

AIX 5L 版本 5.2



词汇表

AIX 5L 版本 5.2



词汇表

注

使用本资料及其支持的产品之前，请阅读第 143 页的『声明』中的信息。

第二版（2001 年 4 月）

除非在新版本中另有生命，此版本适用于所有 AIX 5L V5.2 以及本产品的所有后续发行版。

© Copyright International Business Machines Corporation 1997, 2002. All rights reserved.

目录

关于本书	v
ISO 9000	v
词汇表	1
附录. 声明	143
商标	144

关于本书

本词汇表提供了用于 AIX 操作系统文档库的专业术语定义。本词汇表未定义非技术性字典中定义的术语以及在信息处理中无特殊含义的术语。

本词汇表包含出自以下出版物的术语和定义：

Information Technology Vocabulary，由国际标准化组织和国际电工技术委员会第一专门技术委员会第一委员会（ISO/IEC JTC1/SC1）编写。

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

本产品的开发和制造中使用了 ISO 9000 质量认证体系。

词汇表

[A]

阿拉伯数字 (Arabic numerals)：用来描述十进制数的 10 个数字：数字 0、1、2、3、4、5、6、7、8 和 9。与字体无关。另见罗马数字 (*Roman numerals*)。

安培 (A 或 amp) (ampere, A or amp)：电流度量单位，等于每秒流过 1 库仑的电量，或者等于 1 伏特电压加在 1 欧姆的电阻上产生的电流。

安全节点 (secure node)：能够运行一个或多个应用程序服务器和一个或多个“监视器”系统组件（通常为高度可信的机器）的节点。

安全性 (security)：对数据、系统操作以及设备的保护，以防意外或故意的毁坏，损坏或曝光。

安全注意密钥 (secure attention key, SAK)：一个按键序列，它结束所有与终端相关的进程来为与 TCB 的安全通信提供可信的路径。SAK 序列是 Ctrl-x 后紧跟 Ctrl-r。

安装脚本 (installation script)：由开发人员创建的用来安装程序的 shell 脚本或可执行文件。脚本文件必须遵守特定的准则，以便与操作系统提供的程序安装工具相互兼容。

安装图标 (Install Icon)：在 CDE 中，一个子面板上的项，它用于使用拖放将图标安装到前面板中。

安装映象 (installation image)：一个安装映象以备份格式包含正在安装的软件的副本以及系统需要安装的该软件产品的其它文件的副本。

安装助手 (Installation Assistant)：一个用来执行系统配置任务的应用程序。

安装 (install)：(1) 添加一个程序、程序选项、或软件程序到系统中，它通过这样一种方式：它是可运行的，而且与该系统中的所有受到影响的程序是正确交互。(2) 将一块硬件连接到处理器。(3) 当讨论连接到处理器的内部部分的过程时，所使用的连接过程。

安装 (installation)：(1) 将程序或者程序选项添加到计算机的大容量存储介质的过程，使该程序可运行，且确保该程序与该系统中的所有其它受到影响程序正确交互。(2) 添加设备驱动程序到该系统并且激活驱动程序使其可用的任务。(3) 连接一块硬件到处理器上或者添加软件程序到系统的任务。

按比例缩放 (scaling)：(1) 在计算机图形中，指通过将图像坐标乘以一个常量来放大或缩小整个显示图像或其一

部分。(2) 在编程中，指明固定点或浮点常量的小数部分占据目标代码中数字位置的数量。(3) 在 GL 中，指沿着轴均匀扩展的图元。

按行优先次序 (row-major order)：一种存储数组元素的方法，如当临近的内存元素被访问时，最右边下标改变最快。

按钮抓取 (button grabbing)：使用鼠标按钮制定活动抓取。另见抓取 (*grab*)、指针抓取 (*pointer grabbing*)、键抓取 (*key grabbing*)、被动抓取 (*passive grab*) 和活动抓取 (*active grab*)。

按钮 (button)：(1) 屏幕上可选择字或照片。一旦选中或激活，按钮以和键盘按键开始操作相同的方式开始操作。(2) 按钮包含键盘按钮、鼠标、光笔或拨号和按钮框上的按钮。

