits to the NTN, there can be no subaddress.

subarea node In data communications, a node that uses network addresses for routing, and whose routing tables are affected by changes in the configuration of the network. Subarea nodes can provide boundary function support for peripheral nodes.

subchannel A logical communications path defined in S/370 architecture to perform transfers to a given device.

subclass A class of widgets that inherits resources from a higher class.

subcommand A request for an operation that is within the scope of work requested by a previously issued command.

subcomponent In Ada language, either a component, or a component of another subcomponent.

subdirectory In the file system hierarchy, a directory contained within another directory.

subfolder In CDE, a folder contained within another folder (sometimes called the *parent folder* on page 2-170). When discussing command–line activities, this may be called a *subdirectory*.

subheap In Pascal, part of a heap delimited by a call to MARK. Subheaps are treated in a stack–like manner within a heap.

subhost A communications system that controls attached workstations in addition to communicating with another (usually higher–level) system.

subject identifier (SID)
A string that identifies a user or set of users. Each SID consists of three fields in the form person.group.organization. In an account, each field must have a specific value; in a ACL entry, one or more fields may be a wildcard.

submenu A menu accessed from another menu by a —> symbol. A related menu that can only be reached from a main menu. In AIXwindows, one example of a submenu is an **XmCascadeMenu** widget that appears from the side of an **XmPopupMenu** widget when the mouse pointer is dragged sideways across a main menu item.

subnet One of a group of multiple logical network divisions of a single network, such as can be created by the TCP/IP Interface Program. Synonymous with *subnetwork*.

subnet address
The subdivided part of the local host address, which has been reserved for indicating the subnet. Subnet addressing allows an autonomous system made up of multiple networks to share the same Internet network address.

subnet address mask
A bit mask used by a local system to determine whether a destination is on the same network as the source or if the destination can be reached directly through one of the local interfaces.

subnetwork Synonym for *subnet* on page 2-230.

subobject An object that is in a subordinate relationship to a given object.

subpanel In CDE, an extension of the Front Panel that slides up providing access to additional elements. Subpanels usually contain groups of related elements.

subpattern A discrete element of a *regular expression* on page 2-198.

subprocess A process initiated by another process. Control is transferred back to the main process after the subprocess finishes running.

subprogram (1) A program called by another program, such as a subshell.
(2) In FORTRAN, a program unit that has a FUNCTION, SUBROUTINE, or BLOCK DATA statement as its first statement. Contrast with *main program* on page 2-140.
(3) In Ada language, a subprogram is either a procedure or a function. A procedure specifies a sequence of actions and is invoked by a procedure call statement. A function specifies a sequence of actions and also returns a value called the result, and so a function call is an expression. A subprogram is written as a subprogram declaration, which specifies its name, formal parameters, and (for a function) its result; and a subprogram body which specifies the sequence of actions. The subprogram call specifies the actual parameters that are to be associated with the formal parameters. A subprogram is one of the kinds of program unit. See also *function* on page 2-97 and *procedure* on page 2-185.

subrange scalar type
In Pascal, a type that defines a variable whose value is restricted to some subset of values of a base scalar type. See also *base scalar type* on page 2-18.

subroutine (1) A sequenced set of statements or coded instructions that can be used in one or more computer programs and at one or more points in a computer program.
(2) A routine that can be part of another routine. See also *routine* on page 2-207.
(3) A request by an active process for a service by the system kernel. See also *macro* on page 2-139.

Subroutine ID A unique identification number associated with each subroutine included in an application.

subroutine switch table
Contains the address for the specific handler routine that handles the subroutine.

subscribe In X.25 communications, to a rent an X.25 line, specifying the required facilities.

subscript (1) An integer or variable whose value selects a particular element in a table or an array.
(2) Characters printed one–half line below the normal printing line.

subscript declarator
In an array definition or declaration, the bracketed expressions following the array name. Specifies the number of elements in an array dimension.

subscript quantity
In FORTRAN, a component of a subscript. A subscript quantity is an integer or real constant, variable, or expression.

subserver A system resource or program that is directly controlled by a server program running under control of the System Program Controller.

subset (1) A set each element of which is an element of a specified other set.
(2) A variant form of a programming language with fewer features or more restrictions than the original language.
(3) In telecommunications, a subscriber set such as a telephone.

subshell An instance of the shell program started from an existing shell program.

substring A contiguous subportion of a string.

subsystem (1) A secondary or subordinate system, usually capable of operating independently or synchronously with a controlling system.
(2) The part of communications that handles the requirements of the remote system, isolating most system–dependent considerations from the application program.

subtree A lower–level directory structure.

subtype (1) An IOCINFO ioctl variable that identifies the kind of DLC being queried.
(2) In Ada language, a subtype of a type characterizes a subset of the values of the type. The subset is determined by a constraint on the type. Each value in the set of values of a subtype belongs to the subtype and satisfies the constraint determining the subtype.

subunit For Ada programming, see *body* on page 2-18.

subwidget In AIXwindows and Enhanced X–Windows, a widget class directly beneath a higher widget class in a widget–gadget hierarchy.

suffix (1) A character string attached to the end of a file name that helps identify its file type.
(2) A code dialed by a caller who is already engaged in a call.
(3) A part of a file name, added at the end, separated from other suffixes or the base file name by some punctuation, such as a period (.).

superblock In a file system layout, refers to Block 1, which is used to keep track of the file system and is the most critical part of the file system. It contains information about every allocation or deallocation of a block in the file system. See also *i–list* on page 2-111.

superclass In AIXwindows and Enhanced X–Windows, a class of widgets that passes inheritable resources down the hierarchy to a lower subclass. See also *widget record* on page 2-262.

superclient A diskless client with read and write permission and root access to the remote resources used by other clients. A superclient installs and maintains optional software of Version 3.2 of the operating system on a non–Version 3.2 diskless server.

superobject An object that is in a superior relationship to a given object.

superscalar The capability to execute multiple instructions in a given clock cycle.

superuser See *root user* on page 2-206.

superuser authority
See *root user authority* on page 2-206.

supervisor The part of the operating system control program that coordinates the use of resources, and maintains the flow of processing unit operations.

supervisor call (SVC)
An instruction that interrupts the program being run and passes control to the supervisor so it can perform a specific service indicated by the instruction.

supporters In Ada language, all the compilation units required by the language to allow a unit to be compiled. This consists of the unit's imports, their imports, and so on.

surface characteristics
Characteristics of the style of a written document: readability, sentence length and structure, word length and usage, verb type, and sentence openers.

suspended state
(1) A state in which the resource is temporarily not receiving a request. A start action request returns the resource to the state it was in prior to being suspended.
(2) A software state in which a task is not dispatched by the system and is not contending for the processor.

SVC See *switched virtual circuit* on page 2-218 and *Supervisor call* on page 2-232.

swap interval In GL, the amount of elapsed time between frame buffer swaps. The system waits at least the amount of time specified by the **swap interval** subroutine before honoring a request to exchange the front and back buffers. The swap interval is measured in units of vertical retraces, which occur every 30th of a second on most systems. The swap interval is useful in achieving smooth–flowing animation.

swapping (1) Temporarily removing an active job from main storage, saving it on disk, and processing another job in the area of main storage formerly occupied by the first job.
(2) In a system with virtual storage, a paging technique that writes the active pages of a job to auxiliary storage and reads pages of another job from auxiliary storage into real storage.

swidge See *shadow widget* on page 2-217.

switch A command–line option.

switch expression
(1) The expression that is located between the keyword **switch** and the body of a **switch** statement.
(2) In C language, the controlling expression of a switch statement.

switch table The table used by the file system to locate the entry points of a character device.

switched line n data communications, a connection between computers or devices established by dialing. Contrast with *nonswitched line* on page 2-155.

switched network backup (SNBU)
In data communications, a technique that provides a switched line connection when a nonswitched line fails.

switched virtual circuit (SVC)
In X.25 communications, a virtual circuit that is requested by a virtual call. It is released when the virtual call is cleared. Contrast with *permanent virtual circuit* on page 2-173. See also *virtual circuit* on page 2-258.

SWVPD See *Software Vital Product Data* on page 2-222.

symbol table See *parse* on page 2-170.

symbolic address
A unique line address, such as . (period) or $ (dollar sign), used in place of a line number address to identify location of data. See also *pattern address* on page 2-172.

symbolic debugger (sdb)
A tool that aids in the debugging of programs written in certain high–level languages.

symbolic link Type of file that contains the path name of and acts as a pointer to another file or directory.

symbolic name
A unique name used to represent an entity such as a file or a data item. See also *name* on page 2-150.

symmetrical multiprocessor (SMP) system
A system containing multiple processors that are essentially identical and perform identical functions.

SYN See *synchronization character* on page 2-219.

synchronization character (SYN)
In binary synchronous communications, the transmission control character that provides a signal to the receiving station for timing.

synchronous (1) Two or more processes that depend upon the occurrences of specific events such as common timing signals.
(2) Occurring with a regular or predictable time relationship or sequence.

synchronous data link control (SDLC)
(1) A form of communications line control using commands to control the transfer of data over a communications line. Contrast with *binary synchronous communication* on page 2-17.
(2) A discipline conforming to subsets of the advanced data communications control procedures (ADCCP) of the ANSI and the HDLC of the International Organization for Standardization. It manages synchronous, code–transparent, serial–by–bit information transfer over a link connection. Transmission exchanges can be duplex or half–duplex over switched or nonswitched links. The configuration of the link connection may be point–to–point, multipoint, or loop.

synchronous transmission
(1) In data communications, a method of transmission in which the sending and receiving of characters is controlled by timing signals. Contrast with *asynchronous transmission* on page 2-13.
(2) Data transmission in which the time of occurrence of each signal representing a bit is related to a fixed time base.

syntax (1) The grammatical rules for constructing a command, statement, or program.
(2) In XOM:

- An OM syntax is any of various categories into which the object management specification statically groups values on the basis of their form. These categories are additional to the OM type of the value.
- A category into which an attribute value is placed on the basis of its form.

syntax diagram
A diagram for a command that displays how to enter the command on the command line.

syntax error A compile–time error caused by incorrect syntax. See also *semantic error* on page 2-214.

syntax template
A lexical construct containing an asterisk from which several attribute syntaxes can be derived by substituting text for the asterisk.

system The computer and its associated devices and programs.

System/370 **Host Interface Adapter (HIA)**
An adapter that allows the attachment of a POWERstation or POWERserver to a 5088 Graphics Control Unit.

system address list
The address list, controlled by the system manager, that all users on the system can use with the **xtalk** command to make outgoing X.25 calls. See also *address list* on page 2-6 and *user address list* on page 2-253.

System Application Architecture FORTRAN (SAA FORTRAN)
A superset of the ANSI X3.p – 1978 FORTRAN 77 standard.

system board The main circuit board in the system unit that supports a variety of basic system devices, such as a keyboard, a mouse, and so forth. The system board also supplies other basic system functions.

system call A call by a program to an operating system subroutine.

system console
A console, usually equipped with a keyboard and display screen, that is used by an operator to control and communicate with a system. Synonymous with *console* on page 2-51.

system customization
Specifying the devices, programs, and users for a particular data processing system. Contrast with *configuration* on page 2-48. See also *customization* on page 2-58.

system date The date assigned by the system user during setup and maintained by the system. See *creation date* on page 2-56 and *session date* on page 2-216.

system dump A copy from storage of selected data areas. Synonymous with *kernel dump*.

system image The representation of a program (and its related data) as it exists at the time it resides in system memory.

system management
The tasks involved in maintaining the system in good working order and modifying the system to meet changing requirements.

System Management Interface Tool (SMIT)
A set of menu–driven services that facilitate the performance of such system tasks as software installation and configuration, device configuration and management, problem determination, and storage management. SMIT is provided in both a character–based curses interface and an AIXwindows–based graphical user interface.

system memory
Synonymous with *main storage* on page 2-140, but used in hardware to refer to semiconductor memory (modules).

system menu In AIXwindows, the pulldown in the top left–hand corner of a window that allows users to restore, move, size, minimize, and maximize the window. It also allows users to exit the application or to close a window. Also causes the appearance of a dialog box to contain a list of the active applications. With the optional split window technique, the user views many parts of the same object at one time.

System Network Architecture (SNA)
(1) An architecture for controlling the transfer of information in a data communications network.
(2) The description of the logical structure, formats, protocols, and operating sequences for transmitting information units through, and controlling the configuration and operation of networks.

system node In the hierarchy of device locations, this is the highest node. Every hardware device will lead back to the system node if you follow the connection path. For example, an SCSI disk is connected to an SCSI adapter that is connected to a bus that is connected to the system node.

system parameters
Synonym for *kernel parameters* on page 2-124.

System Performance Measurement Interface (Spmi)
In the Performance Toolbox, the Agent API that allows an application program to register custom performance statistics about its own performance or that of some other system component. Once registered, the custom statistics become available to any consumer of statistics, local or remote. Also permits applications to access statistics on the local system without using the network interface. Such applications are called local data–consumer programs.

system profile A file containing the default values used in system operations.

System Program Controller (SPC)
A system program that controls the operation of other application programs that run in the background (daemons).

system prompt Synonym for *command line* on page 2-42. The system prompt is the symbol that appears at the command line of an operating system. The system prompt indicates that the operating system is ready for the user to enter a command.

System Resource Controller (SRC)
A set of commands and subroutines used to create and control subsystems. The SRC controls subsystem processes using a common command line and the C interface. The SRC is useful when you need a common method to start, stop, and collect status information on processes.

system restart Synonym for *initial program load* on page 2-114.

system ROS The piece of system microcode that is responsible for loading a boot image.

system services control point (SSCP)
In SNA, the focal point within an SNA network for managing the configuration, coordinating network operator and problem determination requests, and providing directory support and other the session services for network end users. Multiple SSCPs, cooperating as peers, can divide the network into domains of control, with each SSCP having a hierarchical control relationship to the physical units and logical units within its domain.

system startup Synonym for *initial program load* on page 2-114.

system time The amount of time that the operating system spends providing services to an application. System time includes time spent by the operating system allocating storage or devices to your program, and time spent processing operating system calls your program makes.

system unit The part of the system that contains the processing unit.

system user A person, device, or system that uses the facilities of a computer system.

T

tab To move a cursor to a preset location on a display screen.

tab group In AIXwindows, a means of organizing **XmPrimitive** widgets into groups for more efficient traversal within and between groups.

table An array of data in which each item can be unambiguously located by means of one or more values.

tablet A special flat surface with a mechanism for indicating positions on it. A tablet is normally used as a locator.

tablet origin A point on a tablet to which all other locations on the tablet correspond. The origin is either the lower–left corner or the center of the tablet.

tabulating window
In Performance Toolbox, a special form of window that tabulates the values of an instrument as data is received and can also calculate a line with a weighted average for each value.

tag (1) In GL, a marker in the display list used as a location for display list editing.
(2) In Interleaf, the alphanumeric name of an autonumber reference. This name can be assigned by Interleaf or by the writer; but it must match the autonumber of the component it is referencing.

tag field In Pascal, the field of a record that defines the structure of the variant part. See also *variant part* on page 2-255.

tag–in A control line bus that passes signals from the control unit to the host on the System 360/370 Parallel Channel Interface.

tag–out A control line bus that passes signals from the host to the control unit on the System 360/370 Parallel Channel Interface.

Tagged Input Format File (TIFF)
A graphics file format using bitmaps.

tape drive A mechanism for moving magnetic tape and controlling its movement.

target (1) A system, a program within a system, or a device that interprets, rejects, or satisfies, and replies to requests received from a source. Contrast with *source* on page 2-222.
(2) The node at which a password is to be installed. If the password specifies a single nodelocked license, the target is the node licensed to run the product. If the password specifies multiple nodelocked licenses (that is, a compound password for nodelocked licenses), or licenses of any other type, then the target is a node running the license server daemon.
(3) For NIM, the client you are installing.

target file A file created by the **make** program that contains a completed program.

target program Synonym for *object program* on page 2-159.

task (1) A basic unit of work to be performed. Some examples include a user task, a server task, and a processor task.
(2) A process and the procedures that run the process.
(3) In a multiprogramming or multiprocessing environment, one or more sequences of instructions treated by a control program as an element of work to be accomplished by a computer.
(4) In Ada language, a task operates in parallel with other parts of the program. It is written as a task specification (which specifies the name of the task and the names and formal parameters of its entries), and a task body which defines its execution. A task unit is one of the kinds of program unit. A task type is a type that permits the subsequent declaration of any number of similar tasks of the type. A value of a task type is said to designate a task.
(5) A C++ task is a lightweight, nonpreemptive routine that you can use to simulate the operation of programs. Tasks are nonpreemptive because only a single task is executing at any one time. Tasks are lightweight because less time and space are required to create a task than a true operating–system process.

task ID In Ada language, analphabetic label or identification for a task. This label is determined by the debugger TASKS option. A task ID is assigned to each task that has not terminated.

Task Library A C++ class library that provides the facilities to write programs that are made up of tasks.

tasking Synonym for *multitasking* on page 2-149.

tbl A preprocessor that formats tables for the **nroff** and **troff** commands.

TCB See *trusted computing base* on page 2-247.

TCP See *Transmission Control Protocol* on page 2-246.

TCP/IP See *Transmission Control Protocol/Internet Protocol* on page 2-246.

TCW Translation control word.

TD Transmit Data used with EIA–232 Protocol.

telecommunication The transmission of control signals and information between computer systems at two or more locations over telecommunication lines, or between a computer system and remote devices.

teleprocessing Processing data that is received from or transmitted to a remote location by way of communication channels. Synonym for *remote access data processing*.

Telnet In TCP/IP, the protocol that opens the connection to the system.

template (1) A representation of a keyboard that includes functions not engraved on the keyboard.
(2) Each command line stored in the buffer.
(3) In enhanced edit mode, a special character buffer associated with the terminal.
(4) In Mailer, an ASCII file you can create for conveniently including frequently typed information, such as your name and electronic mail address, in your mail messages.
(5) A family of C++ classes or functions with variable types.

temporary scalar In a source–level optimization, a scalar that is used to temporarily hold the contents of an array element or the result of a computation to eliminate logically redundant stores and loads of that element or result.

term An expression that is a subportion of another expression, usually composed of other subportions by means of addition or subtraction.

terminal (1) A device, usually equipped with a keyboard and a display device, capable of sending and receiving information over a communications line. See also *workstation* on page 2-265.
(2) In a system or communications network, a point at which data can either enter or leave.
(3) In curses and extended curses, a special screen that represents what the workstation's display screen currently looks like. The terminal screen is identified by a window named **curscr**, which the user does not access directly. Instead, users make changes to the **stdscr** window (or a user–defined screen) and then the **refresh** (or **wrefresh**) window to update the terminal.

Terminal In CDE, in the Front Panel, an item in the Personal Applications subpanel used to open a terminal window.

terminal application
An application running at a workstation (terminal).

terminal descriptor
In ODM, a named variable of type **short**, **long**, **binary, char**, or **vchar** used to define the basic data types in an ODM object class definition. See also *short* on page 2-218, *long* on page 2-136, *binary* on page 2-17, *char* on page 2-33, *descriptor* on page 2-67, *vchar* on page 2-256, and *object class* on page 2-158.

terminal emulator
A program that allows a device such as a microcomputer or personal computer to enter and receive data from a computer system as if it were a particular type of attached terminal.

terminal mapping
To translate between a standard character set and a terminal–specific character set.

terminal screen
Synonym for *display screen* on page 2-72.

terminator The part of the program product that performs the action necessary to end a job or program.

test mode Mode employed in testing a new user interface, in which the compiled application is used to drive the interface. Allows building, modifying, testing, and refining operations without having to compile, link, and debug.

text (1) A type of data consisting of a set of linguistic characters (letters, numbers, and symbols) and formatting controls.
(2) The portion of a program that is able to be run.
(3) In kernel mode, contains kernel program code that runs. It is read only by a user process.
(4) In ASCII and data communications, a sequence of characters treated as an entity when preceded by one start–of–text and ended by one end–of–text communication control character. See also *program text* on page 2-187.
(5) In word processing, information intended for human viewing that is presented in a two–dimensional form, such as data printed on paper or displayed on a screen.
(6) The part of a message that is not the header or control information.

text buffer A text storage area.

text cursor A cursor that indicates where to type a character. The text cursor is controlled by the keyboard.