按钮 (push button)：其内带有文本的圆角矩形。按钮用在对话框中，当选择按钮时，立即出现操作。

按下按键 (button press)：通过按下鼠标按键启动功能。

凹和凸多边形：在计算机图形学中，如果在图形中连接任意两点的线段完全包含在图形中，则该多边形为凸多边形。非凸多边形有时称为凹（多边形）。仅描绘凸多边形的算法比同时描绘凸多边形和凹多边形的算法要简单的多。

[B]

八进制常量 (octal constant)：数字 0（零），紧接着任意 0 到 7 的数字。

八进制 (octal)：(1) 基于八位编号的系统。(2) 属于固定基数的记数法基数为八。

八位元字符串 (octet string)：请参阅对象数据管理器 (*Object Data Manager*)。

八位元 (octet)：(1) 8 位组（也就是一个字节）。(2) 属于选择、条件或者有八个潜在值或说明的选项。另见八位元字符串 (*octet string*)。

版本标识 (version identifier)：License Use Management 中，标识产品版本的字符串；通过版本标识，许可证服务器就可以区分一个产品的不同版本。

版本号 (version number)：程序的版本级别，它是硬件和运行程序的基本操作系统的指示符。版本、发行版、修改和修订级一起组成程序级或程序版本。另见修正号 (*fix*)

number)、修订版本号(modification number)、发行版本号(release number)、版本(version)和程序级别(program level)。

版本控制(version control): 说明软件、硬件或文档的每个修订版的内容的过程。

版本(version): 应用程序或许可程序的特定实例。程序的版本、发行版、修改和修订级。版本、发行版、修订和修正级别一起组成程序级或程序版本。与先前版本相比,程序的新版本通常具有重要的新代码和功能。另见修正号(fix number)、程序级别(program level)、修订版本号(modification number)、发行版本号(release number)和版本号(version number)。

半对话(half-session): 在会话的一端提供数据流控制和传输控制的组件。

半高(half high): 指具有一半高度(41.3 mm 或 1.63 in.), 宽度与长度和全高设备相同的存储设备。另见全高(full high)。

半双工(half duplex, HD 或 HDX): 与通信相关, 用这种方式, 每次只能以一个方向发送数据。对照双工(duplex)。

半字(halfword): 组成半高计算机字并可作为一个单元进行寻址的相邻的位或字符序列。请与字(word)对照。

帮助查看器(Help Viewer): CDE 中显示联机帮助的软件应用程序。

帮助管理器(Help Manager): CDE 中的特殊帮助卷, 列出了在系统上注册的所有联机帮助。

帮助回调(help callback): 调用帮助的函数。

帮助卷(help volume): CDE 中关于某个主题的帮助信息的全部正文。

帮助文件(help file): 与程序源代码分开的文件, 其包含有“基本操作系统”消息服务可以使用的特殊帮助格式的帮助用户定义。

帮助(help): 描述如何使用应用程序软件或如何执行系统操作的一个或多个显示图像。

绑定程序(binder): 请参阅链接编辑器(linkage editor)。

绑定会话(BIND)(bind session, BIND): 在 SNA 产品中, 一个激活两个逻辑单元间会话的请求。另见绑定(bind)。

绑定映像(bind image): 在 SNA 中, 系统服务控制点(SSCP)发送至主逻辑单元(PLU), 然后 PLU 在 BIND

请求中发送至次级逻辑单元(SLU)的会话参数; 这些参数为 LU-LU 会话指定了被提议的协议选项。

绑定(BIND): 请参阅绑定会话(bind session)。

绑定(bind): (1) 把变量与绝对地址、标识符或虚拟地址或程序中的符号地址/标号关联。(2) 在 SNA 中, 一种激活两个逻辑单元间会话的请求。另见绑定会话(bind session)。(3) 有关如何设置绑定的信息, 请参阅联系(set)。

绑定(binding): (1) 一个在客户机和服务器(同时为对象, 导出至对象的接口)之间的临时关联。一个绑定仅对设置此绑定的程序有意义, 它以约束句柄表示。(2) 在多处理器上下文中, 强制线程到指定的物理处理器以获取处理器绑定的好处。

磅(lb)(pound(lb)): 用于重量的计量单位, 等于 16 盎司或 454 克。

包含文件(include file): 包含由一组函数、程序或用户使用的声明的文本文件。与头文件(header file)同义。另见include 语句(include statement)。

包交换技术(packet switching): 通过寻址包路由和传输数据以便仅在包传送期间占用通道。在传送完成时, 通道可以用来传送其它包。和信息包方式操作(packet mode operation)同义。另见电路交换(circuit switching)。