Text Editor In CDE, the software application used to create and edit documents.

text field In CDE, a rectangular area in a window where information is typed. Text fields with keyboard focus have a blinking text insertion cursor.

text formatting program
A program that determines the manner in which data will be placed on a page.

text indicators Symbols at the bottom of the INed window to show directions the user can scroll to view additional portions of the file.

text input mode
Mode in which typed characters are interpreted by an editor as text entered into a file. See also *command mode* on page 2-42 and *last line mode* on page 2-126.

text lock Allows the calling process to lock or unlock its text segments into memory.

text string A sequence of characters (alphanumeric or special) defined by the user.

text widget A text editor for customizing user interfaces and programmatic interfaces.

text wrap See *word wrap* on page 2-263.

textport In GL, a region on the display screen used to present textual output from graphical or nongraphical programs.

texture A pattern used to fill rectangles, convex polygons, arcs, and circles.

tftp See *Trivial File Transfer Protocol* on page 2-247.

thermal output The heating load that a computer system places on the cooling system of the building, measured in British Thermal Units (BTU).

this A C++ keyword that identifies a special type of pointer that references the class object in a member function.

thrashing A condition, caused by a high level of memory over–commitment, in which the system is spending all of its time writing out virtual–memory pages and reading them back in. The application programs make no progress because their pages don't stay in memory long enough to be used. Memory load control is intended to avoid or stop thrashing.

thread The dispatchable entity in AIX Version 4. Each thread represents the current execution state of a single instance of a program. Each user thread runs in the environment provided by a specific process, but multiple threads may share the resources owned by that process.

thread–serial service
A reentrant system service is thread–serial if it blocks the current thread and all other threads that attempt to call the same service or other related services until the first call returns. See also *reentrant service* on page 2-197.

thread–synchronous service
A reentrant system service is thread–synchronous if it blocks only the current thread and allows other threads to execute the same operation during the block. See also *reentrant service* on page 2-197.

threaded application
An application that performs its function by simultaneously using multiple execution paths (threads of control) within a single address space.

threshold (1) A logic operator with the property that if *P* is a statement, *Q* is a statement, *R* is a statement, and so on, then the threshold of *P*, *Q*, *R*, and so on, is true if at least *N* statements are true, and false if less than *N* statements are true. *N* is a specified nonnegative integer called the threshold condition.
(2) In computer graphics, a level above which all gray–scale image data can be represented as white and below which all gray–scale image data can be represented as black.

throughput The *number* of workload operations that can be accomplished per unit of time.

throughput–class negotiation
In X.25 communications, an optional facility that allows a DTE to negotiate the speed at which its packets travel through the packet switching network. See also *optional facilities* on page 2-162.

throw expression
An argument to the exception being thrown.

TIC Transfer in channel.

ticket An application–transparent mechanism that transmits the identity of an initiating principal to its target. A simple ticket contains the principal's identity, a session key, a timestamp, and other information, sealed using the target's secret key. A privilege ticket contains the same information as a simple ticket, and also includes a privilege attribute certificate. A ticket–granting ticket is a ticket to the ticket–granting service; a service ticket is a ticket for a specified service other than the ticket–granting service.

TIDL Transactional Interface Definition Language.

tie–down resistor
A resistor used to hold the input to a logic function or gate to the low level. One end of the resistor is connected to ground or the appropriate negative voltage.

tie–up resistor A resistor used to hold the input to a logic function or gate to the high level. One end of the resistor is connected to the appropriate positive voltage.

TIFF See *Tagged Input Format File* on page 2-237.

tile. (1) A pixmap.
(2) To fill a region with a pixmap.
(3) To replicate a pixmap in two dimensions.

time slice The interval between scheduled checks by the CPU scheduler to see if a different thread should be dispatched. Unscheduled checks may occur as a result of interrupts or system calls.

time stamp (1) A time value expressed in milliseconds, typically since the last server reset. Time–stamp values wrap around usually after 49.7 days. The server, once given its current time, is represented by *timestamp T* and always interprets time stamps from clients by treating half of the time–stamp space as being earlier in time than *T*, and half the time–stamp space as being later in time than *T*. One *timestamp* value, represented by the constant CurrentTime, is never generated by the server. This value is reserved for use in requests to represent the current server time.
(2) Information added to a record or other form of data that shows the date and time at which a computer processed that record or data.
(3) An integer that describes the date and time at which a set of licenses was created. (4.) Records the date and time when an Ada–language compilation unit was last modified.

TIMED Timed Server Protocol. A protocol used to synchronize a host's time with the time of other hosts.

timeslicing A mechanism by which running threads are preempted at fixed intervals. This ensures that every thread is allowed time to execute.

timing loop A loop placed around code whose performance is to be timed, such that the timing loop increases the number of executions of the code within it to a time that can provide a meaningful performance measurement.

title bar The bar that appears across the top of a window and that contains a file name or title. It can be used for moving or activating the window.

TLB See *translation lookaside buffer* on page 2-245.

TLB miss A memory delay that occurs when a memory location is referenced and the page that contains that memory location does not have an entry in the appropriate transition lookaside buffer (instruction or data).

TLI Transport Layer Interface.

TOD Time of day.

toggle (1) A switching device such as a toggle key on a keyboard.
(2) Pertaining to any device having two stable states.
(3) To switch between two modes on a computer or network.

toggle button In AIXwindows and Enhanced X–Windows, a graphical object that simulates a real–world toggle switch; it switches sequentially from one optional state to another.

token. (1) The smallest independent unit of meaning of a program as defined by either the parser or the lexical analyzer. A token can contain data, a language keyword, an identifier, or other parts of a language syntax.
(2) In the **m4** command, any string of letters and digits that the **m4** command recognizes.
(3) A type of macro that the typesetting preprocessor replaces with an assigned string value. See also *string register* on page 2-228.
(4) In a local area network, the symbol of authority passed among data stations to indicate the station temporarily in control of the transmission medium.

token numbers Nonnegative integers that represent the names of tokens.

token ring A type of local area network that was developed under the auspices of the IEEE 802.5 Subcommittee. A token access procedure used with a sequential (ring) topology.

Toolkit In AIXwindows and Enhanced X–Windows, a collection of C language data structures and subroutines that collectively expedite the development of graphical user interfaces for compatible applications written in C language.

top–down An approach to problem solving that starts at the highest level of abstraction and proceeds toward the lowest level.

top–level In AIXwindows and Enhanced X–Windows, pertaining to the widget classes at the top level of a widget–gadget hierarchy.

top–level transaction
A transaction that does not execute within the scope of another transaction. In other words, a top–level transaction is the root of a transaction family, even if it is the only transaction in the family tree.

top–level widget
In AIXwindows and Enhanced X–Windows, widget classes that are at or near the top level of the object class hierarchy, which is known as the Core class.

top–level window
In AIXwindows and Enhanced X–Windows, the main window that contains all other windows associated with a client application.

top shadow In AIXwindows, an arrow band of lighter color across the top of a rectangular graphical object (a widget or gadget) that creates a three–dimensional appearance when the object is manipulated.

topic tree In CDE, in a general help dialog box, an expandable and contractible list of topics that can be selected to display help information.

topological sort
A sorting file that sorts an unordered list of ordered pairs.

tower A set of physical address and protocol information for a particular server. CDS uses this information to locate the system on which a server resides and to determine which protocols are available at the server. Tower values are contained in the **CDS_Towers** attribute associated with the object entry that represents the server in the cell namespace.

TPN See *transaction program name* on page 2-244.

trace (1) To record data that provides a history of events occurring in the system.
(2) A record of the running of a computer program. It exhibits the sequences in which the instructions were run.
(3) To monitor system performance or aid in debugging programs.

trace daemon Reads from the trace device driver and writes to the trace log file.

trace entry Data recorded from a trace event.

trace ID A unique identifier for a traced event.

trace log A file where trace events are recorded.

trace table A storage area that contains a record of the performance of computer program instructions.

trace template Used by the trace formatter to determine how the data contained in trace entries should be formatted.

traceback For the Ada debugger, a listing of the routines that are in the call chain above the code you are debugging. For example, if you set a breakpoint within an Ada procedure and request a call traceback, you see a list of all the procedures that called your Ada procedure, in the order in which they called it. All the calling procedures in the call chain are listed up to, but not including, the operating system calling the original highest level routine.

track (1) A circular path on the surface of a fixed disk or diskette on which information is magnetically recorded and from which recorded information is read.
(2) The path on a data medium associated with a single reading or writing component as the data medium moves past the component.

trailer The portion of a message that contains control information. Trailers are used by the VAX Trailer Encapsulation Protocol. Trailer encapsulation allows the receiving host to receive data on a page–aligned boundary, which is a requirement for utilizing a page–mapped virtual memory environment.

trailer page A trailer page that follows a printed file or a print job.

TRAN Distributed Transaction Service.

transaction (1) An exchange between a workstation and a program, two workstations, or two programs that accomplish a particular action or result. Some examples are the entry of a customer's deposit and the updating of the customer's balance.
(2) In a batch or remote batch entry, a job or job step.

transaction family
Nested transactions that have a common ancestor belong to the same transaction family. All members of a transaction family commit together and drop their locks simultaneously.

transaction ID See *transaction identifier* on page 2-244.

transaction identifier
A unique identifier assigned to each transaction, used to identify all actions associated with that transaction.

transaction program
A program that processes transactions in an SNA network. The two kinds of transaction programs are application transaction programs and service transaction programs. See also *conversion* on page 2-54,

application transaction program on page 2-10, and *service transaction program* on page 2-216.

transaction program name (TPN)
The name of an application program that uses data communications to send or receive data to or from another application program.

transactional RPC
transactional remote procedure call.

transcript In remote communications, a file that contains the written record of commands you enter on the remote system and the remote system's response to those commands.

transfer To send data to one place and to receive data at another place.

transformation (1) In GL, a four–by–four matrix that helps determine the location where three–dimensional drawing will occur, the position of the viewpoint (the viewer's "eye"), and the amount of the scene encompassed and visible. Transformations occur at four points within the graphics pipeline: (1.) Modeling transformation, which maps modeling coordinates into world coordinates. All drawing primitives specify positions that are presumed to be positions in modeling coordinates. Modeling transformation can be used to move the thing being drawn
(2) Viewing transformation, which maps from world coordinates to viewer coordinates. The origin of the viewer coordinate system can be thought of as the location of the viewer's "eye," and viewing transformations can be used to move the "eye" around in world coordinates.
(3) Projection transformation, which defines the boundaries of the clipping region. A projection transformation maps viewer coordinates to normalized device coordinates, and the clipping plane boundaries are at $x = +/- w$, $y = +/- w$, $z = +/- w$. Projection transformations can be used to define what region of the world is visible on the screen.
(4) Viewport, or NDC to DC transformation. The viewport transformation is not a full–fledged four–by–four transformation matrix; only three of the diagonal elements in the matrix can be changed. The viewport determines the mapping from normalized device coordinates to screen (device) coordinates. By default viewports are the same size as the window, although this can be adjusted. See also *normalized device coordinates* on page 2-155, *modeling coordinates* on page 2-146, *rotation* on page 2-207, *eye coordinates* on page 2-88, *screen coordinates* on page 2-210, *selecting region* on page 2-213, *viewing matrix* on page 2-257, *world coordinates* on page 2-265, *clipping* on page 2-38, and *current transformation matrix* on page 2-42.

transient A program or subroutine that does not reside in main storage.

transit delay In X.25 communications, the time it takes a packet to travel from one DTE to the other.

translation (1) The movement of something by a certain distance.
(2) In computer graphics, the moving of a display image in a straight line from one location to another. See also *transformation* on page 2-245.

translation lookaside buffer (TLB)
A table in the CPU that contains cross–references between the virtual and real addresses of recently referenced pages of memory.

translation table
(1) A table that specifies the mapping of events or event sequences to procedure names.
(2) A string containing a list translating the events to procedure calls.

translation table list
See *translation table* on page 2-245.

translations Action procedures that are called for an event or sequence of events.

transmission control characters
Special characters included in a message to control communication over a data link. For example, the sending station and the receiving station use transmission control characters to exchange information. The receiving station uses transmission control characters to indicate errors in data it receives.

Transmission Control Protocol (TCP)
A communications protocol used in ARPA Internet and any network following the U.S. Department of Defense standards for inter–network protocol. Provides a reliable host–to–host protocol between hosts in packet–switched communications networks and in interconnected systems of such networks. It assumes that the Internet Protocol is the underlying protocol.

Transmission Control Protocol/Internet Protocol (TCP/IP)
An industry–standard, nonproprietary communication protocol suite that allows connectivity between equipment from different manufacturers. Its development was funded by the Department of Defense Advanced Research Projects Agency. A communications subsystem that allows you to set up local area and wide area networks.

transmission services (TS)
In SNA, a specification in a session activation request for transmission control (TC) protocols to be supported by a particular session (such as session–level pacing the usage of session–level requests). Each transmission services request defined is identified by a number.

transmit burst A group of transmit packets that are sent without an intervening receive or time–out operation.

transparent (1) In communications, pertaining to transmissions that cannot interfere with data link control, regardless of format or content. Transparent transmissions are unrecognized by data link controls.
(2) In data transmission, pertaining to information that the receiving program or device does not recognize as transmission control characters.
(3) Pertaining to operations or data that are of no significance to the user.

transparent binding
A form of binding that allows the client application to bind to any available server exporting the desired interface.

transparent mode
A method of binary synchronous text transmission in which only transmission control characters preceded by the data link escape(DLE) character are processed as transmission control characters.

Transport Layer
A network service that provides end–to–end communications between two parties, while hiding the details of the communications network. The TCP and ISO TP4 transport protocols provide full–duplex virtual circuits on which delivery is reliable, error free, sequenced, and duplicate free. UDP provides no guarantees (the connectionless RPC protocol provides some guarantees on top of UDP).

transport protocol
A communications protocol from the Transport Layer of the OSI network architecture, such as the Transmission Control Protocol (TCP) or the User Datagram Protocol (UDP).

trap An unprogrammed, hardware–initiated, conditional jump to a specific address. It occurs as a result of an error or certain other conditions. A record is made of the location from which the jump occurred.

trap handler A user–defined trap routine used when a trap occurs. See also *exception* on page 2-84.

Trash Can In CDE, a container for deleted files or folders.

traversal See *keyboard traversal* on page 2-125.

tree delta A SCCS file that has a trunk, with changes identified by a release and level, and with one or more branches with deltas identified by an SID.

tree structure A hierarchical calling sequence that consists of both a root segment and one or more levels of the segments called by way of the root segment.

tree view In CDE, a view of a folder or files that includes all lower–level folders in the search path.

triangular array
A two–dimensional array in which active elements are found only on one side of a hypothetical axis delineated by the first and last elements of the array.

trimming loops
In GL, a set of oriented closed curves used to set the boundaries of a NURBS surface. See also *NURBS* on page 2-157.

Trivial File Transfer Protocol
Transfers files between hosts using minimal protocol.

troff A phototypesetting utility originally designed to support a Graphics Systems phototypesetting machine, but now capable of supporting a variety of phototypesetters.

true color In Enhanced X–Windows,

1. A degenerate case of direct color in which the subfields in the pixel value directly encode the corresponding RGB values. That is, the color map has predefined read–only RGB values. The values are typically near–linear increasing ramps.
2. Also TrueColor, a value.
3. Refers to the use of 24–bits per pixel direct RGB, where there are 8 bits (256 levels) of red, 8 bits (256 levels) of green, and 8 bits (256 levels) of blue, for a total of 256*256*256 or 16,777,216 different combinations of red, green, and blue intensities. The number of different colored pixels that can then be simultaneously displayed is only limited by the number of pixels displayable from the frame buffer.

truncate (1) To end a computational process in accordance with some rule; for example, to end the evaluation of a power series at a specified term.
(2) To remove the beginning or ending elements of a string.
(3) To drop data that cannot be printed or displayed in the line width specified or available. Contrast with *fold* on page 2-93.
(4) To shorten a field or statement to a specified length.

trust peer A characterization of one cell with respect to another with which the cell maintains a mutual authentication surrogate.

trusted computing base (TCB)
The part of the system that contains all the elements that support the security policy of the system. The trusted computing base includes all hardware, microcode, and software that protect information on the system.

trusted device An IPL device such as a fixed disk or diskette drive, where the IPL ROM code can find an IPL record and IPL code.

trusted environment
A clean environment in which all untrusted processes have been killed order to ensure security for communications between the user and the operating system.

trusted path See *trusted environment* on page 2-247.

trusted process
A process in which a particular standard of security has been met.

try block A C++ block in which a known exception is passed to a handler.

TS See *transmission services* on page 2-246.

TSO Time Sharing Option.

tty In the operating system, any device that uses the **termio** standard terminal device interface. *tty* devices typically perform input and output on a character–by–character basis.

tty device See *tty* on page 2-248.

tuning See *hand tuning* on page 2-105.

turnaround Changing a communications line from transmit mode to receive mode, or from receive mode to transmit mode.

twist A rotation around the line of sight.

two and one–half dimension
See *2–1/2 dimension* on page 1-1.

two–way channel
In X.25 communications, a logical channel that allows both incoming and outgoing calls. Contrast with *one–way channel* on page 2-161.

twos complement
The radix complement in the pure binary numeration system. The twos complement is derived by taking the ones complement and then adding one to the resulting number. In a twos complement system, the twos complement of a number *n* is – *n*. See also *ones complement* on page 2-161.

type (1) In Enhanced X–Windows, an arbitrary atom used to identify the data. A type is solely for the benefit of clients and is not interpreted by the server. Enhanced X-Windows predefines type atoms for many frequently used types. Clients also can define new types.
(2) In Pascal, see *data type* on page 2-62.
(3) In NCS, a class of object. All objects of a specific type can be accessed though the same interface or interfaces.
(4) In Ada language, a type characterizes both a set of values, and a set of operations applicable to those values. A type definition is a language construct that defines a type. A particular type is either an access type, an array type, a private type, a record type, a scalar type, or a task type.
(5) In XOM, a category into which attribute values are placed on the basis of their purpose.

type compatibility
See *compatible types* on page 2-44.

type conversion
A routine or set of routines that enables an application to change a specified string of data from one declared type to another. In AIXwindows programming, type conversion is performed on strings using conversion information contained in the MRM database.

type declaration
The specification of the type and, optionally, the length of a variable or function in a specification statement.

type definition A definition of a name for a data type.

type identifier The name given to a declared type. See also *type specifier* on page 2-249.

type regrouping
An optimization that involves reodering mixed–type expressions so that all variables of a given type are grouped together.

type specifier A name of a data type. See also *type identifier* on page 2-248.

type style The form of characters of a given size, style, and design within the set of the same font.

type UUID (1) A UUID that permanently identifies a particular type. Both the RPC run time library and the Location Broker use type UUIDs to specify types.
(2) In DCE RPC, the UUID that identifies a particular type of object and an associated manager. See also *object* on page 2-158 and *Universal Unique Identifier UUID* on page 2-251.

typematic key A key that, when held down, repeats its function multiple times.

U

UA See *unnumbered acknowledge* on page 2-252.

UART See *Universal Asynchronous Receive/Transmit* on page 2-251.

ublock See *user block* on page 2-253.

UCT See *Universal Coordinated Time* on page 2-251.

UDP See *User Datagram Protocol* on page 2-253.

UDP/IP User Datagram Protocol/Internet Protocol.

UFS See *UNIX File System* on page 2-251.

UI See *unnumbered information frame* on page 2-252.

UID See *user number* on page 2-254 and AIXwindows *User Interface Definition* on page 2-7.