包头(packet header): 在 X.25 通信技术中, 控制在包的开始的信息; 包的内容取决于包类型。

包循环(pack loop): 压缩稀疏数组的现用单元格到另一个数组中的连续单元格的循环, 以便作为结果的数组在第一个和最后一个现用单元格间不包含空单元格。

包组装机/拆装机(packet assembler/disassembler, PAD): 在 X.25 通信技术中, 作为到 X.25 网络的异步(起止式)连接设备。

包(packet); 信息包; 分组: 在数据通信中的二进制数字序列, 包含数据和控制信号, 这在传送和交换时作为一个混合的整体。数据、呼叫控制信号、以及出错控制信息以指定的格式排列。另见呼叫接受信息包(call-accepted packet)、呼叫连接信息包(call-connected packet)、呼叫请求信息包(call-request packet)、清除确认信息包(clear-confirmation packet)、清除指示信息包(clear-indication packet)、清除请求信息包(clear-request packet)、数据包(data packet)、进入呼叫信息包(incoming-call packet)、中断包(interrupt packet)、中断确认包(interrupt-confirmation packet)、复位请求包(reset-request packet)、复位确认包(reset-confirmation packet)、地址字段(address field)、以及重新启动确认包(restart-confirmation packet)。

保存安装 (preservation installation)： 一种安装方法，当在系统上安装 BOS 前一版本并要在根卷组中保留用户数据时使用它。但是，该方法会覆盖 */usr*、*/tmp*、*/var* 和 */*（根）文件系统，因此会丢失这些目录中的任何用户数据。系统配置必须在完成保存安装后进行。

保存的用户标识 (saved user ID)： 运行 *setuid* 程序时获得的用户标识。保存的用户标识与运行的文件的所有者相同。如果运行的文件不是 *setuid*，则保存的用户标识设置为父标识的有效用户标识。

保存集 (saveset)： 当连接关闭时不应该破坏而应该重新映射或取消映射的窗口客户机的列表。通常如果管理器异常结束，窗口管理器用它来避免窗口丢失。

保护级别 (protection level)： 安全网络通信受保护的等级。

保护字段 (protected field)： 用户不能进入、修改或删除数据的显示字段。

保护 (protection)： 限制访问或使用全部或部分计算机系统的一种安排。

保留字符 (reserved character)： 一种字符或符号，该符号有特殊（非文字）含义，除非被引证。

保留字 (reserved word)： 一种字，为某种特殊目的在编程语言中定义它，且它不能作为用户声明的标识出现。

暴露事件 (exposure event)： 在“增强的 X-Windows”中，当上下文丢失以及窗口被遮住或重新配置的时候，发送给客户机以通知它们的事件。服务器不保证在窗口被遮住或重新配置的时候保存其上下文。

报文尾 (trailer)： 报文一部分，其中包含控制信息。“VAX 报文尾封装协议”使用报文尾。报文尾封装允许接收主机在页面对齐的边界上接收数据，这是利用页面映射的虚拟内存环境的要求。

背景颜色 (background color)： 在 AIXwindows 和“增强的 X-Windows”中，分配给出现在显示小窗口或小配件中前景元素后面的图形字段的单一电子颜色。对照前景色 (*foreground color*)。

背景 (backdrop)： 左 CDE 中，覆盖工作区背景的模式。

被动网关 (passive gateway)： 不交换路由信息的网关。其路由信息不限定地包含在路由表中，并包含在任何发送了的路由信息中。对照活动网关 (*active gateway*)。

被动抓取 (passive grab)： 在“增强的 X-Windows”中，抓取一个键或一个按钮是被动抓取。当实际按键或按钮，抓取便变为活动抓取。对照活动抓取 (*active grab*)。

另见抓取 (*grab*)、按钮抓取 (*button grabbing*)、指针抓取 (*pointer grabbing*) 和键抓取 (*key grabbing*)。

备份副本 (backup copy)： 一份副本，通常为保存的一个或一组文件，以备原始文件无意地更改或破坏。

备份格式文件 (backup format file)： 处于备份格式的文件。

备份格式 (backup format)： 当命令 *backup* 制作了文件的副本时，它以这种格式写文件。这种格式的文件在其可用之前必须以命令 *restore* 进行恢复。

备份系统 (backup system)： 请参阅转储 (*dump*) 和恢复 (*restore*)。

备份 (back up)： 为保护目的而把信息拷贝到（通常为）软盘或磁带。

备份 (backup)： 可在发生故障或数据丢失时使用的系统、设备、文件或设施。

被叫线路地址修改通知 (called line address modified notification)： X.25 通信中，一可选 CCITT 指定设施。

贝赛尔三次曲线 (Bezier cubic curve)： 在 GL 中，一条三次样条曲线近似于一个经过第一个和第四个控制点的四控制点的集合，在两个样条曲线段相交的地方有连续的斜率。另见参数三次曲线 (*parametric cubic curve*)。

被扫图像 (scanned image)： 有顺序的，一部分一部分检查的图像。

备用单元 (隔离单元) (alternate cell, isolated cell)： 一 NCS 单元，限制对网络中部分节点组的访问。该单元类型通常用于限制网络中特定节点集的 License Use Management 许可证事务。

备用电池 (battery backup)： 一种电源，在主交流电源断电期间，允许处理器抽屉和其它数据存储设备继续运作。

备用字符集 (alternate character set)： 一字符集，包括一些为某些打印机定义的特殊字符（如数学字符和希腊字符）。

本地变量 (local variable)： 定义在一个程序模块或者过程中的一个符号，仅仅用在那个程序模块或者过程中。

本地存储器 (local storage)： 一个直接从用户系统中访问（不需要远程通信）的设备，在这里信息可以保持也可以被检索。

本地单元 (local cell)： 本地计算机所属的单元。另见外部单元 (*foreign cell*)。

本地地址 (local address)： 指定当前网络或者主机的地址。本地地址通常参照为本地主机地址或者本地网络地址来区分这两种类型。

本地调步 (local pacing)： 本地系统生成调步以尝试控制远程系统的输出，即本地系统的输入。参阅调步 (*pacing*)。

本地回送 (local echo)： 在一个通信工作站上，一个位置上每个信号显示两次：一次当输入到本地工作站，在次从通信链路返回。

本地监听 (local listen)： 在本地节点的一个链接站等待来自于一个远程站的进入呼叫。

本地名称 (local name)： 一个有意义的和可用的名称，它仅仅来自于入口存在的单元。本地名称是一个全局名的缩短格式。本地名称开始于前缀 */:*，而不包含一个单元名称。

本地位置代理 (Local Location Broker, LLB)： (1) NCS 位置调度部件。一台保持本地主机上的对象信息的服务器。LLB 也提供了位置代理转发工具。(2) 一个从 License Use Management 服务器提供到全局位置代理的接口的服务。LLB 守护程序 (*llbd*) 没有关于网络级服务的信息。它持续在后台运行以拦截和转发信息到 *glbd*。另见位置代理 (*Location Broker*) 和位置调度客户机代理 (*Location Broker Client Agent*)。

本地语言支持 (national language support)： 用于非美式英语的语言的转换子例程，它转换各种不同的字符集与日期和时间格式。与 *国际字符支持 (international character support)* 同义。

本地终端号 (national terminal number, NTN)： 在 X.25 通信中，在网络用户地址中服从国家代码的 1 到 12 位数字的数。

本地主机 (local host)： 在 TCP/IP 中，一个特定的操作员工作所在的网络主机。与 *当前主机 (current hose)* 同义。

本地作用域 (local scope)： 一个块中声明的一个名称具有本地作用域，而且仅仅可以应用在该块中。

本地 (local)： (1) 适合于一个直接由您的系统访问，而不需要用通信线路的设备，文件，或者系统。对照远程 (*remote*)。(2) 适合于那些仅仅被定义和利用在一部分计算机程序中的信息。

本征函数 (intrinsic function)： 在 FORTRAN 语言中，一种随运行时环境一起提供的函数 (执行数学、字符、逻辑或位操作运算)。

本征 (intrinsics)： 在“增强的 X-Windows”中，为组合小窗口、它们的子类和其它的客户机之间的构造和接口提供的一组管理机制。并且，本征提供了组织小窗口集合到应用程序中的能力。

崩溃 (crash)： 计算机服务的意外中断，通常由严重的硬件或软件故障导致。

笔划文本 (stroke text)： 可编程字符集 (*programmable character set*) 和几何文本 (*geometric text*) 的同义词。