UIL See AIXwindows *User Interface Language* on page 2-7.

ultimate consumer
The target of data in an input and output operation. An ultimate consumer can be a file, a device, or an array of bytes in memory.

ultimate producer
The source of data in an input and output operation. An ultimate producer can be a file, a device, or an array of bytes in memory.

umask The file–mode creation mask. The default permissions that are set automatically when a file is created. These defaults can be changed by including an appropriate **umask** command in the system profile.

unary expression
An expression that contains one operand.

unary operator An operator that represents an operation on one operand. Contrast with *binary operator* on page 2-20.

unblocked In an Enterprise Systems Connection Director, the attribute that, when set, establishes communications capability for a specific port. Contrast with *block* on page 2-19.

unconditional branch
A branch that is taken every time it is encountered.

unconfigure (1) Indicates that a user is taking a device from the available (configured) state to the defined state. This is accomplished by running the unconfigure method for a device. The device status field in the Customized Devices Object Class would reflect this action.
(2) To take out of use by the current computer system.

undefine (1) Indicates that a user is taking a device instance out of the system. This is accomplished by running the undefine method for the device. All information for the device in the Customized Database is purged by this operation.
(2) To cause a command to no longer recognized by the current computer system.

underlying editor
A large editor program, such as vi, of which another editor program, such as vedit, is a *limited subset*.

unescaped In an expression, a character that is not preceded by an escape sequence and is therefore interpreted as a control character. See also *escape sequence* on page 2-83.

unformatted file
A file displayed with data that is not arranged with particular characters.

unidirectional printing
A printing method in which the print head on the printer prints only while it moves in one direction, instead of also printing while it moves in the opposite direction. This method of printing usually produces higher–quality print output.

union A variable that can hold any one of several data types, but only one data type at a time.

union tag The identifier that names a union data type.

uniprocessor A system containing a single processor. As used in this book, the phrase ”comparable uniprocessor” means a system designed to have only a single processor, with the same CPU–clock speed and cache capacity as the SMP system being discussed, running a uniprocessor version of the operating system. Contrast with single–processor SMP system on page 2-220.

unit (1) In FORTRAN, a means of referring to a file to use input/output statements. A unit can be connected or not connected to a file. If connected, it refers to the file. The connection is symmetric; that is, if a unit is connected to a file, the file is connected to the unit.
(2) See also *compilable unit* on page 2-44.
(3) For Ada programming, synonym for *compilation unit* on page 2-45.

unit cube In GL, a volume defined by the following planes: $x = -1$, $x = 1$, $y = -1$, $y = 1$, $z = -1$, $z = 1$. See also *normalized device coordinates* on page 2-155.

unit identifier In FORTRAN, the number that specifies an external unit or internal file. The number can be one of the following: an integer expression whose value must be zero or positive, an * (asterisk) that corresponds to unit 5 for input or unit 6 for output, or the name of a character array, character array element, or character substring for an internal file.

Universal Asynchronous Receive/Transmit
A circuit used in asynchronous data communication applications to provide all the necessary logic to recover data in a serial–in parallel–out fashion and to transmit data in a parallel–in serial–out fashion. It is usually full–duplex, that is, it can transmit and receive simultaneously with the option to handle various data work length.

Universal Coordinated Time (UCT)
The new standard term for worldwide time–telling that has the same meaning as Greenwich Mean Time.

Universal Unique Identifier (UUID)
A 128–bit value used for identification. NCS uses UUIDs to identify cells, interfaces, objects, and types. The UUID for a cell, generated by the **uuid_gen** tool, is completely unique, having been created based on the unique system ID of the workstation and a time stamp. Once generated, the cell UUID is placed in the **glb_obj.txt** file. See also *object UUID* on page 2-160 and *type UUID* on page 2-249.

UNIX File System (UFS)
A section of the UNIX file tree that is physically contained on a single device or disk partition and that can be separately mounted, dismounted, and administered.

UNIX–to–UNIX Copy Program (UUCP)
(1) A group of commands, programs, and files, present on most UNIX systems, that allows the user to communicate with another UNIX system over a dedicated line or a telephone line. See also *Basic Networking Utilities* on page 2-19.
(2) The command (**uucp**) that starts file copying from one or more sources to a single destination.

unmanaged widget
A widget whose size cannot be changed.

unmapped window
A window that is not visible on the screen.

unmarshal In NCS, to copy data from a Remote Procedure Call packet. Stubs perform unmarshalling. Contrast with *marshal* on page 2-141.

unnumbered acknowledge (UA)
A link control frame.

unnumbered information (UI) frame
A frame in unnumbered format, used to transfer unnumbered control functions.

unpredictable A violation of an architecture rule that an implementation is not required to report. Results can include an error report from a threads call, the operating system, or the hardware; a hang or deadlock of the program; or an incorrect operation of the program without indication of error. See also *illegal* on page 2-111.

unprivileged state
A hardware protection state in which the processor can only run unprivileged instructions. The processor's unprivileged state supports the virtual machine's operating system state and problem state. Contrast with *privileged state* on page 2-185.

unreachable code
Code that cannot be reached during program execution. Unreachable code is detected and removed as part of optimization.

unused subprograms
In Ada language, subprograms unused only within the context of a specific program or set of units. For example, a program might call only a small subset of subprograms in a utility package. The remaining uncalled (unreachable) subprograms constitute unreachable code in the context of that program.

unviewable Pertaining to a mapped window with an unmapped ancestor.

update (1) The procedure of modifying a program or program option that exists on the mass storage medium of a computer, making the program executable, and ensuring that the modified program interacts properly with all other affected programs in the system.
(2) An improvement for some part of the system.
(3) To add, change, or delete items.
(4) To modify a master file with current information according to a specified procedure.

update propagation
An immediate attempt to apply a change to all replicas of the CDS directory in which the change was just made. An update propagation delivers changes in a more efficient and timely way than a skulk, which is the periodic distribution of a whole collection of changes.

update script A shell script or executable file created by the developer of an application program to update the program. The script file must follow specific guidelines to be compatible with the program update tools that are provided in the operating system.

Update Timestamp (UTS)
An attribute that identifies the time at which the most recent change was made to any attribute of a particular CDS name. For directories, the UTS reflects changes made only to attributes that apply to the directory as a whole (not one of its replicas).

upgrade Software that fixes a defect in a previously released software product.

upgrade locks Locks used instead of read locks that announce the potential need to also modify the protected data. If an application obtains a read lock and possibly a write lock to the same data, an upgrade lock is less likely to deadlock.

upload To transfer data from one computer to another. Typically, users upload from a small computer to a larger one.

upstream The direction from driver to stream head.

use clause In Ada language, a clause that achieves direct visibility of declarations that appear in the visible parts of named packages.

use–once license
In License Use Management, a type of license administrated by the license server that can be used for a single instance of invoking a product or of using a service. The license server decrements the number of use–once licenses each time the product is used.

user (1) The name associated with an account.
(2) Anyone requiring the services of a computing system.

user account See *account* on page 2-4.

user address list
The address list that an individual can use with the **xtalk** command to make outgoing X.25 calls. See also *address list* on page 2-6 and *system address list* on page 2-234.

user area The parts of main storage and disk available to the user.

user block A data structure maintained by the kernel that contains system information about a user process, such as its real and effective user IDs, the list of open file descriptors, and signal–handling settings. The **user** structure (defined in the **/usr/include/sys/user.h** header file) specifies the exact information that is kept in the user block. See also *per–process data area* on page 2-173.

User Datagram Protocol (UDP)
A packet–level protocol built directly on the Internet Protocol layer. UPD uses application–to–application programs between host systems.

user data segment
In kernel mode, the virtual memory segment that contains user data, which consists of initialized data variables.

user–defined variable
A shell variable to which the user assigns a character string as a value.

user file A text file that specifies the users who may (or may not) use licensed software products.

user ID See *user identification* on page 2-254.

user identification (user ID)
(1) One to eight alphanumeric characters, beginning with an alphabetic, #, $, or > character, that identifies a user. This string of characters limits the functions and information the operator can use. Often, the user ID can be substituted in commands that take a user's login name as a value. See also *user number* on page 2-254.
(2) A parameter that specifies the user ID under which the application or transaction program runs. Contrast with *user name* on page 2-254.

user interface The hardware, software, or both by which a user communicates with a system, program, or device. Examples are a keyboard, mouse, command language, or windowing subsystem.

user mode A mode in which a process is carried out in the user's program rather than in the kernel. Contrast with *kernel mode* on page 2-124.

user name A string of characters that uniquely identifies a user to the system. Contrast with *user identification* on page 2-254.

user number (UID)
A number that uniquely identifies a user to the system. It is the internal number associated with a user ID. See also *user identification (user ID)* on page 2-254.

user profile A file in the user's home directory named**.profile** that contains shell commands that set initial user–defined characteristics and defaults for the login session.

user space The address space seen by a process in user mode. See also *user structure* on page 2-254.

user structure In kernel mode, the data area that contains information that must be accessible while a process runs. One user structure is allocated for each active process. See also *per–process data area* on page 2-173 and *user block* on page 2-253.

user time The amount of time a program is running in the CPU. Does not include time associated with operating system services provided to the program, the program's I/O time, or time in which other processes preempt the program's use of the CPU.

USOC–RJ11 A miniature telephone jack.

UTC See *Universal Coordinated Time* on page 2-251.

utility (1) A service. In programming, a program that performs a common service function.
(2) The capability of a system, program, or device to perform the functions for which it is designed.

UTS See *Update Timestamp* on page 2-253.

UUCP See *UNIX–to–UNIX Copy Program* on page 2-252.

UUCP login ID A login name, provided with the Basic Networking Utilities (BNU), that has complete access to all BNU files and directories. See also *Basic Networking Utilities* on page 2-19.

UUID See *Universal Unique Identifier* on page 2-251.

V

V Volt.

V.24 The 24th CCITT recommendation in the V series, listing the definitions for interchange circuits between a DTE and a DCE.

V.35 The 35th CCITT recommendation in the V series, defining data transmission at 48 kilobits per second using 60–180 kHz group band circuits.

v–node Virtual i–node. An object in a file system that represents a file. Unlike an i–node, there is no one–to–one correspondence between a v–node and the file system; multiple v–nodes can refer to a single file (a single i–node). V–nodes are used to communicate between the upper half of the file system (the logical file system) and the file system implementations (such as the journaled file system or the network file system).

valid (1) Allowed.
(2) True, or conforming to an appropriate standard or authority.

validation In X.25 communications, the process by which the receiving DTE accepts the packet size, packet window size, and throughput class sent by the sending DTE, on the conditions that they are valid. Contrast with *negotiation* on page 2-151.

valuator (1) An input device that provides a scale value; for example, a thumb wheel or a potentiometer.
(2) In GL, an input/output device that returns a value in a range. For example, a mouse is logically two valuators: the *x* position and the *y* position. See also *dial* on page 2-69.

value (1) A set of characters or a quantity associated with a parameter or name.
(2) In programming, the contents of a variable or a storage location.
(3) A specific occurrence of an attribute, such as blue for the attribute color.
(4) A quantity assigned to a constant, variable, parameter, or symbol.
(5) In XOM, an arbitrarily complex information item that can be viewed as a characteristic or property of an object. See also *attribute value* on page 2-14.

variable (1) A name used to represent a data item whose value can change while the program is running. Contrast with *constant* on page 2-51.
(2) In programming languages, a language object that can take different values at different times.
(3) A quantity that can assume any of a given set of values.
(4) For Ada programming, see *object* on page 2-158.

variable length field
A field of varying length that contains data prefaced by an internal, opaque field providing the length of the field.

variable substitution
The ability to change and display the values of variables in a string, replacing a name of a variable with the value it represents.

variant part In Pascal, the part of a record that can vary from one instance of the record to another. The variant part consists of alternate sequences of fields that share the same physical storage. In Ada language, a variant part of a record specifies alternative record components, depending on a discriminant of the record. Each value of the discriminant establishes a particular alternative of the variant part. See also *tag field* on page 2-237.