比例因子 (scale factor)： (1) 在实数中指定小数点位置的数。(2) 在按比例缩放中用作乘数的数。

比例因子 (scaling factor)： 多处理器上工作负荷的吞吐量除以可比单处理器 (不在单处理器 SMP 系统上) 上工作负荷的吞吐量。

比赛条件 (race condition)： 对于 *signal* 子例程，当信号操作设置为 *SIG_DFL* 值时信号发生的条件，并且信号捕捉功能还未为此信号将自己建立为捕捉器。

闭合用户组 (closed user group, CUG)： X.25 通信中一个分配给某个设施的用户子组，使得一个子组成员仅与该子组的其它成员通信。DTE 可以属于多个闭合用户组。另见 *双边闭合用户组 (bilateral closed user group)* 和 *可选设施 (optional facility)*。

必需参数 (required parameter)： 不自动提供值的一种参数。用户必须提供一个值。

必需列表 (required list)： 在“增强的 X-Windows”中的一个有序列表，它含有已安装色图的一个子集。

必需值 (required value)： 同必需参数 (*required parameter*)。

必要 (requisite)： 一种软件产品或服务更新，它必须与另一种软件产品或服务更新安装一起被安装。如果您尝试在没有必需的必要软件情况下就安装软件产品或服务更新，则系统消息显示必需的必要软件的名称。

闲置系统 (quiet system)： 其上不运行除其性能被定时的进程外的进程的系统，且所有通信设备已从该系统断开连接。

边界对齐 (boundary alignment)： 在信息单元的完整边界上，一固定长度字段 (如半字或双字) 在主存储器中的位置。例如，一个字边界是可以被四除尽的存储器地址。

边界框 (bounding box)： (1) 计算机图形学中字符边界框是包含在 *x*、*y* 原点的形状的最小矩形。(2) 在 GL 中，一个约束图元的二维矩形。边界框可用来确定图元是否位于剪切区域的内部。另见 *字符单元 (character cell)* 和 *剪切 (clipping)*。

边界 (border) : 把显示对象从屏幕上任何其它事物中分离出来的可视边界。

编程接口 (programmatic interface) : 在 AIXwindows 中的一种应用程序, 它以这样的一种方式创建, 将在有其它应用程序并发运行的多客户机环境中操作。客户机通过 Xlib 调用或 Xlib 上的库构建与窗口管理器进行通信。

编辑缓冲区 (edit buffer) : 由编辑器使用的临时存储区域。

编辑 (edit) : (1) 增加, 改变, 删除, 重新安排或修改数据的形式或格式。(2) 检查信息的准确性并表明是否发现错误。

编译程序 (compiler) : (1) 一个将源程序转换为可执行程序 (对象程序) 的程序。(2) 一个将用高级编程语言编写的指令转换为机器语言的程序。

编译单元 (compilation unit) : 计算机程序的一部分, 足以完成正确编译。Pascal 有两种单元类型: 程序单元和段单元。Ada 语言中, 一个编译单元是程序单元的声明或主体, 作为独立文本编译。它可通过上下文子句可选地进行, 通过多于一个子句为其依赖的其它编译单元命名。另见编译单元 (*compilable unit*) 和外部变量 (*external variable*) 。

编译器指令语句 (compiler directing statement) : 与编译器指令 (*compiler directive*) 同义。

编译器指令 (compiler directive) : 控制编译器做什么而不是用户程序做什么的语句。

编译时间 (compilation time) : 源程序从高级语言转换为机器语言所需的时间。

编译指示 (pragma) : 在 Ada 语言中, 传达信息给编译器。

编译 (compilation) : Ada 语言中, 编译通常指把 Ada 源程序转换为可执行对象模块。使用 Ada 语言调试器时, 编译由单一文件中的一个或多个编译单元构成。如果在一个文件中包含三个软件包规范和两个软件包体, 那么该文件表示一个包含五个编译单元的编译: 三个库单元和两个次单元。在一个编译中通常只有一个编译单元。

编译 (compile) : 将用高级编程语言编写的程序转换为中间语言、汇编语言或机器语言。另见解释例程 (*interpreted routine*) 。

变成幻影 (ghosted) : 不可用选项的描述。菜单项变成幻影或“变灰”表示某个标准选项在当前环境中不可用。

变换 (transformation) : (1) GL 中的一个 4*4 矩阵, 此矩阵帮助确定发生三维绘图的位置、观察点 (查看器的“眼睛”) 的位置以及包含在内的且可视的场景的数量。

变换发生在图形流水线内的四个点上: (1) 模拟变换, 它将模拟坐标映射为世界坐标。所有绘制图元指定假定为模拟坐标的位置。模型化变换可用于正在绘制的图像。(2) 取景变换, 它从世界坐标映射至查看器坐标。查看器坐标系统的原点可看作查看器的“眼睛”的位置, 且取景变换可用于在世界坐标中四处移动此“眼睛”。(3) 投影变换, 它定义剪切区域内的边界。投影变换将查看器坐标映射至规格化设备坐标, 且剪切平面位于 $x = +/-w, y = +/-w, z = +/-w$ 。投影变换可用于定义屏幕上可视的区域。(4) 视口或“NDC 至 DC”变换。视口变换不是满秩的、4*4 的变换矩阵; 矩阵中的对角线元素中只有三个可以更改。视口确定从规格化设备坐标映射至屏幕 (设备) 坐标。缺省情况下, 视口与窗口相同大小, 虽然这可以调整。另见规格化设备坐标 (*normalized device coordinates*)、模拟坐标 (*modeling coordinates*)、旋转 (*rotation*)、视觉坐标 (*eye coordinates*)、屏幕坐标 (*screen coordinates*)、选择区域 (*selecting region*)、查看矩阵 (*viewing matrix*)、世界坐标 (*world coordinates*)、剪切 (*clipping*) 和当前变换矩阵 (*current transformation matrix*) 。

变灰 (grayed) : 在菜单或列表框中列出, 但不能进行选择的某个命令或选项的特征。

便捷创建子例程 (convenience creation subroutine) : 创建某种有用的小窗口组合 (通称便捷小窗口或便捷对话框), 包括相应的 Shell 小窗口 (如有必要)。另见便捷界面 (*convenience interface*) 。

便捷对话框 (convenience dialog) : 一个由 AIXwindows 对话框便捷子例程创建的小窗口或小窗口集。

便捷函数 (convenience function) : 一个自动地执行频繁需要的任务系列以创建或管理小窗口或其它代码结构的函数。便捷函数安装在“AIXwindows 工具箱”中。

便捷界面 (convenience interface) : 一种由 AIXwindows 便捷创建子例程创建的界面。另见便捷创建子例程 (*convenience creation subroutine*) 。

遍历 (traversal) : 请参阅键盘遍历 (*keyboard traversal*) 。

变量长度字段 (variable length field) : 具有变化长度的字段, 它包含以内部、不透明字段开始的数据, 此字段提供字段长度。

变量传递 (pass-by-VAR) : 在 Pascal 中, 通过传递变量的地址到调用例程的参数传递机制。如果调用例程修改形式参数, 也更改相应的实参。与读/写引用传递 (*pass-by-read/write-reference*) 同义。

变量替换 (variable substitution) : 更改并显示字符串中变量值、用变量表示的值替代它的名称的能力。

变量 (variable) : (1) 用于表示某数据项的名称, 当程序运行时, 此数据项的值可以更改。对照常量 (*constant*)。 (2) 编程语言中, 可以在不同时候赋不同值的语言对象。 (3) 可以假定给定值集中任何值的数量。 (4) 对于 Ada 编程, 请参阅对象 (*object*)。

变体部分 (variant part) : Pascal 中记录的一部分, 它可以使一个记录实例不同于另一个。变体部分由共享同一个物理存储器的备用字段序列构成。Ada 语言中, 记录的变体部分根据记录的判别式指定备用记录组成部分。此判别式的每个值建立了变体部件的一个特殊备用。另见标记字段 (*tag field*)。

变址指令 (indexed instruction) : 使用变址地址的指令。

变址字段 (indexed field) : 包含树形数据路径的结构化数据文件中的区域。

标度 (scale) : *nroff* 和 *troff* 命令使用的量度的增量。排字机所有受支持的标度都转换为名为“机器单元 (*u*)”的标度。

标号语句 (labeled statement) : (1) 一个编程语言语句, 它包含一个或者多个标识符后跟一个冒号和一个语句。 (2) 之前有一个标签的一个空语句。

标号 (label) ; 标签 : (1) 磁盘或者软盘中标识文件的目录名称。 (2) 在一个指令字段, 指定给指令开始位置的符号名称。 (3) 在编程语言中, 命名一条语句并包含一个标识符的构造。另见文件名 (*file name*)。 (4) 一个标识符后跟一个冒号, 用来标识程序中的一个语句。一般是一个 *goto* 或者 *switch* 语句的目标。另见语句标号 (*statement label*)。