VC See *virtual circuit* on page 2-258.

vchar In ODM, a terminal descriptor type used to define a variable as a variable–length, null–terminated string. See also *terminal descriptor* on page 2-239.

VCI Virtual channel identifier in an ATM network.

VDD See *virtual device driver* on page 2-257.

vector (1) An array of one dimension.
(2) A quantity usually characterized by an ordered set of numbers.
(3) In computer graphics, a directed line segment.

vector graphics
(1) Graphics for which the display images are generated from coordinates, as opposed to an array of pixels.
(2) The most common class of graphics, where all vector output consists of lines and curves drawn point–to–point by the output unit as ordered by the computer.

vector product Another term for the vector cross product. If a = (a1, a2, a3) and b = (b1, b2, b3) are two three–dimensional vectors, the vector product a times b = (a2b3–b2a3, a3b1–b3a1, a1b2–b1a2).

vendor ID In License Use Management, the identifier of a vendor of licensed products. By means of vendor IDs, license servers can distinguish among any number of vendors established in a network. Vendor IDs are an License Use Management–specific usage of Network Computing System Universal Unique Identifiers (UUIDs).

vendor password
In License Use Management, a string encoded with information about a vendor that, together with a vendor ID, establishes the vendor of a licensed product in a license database.

verify (1) To confirm the correctness.
(2) To determine whether a transcription of data or other operation has been accomplished accurately.
(3) In software installation, the verify procedure instructs the system to verify the software you are installing. The system confirms that your software files are the correct length and contain the correct number of digits and characters. If any errors are reported, it might be necessary to install the software product again. The verification process can add a significant amount of time to the installation process.

Versatile Message Transfer Protocol (VMTP)
A protocol that provides datagram communication service at the user level. Unlike most programs that use UDP/IP, programs using VMTP do not have to implement time out, retransmission, or estimation of network delays because the VMTP protocol provides end–to–end datagram delivery.

version A particular instance of an application program or licensed program. The version, release, modification, and fix levels of a program. The version, release, modification, and fix levels together comprise the program level or version of a program. A new version of a program usually has significant new code or functions compared to the previous version. See also *fix number* on page 2-92, *program level* on page 2-187, *modification number* on page 2-141, *release number* on page 2-199, and *version number* on page 2-257.

version control
The process by which the contents of each revision of software, hardware, or documentation are accounted for.

version identifier
In License Use Management, a string that identifies a version of a product; by means of version identifiers, the license server distinguishes among different versions of a product.

version number
The version level of a program, which is an indicator of the hardware and basic operating system upon which the program operates. The version, release, modification, and fix levels together comprise the program level or version of a program. See also *fix number* on page 2-92, *modification number* on page 2-141, *release number* on page 2-199, *version* on page 2-256 and *program level* on page 2-187.

vertical retrace (1) The rate at which the monitor is refreshed. A 60 Hz monitor is redrawn 60 times per second. Synonymous with *refresh rate*.
(2) The action of moving the electron beam from the bottom to the top of the screen.

vertical retrace period
The amount of elapsed time between retraces of the screen. All video monitors use an electron beam to sweep the phosphors at the face of the monitor. Because the phosphors glow for only a brief period of time, the entire screen must be reswept periodically by the electron beam. On most monitors, this is done 30 times per second (30 Hz). Thus, the vertical retrace period is 1/30 second.

vhandle An identifier for each object in VEOS. vhandles are used in graphics programming to identify software objects specified in VEOS. The vhandle can exist longer than the object that it identifies. After a software object such as a swidget is destroyed, the vhandle for that object remains defined.

video lookup table (VLT)
A colormap implemented in hardware.

viewable Pertaining to a mapped window whose ancestors are all mapped; not necessarily visible. Graphics requests can be performed on a window when it is not viewable, but output will not be retained unless the server is maintaining backing store.

viewer coordinates
Synonym for *eye coordinates* on page 2-88.

viewing coordinates
Synonym for *eye coordinates* on page 2-88.

viewing matrix In GL, a matrix used to describe the location of the viewer (the virtual eye looking upon a scene) in relation to the world. See also *field of view* on page 2-89, *transformation* on page 2-245, and *world coordinates* on page 2-265.

viewport In GL, the mapping from normalized device coordinates to device coordinates. The viewport maps the unit cube $x / w = +/-1$, $y / w = +/-1$, $z / w = +/-1$ to the screen space, as measured in pixels. The viewport is the last transformation in the graphics pipeline. The viewport can be smaller or larger than the window and smaller or larger than the screenmask, although in most applications, it is the same size.

virtual address The address of a location in virtual storage. A virtual address must be translated into a real address for the data or instruction it addresses to be accessible to the CPU.

virtual call facility
In data communications, a user facility in which a call–setup procedure and a call–clearing procedure determine a period of communication between two data terminal equipment (DTEs) in which user data is transferred in the network in the packet mode of operation. All user data is delivered from the network in the order it is received by the network. It is the packet network equivalent of a dialed line.

virtual circuit (VC)
In X.25 communications, those facilities provided by a network that give the appearance to the user of an actual connection. See also *switched virtual circuit* on page 2-218 and *permanent virtual circuit* on page 2-173.

virtual device A device that appears to the user as a separate entity, but is actually a shared portion of a real device.

virtual device driver (VDD)
Synonym for *device handler* on page 2-68.

virtual file system (VFS)
A remote file system that has been mounted so that it is accessible to the local user.

virtual function
A C++ member function that is declared with the keyword **virtual**. The implementation that is executed when you make a call to a virtual function depends on the type of the object for which it is called. This is determined at run time.

virtual key binding
In AIXwindows, the user–designed key definitions.

virtual memory Addressable space that appears to be real storage. From virtual storage, instructions and data are mapped into real storage locations. The size of virtual storage is limited by the addressing scheme of the computer system and by the amount of auxiliary storage available, not by the actual number of system memory locations. Contrast with *real memory* on page 2-195. Synonymous with *virtual storage*.

virtual mount point
The directory or file in the file tree where another file system is mounted. For example, if **/dev/hd9** is mounted on `/fred`, then `/fred` is the virtual mount point.

virtual printer A view of a printer that refers only to the high–level data stream (such as ASCII or PostScript) that the printer understands. It does not include any information about how the printer hardware is attached to the host computer or the protocol used for transferring bytes of data to and from the printer.

virtual printer definition
A set of attributes values that describe a particular data stream for a particular printer.

virtual storage Synonym for *virtual memory* on page 2-258.

Virtual Storage Extended (VSE)
An operating system that is an extension of DOS. A VSE system consists of licensed VSE/Advanced Functions support plus all programs required to meet the data processing needs of the user. Together with the hardware it controls, VSE forms a complete data processing system. Its current version is called VSE/ESA.

Virtual Storage Extended/Enterprise Systems Architecture(VSE/ESA)
The VSE operating system operating in an ESA environment. ESA is an extension to the System/370 architecture and includes an advanced addressability feature that provides access registers.

Virtual Storage Extended/System Product (VSE/SP)
A licensed program providing VSE operating system support.

Virtual Telecommunication Access Method (VTAM)
A licensed program that controls communication and data flow in an SNA network. It provides single–domain, multiple–domain, and interconnected network capability, and also supports application programs and subsystems.

visibility In Ada language, at a given point in a program text, the declaration of an entity with a certain identifier is said to be visible if the entity is an acceptable meaning for an occurrence at that point of the identifier. The declaration is visible by selection at the place of the selector in a selected component or at the place of the name in a named association. Otherwise, the declaration is directly visible, that is, if the identifier alone has that meaning.

visible (1) A region of a window that is mapped and not occluded on the screen by another window.
(2) Visibility of C++ identifiers is based on scoping rules and is independent of access.

visible part For Ada programming, see *package* on page 2-166.

VLSI Very large scale integration.

VLSI circuit Very large scale integrated circuit.

VLT See *video lookup table* on page 2-257.

VM Virtual machine.

VM/CMS A type of operating system used on a System/370 computer.

VMTP See *Versatile Message Transaction Protocol* on page 2-256.

VOL See *Volume Service* on page 2-260.

Vol ID See *Volume ID* on page 2-260.

volatile attribute
The keyword **volatile** located in a definition, declaration, or cast. It causes the C language compiler to place the value of the data object in storage and to reload this value at each reference to the data object.

volatile register
In a C language program, a register whose value on entry need not be preserved when the called routine returns.

volume (1) A certain portion of data, together with its data carrier, that can be handled conveniently as a unit.
(2) The level of sound of the system.
(3) The physical storage location of a file system. See also *log volume* on page 2-134.

volume group (VG)
A set of one or more physical volumes from which space can be allocated to one or more logical volumes. A collection of 1 to 32 physical volumes (read–write fixed–disk drives) of varying size and type. See also *logical volume* on page 2-136.

Volume ID (Vol ID)
A series of characters, recorded on the diskette, used to identify the diskette to the user and to the system.

volume label An area on tape or disk that is used to identify the tape volume and its owner.

Volume Service (VOL)
The component of Encina Base that addresses storage.

VPD See *Software Vital Product Data (SWVPD)* on page 2-222.

VPI Virtual path identifier in an ATM network.

VRAM Video random–access memory.

VSE See *Virtual Storage Extended* on page 2-258.

VSE/ESA ee *Virtual Storage Extended/Enterprise Systems Architecture* on page 2-259.

VSE/SP See *Virtual Storage Extended/System Product* on page 2-259.

VTAM See *Virtual Telecommunication Access Method* on page 2-259.

VTL Vendor Technology Logic

W

WAN See *wide area network* on page 2-261.