标记 : (1) 程序含义的最小的独立单元, 由解析器或词法分析器定义。一个标记可以包括数据、语言关键字、标识符或语言语法的其它部分。 (2) *m4* 命令中 *m4* 命令可以识别的任何字母字符串和数字字符串。 (3) 一种宏的类型, 带有指定字符串值的排版预处理器取代此宏。另见字符串寄存器 (*string register*)。 (4) 本地局域网中, 在数据站之间传递的权限符号, 用来指明站点正处于临时控制传输介质的状态中。

标记编号 (token numbers) : 表示标记名称的非负整数。

标记块 (mark block) : 在 Pascal 中, 表明堆中一个子堆的一个动态块头。

标记输入格式文件 (Tagged Input Format File, TIFF) : 使用位图的图形文件格式。

标记字段 (tag field) : Pascal 中定义变体部分结构的记录的字段。另见变体部分 (*variant part*)。

标记 (marker) : (1) 非交互式窗格中的一个可视化符号, 它显示窗格在上一次交互后的光标位置。 (2) 在计算机图形中, 用于标识一个特殊位置的带有特定外观的符号。

标记 (tag) : (1) GL 中, 显示列表上的一个标记, 用作编辑显示列表的位置。 (2) Interleaf 中, 自动编号引用的字母数字名称。此名称可由 Interleaf 或编写程序指定; 但是它必须匹配它正引用的组成部分的自动编号。

标量类型 (scalar type) : 在运行时定义包含单个值的变量的一种类型。对照结构类型 (*structured type*)。在 Ada 编程中, 一个对象或标量类型的值没有组成部分。标量类型是离散类型或真实类型。标量类型的值是按序排列的。

标量设备 (valuator) : (1) 提供标度值的输入设备, 例如微型旋轮或电位计。 (2) GL 中在范围内返回值的输入/输出设备。例如, 一个鼠标逻辑上是两个标量设备: *x* 位置或 *y* 位置。另见标度盘 (*dial*)。

标量 (scalar) : 一个算术对象或枚举对象。

标识符 (identifier, ID) : (1) 引用一个数据对象的名称。标识符包含一些字母、数字和下划线的组合, 但其首字符不能是数字。 (2) 在编程语言中, 对某个语言对象进行命名的词法单位, 例如数组、记录、标签或过程的名称。标识符通常以字母开始, 其后可选地跟字母、数字或其它字符。 (3) 将一个程序、设备或系统标识成另一个程序、设备或系统的位序列或字符序列。 (4) 在“增强的 X-Windows”中, 指与某种资源关联的唯一值, 客户机程序用它来对该资源命名。可在任何连接上使用标识符来命名资源。

标题栏 (title bar) : 出现在窗口顶部且包含一个文件名或标题的栏。它可用于移动或激活窗口。

标题 (caption) : 与表或图像相关联或用以描述的文本。

标志 (flag) : (1) 出现在命令行的修饰符, 命令行带有定义该命令操作的命令名称。标志前通常有一短划线。 (2) 显示开关的设置的指示符或参数。 (3) 对某些条件的出现发信号的字符, 如字的结束。 (4) 描述处理单元的情况的内部指示符。在 PC 机 AT 处理单元中, 表示如果操作的结果为零、如果启用中断与其它情况的标志。与条件码 (*condition code*) 同义。

标准错误 (standard error, STDERR) : 许多程序放置错误消息的地方。

标准屏幕 (standard screen) : 扩展 *curses* 库中的一个屏幕的内存映象, 各例程将对它进行更改。

标准输出 (standard output, STDOUT) : 来自命令的数据的主要目的地。标准输出输出至显示器, 除非重定向或使用管道, 在这种情况下, 标准输出输出至文件或另一个命令。

标准输入 (standard input, STDIN) : 进入命令的数据的主源。除非重定向或使用管道 (在这种情况下, 标准输入来自文件或来自另一个命令的输出), 标准输入来自键盘。

标准 I/O 板 (Standard I/O Board) : “标准 I/O 板”提供一组 I/O 功能, 这对于大多数系统设备而言是基本功能。常用标准 I/O 功能部件是键盘、图形输入板、话筒、鼠标、串口、并口和软盘适配器。