WAN links Communications connections between groups of computers that are spread across a large geographical distance. Modem connections, T1 lines, and satellite hookups are some common examples. See also *wide area network* on page 2-261.

water mark A limit value used in flow control. Each queue has a high–water mark and a low–water mark. The high–water mark value indicates the upper limit related to the number of bytes contained on the queue. When the characters in a queue reach the high–water mark, STREAMS causes another queue that attempts to send a message to this queue to become blocked. When the characters in this queue are reduced to the low–water mark value, the other queue is unblocked by STREAMS.

Web-based System Manager
A graphical user interface (GUI) tool for managing the operating system. Based on the OO (Object Oriented) model, Web-based System Manager enables users to perform administration tasks by manipulating icons representing objects in the system, as an alternative to learning and remembering complex commands.

well–known host name
A conventional name associated with an Internet Protocol address on a particular network (for example, the **nameserver** and **timeserver** servers).

well–known port
A conventional port assignment used by hosts that support the same protocols, whether or not the hosts are on the same network. Synonymous with *contact port*.

while statement
A C language looping statement that contains the keyword **while** followed by an expression in parentheses (the condition) and a statement (the action).

white space Space characters, tab characters, and new–line characters.

wide area network (WAN)
A network that provides data communication capability in geographic areas larger than those serviced by local area networks. A network that includes computers spread across a large geographical distance, usually involving several cities, states, or countries.

wide band channel
A communications channel that has a greater bandwidth than a voice channel; therefore, it is capable of transmitting data at high speeds. Synonym for *broadband channel* on page 2-20.

widening An expansion of the size of a value (for example, **short** to **int**) by padding bits located to the left of the value with a copy of the sign bit.

widget (1) The fundamental data type of the Enhanced X-Windows Toolkit. (2) An object providing a user–interface abstraction; for example, **Scrollbar** widget. It is the combination of an Enhanced X-Windows (or subwindow) and its associated semantics. Logically, it is a rectangle with associated input and output semantics, although some can be input–only or output–only. Each widget belongs to one widget class. A widget implements procedures through its widget class structure. See also *composite widget* on page 2-46, *core widget* on page 2-39, *primitive widget* on page 2-184, and *shell widget* on page 2-218.
(3) A widget is a graphic device capable of receiving input from the keyboard and the mouse and communicating with an application or another widget by means of a callback. Every widget is a member of only one class and always has a window associated with it.

widget class The general group that a specific widget belongs to, otherwise known as the widget type. Physically, it is a pointer to a structure. Synonymous with *widget type*. See also *class* on page 2-36.

widget gravity Synonym for *window gravity* on page 2-263.

widget hierarchy
Synonymous with *widget tree* on page 2-262.

widget ID A unique identification number associated with each widget instantiated in an interface.

widget instance
A specific widget object as opposed to a general widget class. It is composed of a data structure containing instance–specific values and another data structure containing information applicable to all widgets of that class. See also *instance* on page 2-116.

widget menu A menu that allows the user to perform any number of actions, such as cutting, copying, and pasting, to selected widgets.

widget programmer
A programmer who adds new widgets to the Enhanced X-Windows (or other) Toolkit.

widget record A collection of related data objects, such as variables and parameters, associated with any given widget. See also *instance record* on page 2-116, *record* on page 2-196, *class record* on page 2-37, and *superclass* on page 2-232.

widget tree (1) The symbolic structure for Enhanced X-Windows Toolkit code. The basic element is a widget class. See also *leaves* on page 2-127,

intermediate nodes on page 2-119, and *root* on page 2-206.
(2) A widget tree is a hierarchy of widgets within a specific client application. The **Shell** widget is the root of the widget tree. Widgets with no children of any kind are leaves of the tree. Synonymous with *widget hierarchy*.

widget type Synonym for *widget class* on page 2-262.

widget visibility
Contrast with *obscure* on page 2-160 and *occlude* on page 2-160.

wildcard Special characters such as * (asterisk) or ? (question mark) that can be used to match one or more characters. Synonymous with *pattern–matching character*.

window (1) In AIXwindows, rectangular area of the screen that can be moved about, placed on top of or pulled under other windows, or iconized by the user.
(2) In GL, all drawing inside the window is done by the GL process that created that window, and is totally under the control of that process. However, the drawing of the window borders together with the window placement/iconization, is under the control of the window manager; for example, the AIXwindows Window Manager. For most simple GL programs, the viewport and screenmask are set to the same size as the window. Do not confuse an AIXwindows subroutine with the GL window subroutine, which defines a frustum in world space. See also *clipping* on page 2-38 and *current window* on page 2-42
(3) In curses and extended curses, the internal representation of what a portion of the display may look like at some point in time. Windows can be any size, from the entire display screen to a single character.
(4) In data communications, the number of data packets a DTE or DCE can send across a logical channel before waiting for authorization to send another data packet. The window is the main mechanism of pacing, or flow control, of packets. See also *frame window* on page 2-96 and *packet window* on page 2-167.

window gravity
The attraction of a subwindow to some part of its parent. Window gravity causes subwindows to be automatically repositioned, relative to an edge, corner, or center of a window when resized. Synonymous with *widget gravity.* See also *gravity* on page 2-103.

window icon In CDE, a minimized window.

window ID A unique identification number associated with each newly opened window in an AIXwindows or Enhanced X-Windows environment.

window manager
In Common Desktop Environment, the program that displays and controls windows on your screen. Software that manages the multiple windows associated with AIXwindows and Enhanced X–Windows. In a graphics environment, the client that manipulates windows on a screen and provides much of the user interface. See also *mwm* on page 2-149.

Window Manager
In CDE, the software application that provides users with the capability to manipulate windows on the workspace; for example, opening, resizing, moving, and closing windows.

Window menu In CDE, the menu displayed by choosing the Window menu button. The menu provides choices that manipulate the location or size of the window, such as Move, Size, Minimize, and Maximize.

Window menu button
In CDE, the control at the upper left corner of a window, next to the title bar. Choosing it displays the Window menu.

wire frame A graphics surface–drawing technique in which the edges and contours of a primitive are represented by simple lines.

with clause For Ada programming, see *compilation unit* on page 2-45.

word (1) A contiguous series of 32 bits (four bytes) in storage, addressable as a unit. The address of the first byte of a word is evenly divisible by four. Synonymous with *fullword*, *machine word*, and *computer word*. Contrast with *halfword* on page 2-105.
(2) A character string considered as a unit for a given purpose.

word wrap In word processing, a feature that automatically moves text to the next line if the text does not fit within the margins.

work area In CDE, the part of a window where controls and text appear.

work file A file used for temporary storage of data being processed.

working directory
Synonym for *current directory* on page 2-57.

working segment
A segment whose pages are backed by slots in the disk paging space rather than by a permanent location on disk.

working set The parts of a program's executable code, data areas, or both that are being used intensively and are therefore important to keep in the fastest possible type of storage. Thus a program's instruction cache working set is the set of program cache lines that need to be kept in the instruction cache if the program is to run at maximum speed.

workload A sequence of requests, such as commands, I/O operations, and subroutine–library calls, that constitute the work being done by a system. The term normally refers to a workload that has been captured in such a way as to be repeatable (via shell scripts, remote terminal emulators), so that it can be used to measure the performance effect of changes to the system.

workload concurrency
The degree to which the system approaches the ideal of always having as many dispatchable threads as there are processors.

workspace (1) In XDS/XOM, a space in which OM objects of certain OM classes can be created, together with an implementation of the object management functions that supports those OM classes.
(2) In CDE, the current screen display, the icons and windows it contains, and the unoccupied screen area where icons can be placed.

workspace background
In CDE, the portion of the display not covered by windows or icons.

workspace icon
In CDE, an icon that has been copied from File Manager to the workspace.

workspace interface
The interface as realized, for the dispatcher's benefit, by each workspace individually.

Workspace Manager
In CDE, the software application that controls the size, placement, and operation of windows within multiple workspaces.

Workspace menu
In CDE, the menu displayed by pointing at an unoccupied area of the workspace and clicking button 3 on the mouse.

workspace object
In CDE, an object that resides in a workspace, rather than inside a viewer in a window. Workspace objects include windows, icons (minimized windows), and objects that have been dragged from File Manager and Application Manager and dropped on a workspace.

workspace switch
In CDE, a control that enables you to select one workspace from among several workspaces.

workspace switch area
In CDE, the rectangular area in the center of the Front Panel that contains the workspace switches, the Lock control, the Exit button, and the busy light.

workstation (1) A configuration of input/output equipment at which an operator works. (2) A terminal or microcomputer, usually one that is connected to a mainframe or to a network, at which a user can perform applications. See also *terminal* on page 2-239.

world coordinates
In GL, the user–defined coordinate system in which an image is described. Modeling commands are used to position primitives in world space. Viewing and projection transformations define the mapping of the world space to screen space. Synonymous with *world space*. See also *modeling coordinates* on page 2-146, *eye coordinates* on page 2-88, *primitive coordinates* on page 2-184, *screen coordinates* on page 2-210, *viewing matrxi* on page 2-257, and *transformation* on page 2-245.

world space Synonym for *world coordinates* on page 2-265.

wrap around (1) The movement of the point of reference in a file from the end of one line to the beginning of the next, or from one end of a file to the other.
(2) In display–based word processing equipment, the automatic disposition of a printable line of text onto two or more display lines, necessitated by the horizontal limits of the display.
(3) The continuation of an operation from the maximum addressable location in storage to the first addressable location.
(4) The continuation of register addresses from the highest register address to the lowest.

wrap test A test that checks attachment or controller circuitry without testing the device itself by returning the output of the device as input.

Wrap To Fit In CDE, in Text Editor, this setting controls whether lines are automatically wrapped to fit the window width as you enter text. If set, lines wrap at the edge of the window. If not set, you must press Return to end the line.

write–ahead logging
A logging mechanism wherein all of the log records associated with a transaction are written to the log before the transaction actually commits. This guarantees that those log records will be present in the log and can therefore be used to restore recoverable data to a correct state should the system fail at the exact time of a transaction commit.

write back cache
In Enhanced X–Windows, GCs cached by the library to allow merging independent change requests into one protocol request. See also *cache* on page 2-28.

write queue The message queue in a module or driver containing messages moving downstream. Associated with output from a user process.

write verification
A mode in which the system automatically performs a read operation after performing a write operation. It then compares the data to make sure they are the same.

writemask A set of 8 or 12 bits (depending on the frame buffer configuration), one bit for each bitplane of the frame buffer. During any drawing operation, only those planes enabled by a 1 (one) in the bit mask can be altered. Planes set to 0 (zero) are marked read only.

X

X Atom An Enhanced X-Windows atom. See also *atom* on page 2-13.

X resource In AIXwindows, synonymous with *resource* on page 2-203.

X resource file In AIXwindows, an ASCII file that includes the definition of all property values that were specified as Public in the Widget Property Editor. Such files are editable by the end user.

X Server See *server* on page 2-215.

X Toolkit See AIXwindows *Toolkit* on page 2-7.

X–Windows A network–transparent windowing system developed by MIT. It is the basis for Enhanced X–Windows. See also *X11 client* on page 2-266.

X.3 The CCITT recommendation that specifies the service provided to an asynchronous (start/stop) device by a packet assembler/disassembler (PAD).

X11 client An application that is compatible with version X11 of Enhanced X–Windows. See also *X–Windows* on page 2-266.

X.21 In data communications, the 21st CCITT recommendation in the X series, defining the connection of data terminal equipment to an X.21 public data network for digital leased and circuit–switched services. In X.25 communications, X.21 is available at the physical level only.

X.21 bis A CCITT recommendation, defining the use on public data networks of DTEs designed for interface to synchronous V series modems.

X.25 The 25th CCITT recommendation in the X series, defining the interface between data terminal equipment and packet switching data networks.

X.25 adapter Synonym for X.25 Interface Co-Processor/2 on page 0.X.25 Interface Co-Processor/2
The separately orderable adapter card that attaches a system unit to an X.25 packet switching data network. Synonymous with *X.25 adapter*.

X.25 line In X.25 communications, the physical link between the DTE and the DCE, and the service subscribed to. See also *communications line* on page 2-44.

X.25 link The X.25 line from the X.25 adapter to the network terminating unit. In the X.25 API, a link is equivalent to an X.25 port. See also *X.25 port* on page 2-266 and *data link* on page 2-61.

X.25 network A service providing packet–switched data transmission that conforms to Recommendation X.25 adopted by the CCITT.

X.25 port A device in the **/dev** directory that corresponds to an X.25 link. See also *X.25 link* on page 2-266.

X.28 The CCITT recommendation that specifies the user interface between an asynchronous (start/stop) device and a packet assembler/disassembler (PAD).

X.29 The CCITT recommendation that specifies the user interface between a DTE and a remote packet assembler/disassembler (PAD).

X.121 The 121st CCITT recommendation in the X series, defining a convention for the network user address (NUA).

XCOFF See *extended common object file format* on page 2-86.

xcomms The command that provides a menu panel for other X.25 commands (the **xmanage**, **xroute**, and **xtalk** commands).

XDR See *External Data Representation* on page 2-87.

XDS The X/Open Directory Service.

XGSL See *Graphics Support Library* on page 2-103.

XID See *exchange identification* on page 2-84.

XID Node ID A field that provides the node ID of the physical unit. This value is the ID that is exchanged with the remote physical unit when a connection is first established during the XID operation.

Xlib In Enhanced X–Windows, a C language subroutine library that client programs use to interface with the windowing system.

Xlib call See *Xlib* on page 2-267.

xmanage The command that enables system managers to manage X.25 ports.

Xmodem protocol
A communications protocol in which messages are sent in blocks of 128 characters, surrounded by control characters. This protocol also uses the **checksum** program for error checking. The xmodem protocol can detect data transmission errors and then retransmit the data.

xmonitor The command that enables system managers to monitor activity on X.25 ports.

xmpeek A program that allows you to ask any host about the status of its **xmservd** daemon.

XNS Xerox Network Systems. The network architecture developed by the Xerox Corporation in the 1970s. The XNS Internet protocol suite is similar to the TCP/IP suite. However, different packet formats and terminology are used.

XOM X/Open Object Management.

xroute The command that enables system managers to manage the X.25 routing list

Xt intrinsics See *intrinsics* on page 2-121.

xtalk The command that enables users to use X.25 to converse with other users, exchange messages and files, and manage X.25 address lists.

XTI X/Open Transport Interface. A library implementation, as specified by X/OPEN CAE Specification of X/Open Transport Interface and fully conformant to X/OPEN and XPG4 Common Application Environment (CAE) specification, that defines a set of transport–level services that are independent of any specific transport provider’s protocol or its framework.

XTISO XTI over SOckets.

XY format The format of a pixmap organized as a set of bitmaps representing individual bit planes that appear in most–significant to least–significant bit order. See also *Z format* on page 2-269.

Y

Y/C signal A yellow/chroma signal. Y(yellow) represents luminance and C represents chroma (color). The luminance contains both the image detail and intensity levels, while the chroma has the color information for the image.

yacc source file
File containing yacc programming language code.

yank To copy a word or line of text into memory.

Z

z–buffer or z–buffering
In 3D computer graphics, applies both to the device and the techniques commonly used as an aid in removing hidden lines and hidden surfaces. If z–buffering is enabled, each pixel stores a depth value as well as a color value. In simple terms, the depth can be thought of as the distance from the viewer's eye to the pixel. Whenever a drawing routine tries to update a pixel, it first checks the current pixel's "depth" or "z–value" and will only update that pixel with new values if the new pixel is closer than the current pixel. The region of memory that stores the z–values is also referred to as the z–buffer. See also *hidden surface* on page 2-107.

Z format The format of a pixmap organized as a set of pixel values in scanline order. See also *XY format* on page 2-267.

zero bit insertion and deletion
Inserting a zero after every four 1–bits and then removing the zeros to return the data to normal.

zero suppression
The removal or substitution of blanks for leading zeros in a number. For example, 00057 becomes 57 when using zero suppression.

zombie process
An ended process whose entry remains in the process table, but to which a user or kernel space is not allocated. A process becomes a zombie process when it issues the **exit** subroutine and the following circumstances occur: Its parent process is not running a **wait** subroutine and has not set its **SIGCLD** signal action to the **SIG_IGN** value indicating that it does not intend to wait for its children to finish.

zone of authority
The set of names managed by a single name server.

zoom factor A multiplier to determine the amount of enlargement of a specified screen rectangle. The *x* zoom factor determines the enlargement in the *x* direction; the *y* zoom factor determines the enlargement in the *y* direction.

zooming In computer graphics, the progressive scaling of a display image to give the visual impression of going from a faraway view of an image to a close–up view, or a close–up view to a faraway view.

Vos remarques sur ce document / Technical publication remark form

Titre / Title : Bull Glossary

Nº Reférence / Reference Nº : 86 A2 29EF 01

Daté / Dated : October 2002

ERREURS DETECTEES / ERRORS IN PUBLICATION

AMELIORATIONS SUGGEREES / SUGGESTIONS FOR IMPROVEMENT TO PUBLICATION

Vos remarques et suggestions seront examinées attentivement.
Si vous désirez une réponse écrite, veuillez indiquer ci-après votre adresse postale complète.

Your comments will be promptly investigated by qualified technical personnel and action will be taken as required.
If you require a written reply, please furnish your complete mailing address below.

NOM / NAME : _____ Date : _____

SOCIETE / COMPANY : _____

ADRESSE / ADDRESS : _____

_____

Remettez cet imprimé à un responsable BULL ou envoyez-le directement à :

Please give this technical publication remark form to your BULL representative or mail to:

BULL CEDOC
357 AVENUE PATTON
B.P.20845
49008 ANGERS CEDEX 01
FRANCE

Technical Publications Ordering Form

Bon de Commande de Documents Techniques

To order additional publications, please fill up a copy of this form and send it via mail to:
Pour commander des documents techniques, remplissez une copie de ce formulaire et envoyez-la à :

BULL CEDOC
ATTN / Mr. L. CHERUBIN
357 AVENUE PATTON
B.P.20845
49008 ANGERS CEDEX 01
FRANCE

Phone / Téléphone :	+33 (0) 2 41 73 63 96
FAX / Télécopie	+33 (0) 2 41 73 60 19
E–Mail / Courrier Electronique :	srv.Cedoc@franp.bull.fr

Or visit our web sites at: / Ou visitez nos sites web à:

http://www.logistics.bull.net/cedoc

`http://www–frec.bull.com` **`http://www.bull.com`**

CEDOC Reference # No Référence CEDOC	**Qty** Qté	**CEDOC Reference #** No Référence CEDOC	**Qty** Qté	**CEDOC Reference #** No Référence CEDOC	**Qty** Qté
__ __ ____ _ [__]		__ __ ____ _ [__]		__ __ ____ _ [__]	
__ __ ____ _ [__]		__ __ ____ _ [__]		__ __ ____ _ [__]	
__ __ ____ _ [__]		__ __ ____ _ [__]		__ __ ____ _ [__]	
__ __ ____ _ [__]		__ __ ____ _ [__]		__ __ ____ _ [__]	
__ __ ____ _ [__]		__ __ ____ _ [__]		__ __ ____ _ [__]	
__ __ ____ _ [__]		__ __ ____ _ [__]		__ __ ____ _ [__]	
__ __ ____ _ [__]		__ __ ____ _ [__]		__ __ ____ _ [__]	
[_ _] : **no revision number means latest revision** / pas de numéro de révision signifie révision la plus récente					

NOM / NAME : ____________________________ Date : __________

SOCIETE / COMPANY : ____________________________

ADRESSE / ADDRESS : ____________________________

____

PHONE / TELEPHONE : ____________________ FAX : ____________________

E–MAIL : ____________________________

For Bull Subsidiaries / Pour les Filiales Bull :

Identification: ____________________________

For Bull Affiliated Customers / Pour les Clients Affiliés Bull :

Customer Code / Code Client : ____________________________

For Bull Internal Customers / Pour les Clients Internes Bull :

Budgetary Section / Section Budgétaire : ____________________________

For Others / Pour les Autres :

Please ask your Bull representative. / Merci de demander à votre contact Bull.

PLACE BAR CODE IN LOWER LEFT CORNER

BULL CEDOC
357 AVENUE PATTON
B.P.20845
49008 ANGERS CEDEX 01
FRANCE

ORDER REFERENCE
86 A2 29EF 01

Bull

Utiliser les marques de découpe pour obtenir les étiquettes.
Use the cut marks to get the labels.

AIX
Glossary

86 A2 29EF 01

AIX
Glossary

86 A2 29EF 01

AIX
Glossary

86 A2 29EF 01