表达式语句 (expression statement) : C 语言中以; (分号) 结尾的表达式。可以使用表达式语句把表达式的值赋给一个变量或调用函数。

表达式 (expression) : (1) 值的表示。例如, 单独或与运算符结合出现的变量和常量。(2) 编程语言中的一种语言构造, 用于从一个或多个操作数 (例如, 文字、标识符、数组引用和函数调用) 计算值。在 Ada 语言中, 表达式定义了一个值的计算。(3) 符号的配置。

表面特征 (surface characteristics) : 书面文档的样式特征: 可读性、句子长度和结构、字长度和用法、动词类型和句子起始部分。

表示地址 (presentation address) : 用于标识一组表示服务存取点的明确的名称。不严格地说, 它可以是 OSI 服务的网络地址。另见 *地址 (address)*。

表示服务存取点 (Presentation Service Access Point, PSAP) : OSI 通信伙伴的地址。它寻址计算机中的应用程序。

表示空间 (presentation space) : 包含与窗口有关的数据和属性的一个数组。

表示子句 (representation clause) : 在 Ada 语言中的一种子句, 它将类型、对象或任务映射选择中的编译器指引到执行程序的基本机器的功能部件上。在一些情况下, 表示子句完全指定映射; 在其它情况下, 它们提供选择映射的条件。

表 (table) : 一个数据数组, 其中的每个项都能通过一个或多个值明确定位。

别名处理 (aliasing) : 一种试图确定存在什么别名的编译过程, 以便优化不会导致错误的程序结果。

别名 (alias) : (1) 节点或文件的替代名称, 可用来代替节点或文件的真实名称。(2) 计算机程序中数据元素或点的替代标号。(3) 已分区数据集成员的替代名称。(4) 用于网络的非正式名称。与 *别名 (nickname)* 同义。(4) 两数据项之间或一数据项和一指针之间的假定或实际关联。

别名 (nickname) : *别名 (alias)* 的同义词。

并行部件 (corequisite) : 必须与另一个指定产品或更新同时安装的产品或更新。

并行处理 (parallel processing) : 在相同的活动内同时执行多个任务的情况。对照 *串行处理 (serial processing)*。

并行传输 (parallel transmission) : (1) 同时传送字符的所有位。(2) 在数据通信中, 同时传送构成同个数据报或数据信号的许多信号元素。

并行设备 (parallel device) : 可以执行两个或更多并发活动的设备。对照 *串行设备 (serial device)*。

并行使用许可证 (concurrent-use license) : 在 License Use Management 中的一种许可证类型, 由许可证服务器管理, 可以供连接到许可证服务器节点的任何节点上的用户使用。并行使用许可证允许按照许可证中规定的用户数并行使用软件产品。

并行通道 (parallel channel) : 在控制器和大型机处理器间的通信协议。

拨号代码 (dial-code) : 在 BNU 中表示电话号码或部分电话号码的代码。

拨号和开关盒 (dial and switch box) : 有 8 个拨号 (标量设备) 和 32 个开关的 I/O 设备。开关盒又称 “按钮盒” (button box) 或 “轻型可编程功能键” (lighted programmable function keys, LPMK)。

拨号目录 (dialing directory) : 在 ATE 中可以与 “异步终端仿真” (Asynchronous Terminal Emulation, ATE) 一起调用的电话号码列表。它类似于电话目录中的页。

拨号盘 (dial) : (1) 一种计算机输入设备, 使用户能够设置参数值。拨号盘是一类标量设备类型。另见 *标量设备 (valuator)*。(2) 用来通过拇指滚轮输入变量的 I/O 设备。

波特率 (baud rate) : 在远程通信中, 传输速率与信号事件同义。波特率通常以位/秒表示。

波特 (baud) : (1) 在一个通信信道中, 每秒内信号级别、频率或相位的更改次数。如果每个波特代表 1 位数据, 那么波特与位/秒相同。非正式地 (如程序员所用) 与 “位/秒” 同义。以法国发明家 J. Baudot (1845-1903) 的名称命名。但是, 一次信号更改 (1 波特) 等于多于 1 位的数据是可能的。(2) 一种信号速率单位, 等于每秒内离散状态或信号事件的数。例如, 1 波特在莫尔斯电码中等于每秒半个点周期、在二进制信号序列中等于每秒 1 位及在八种不同状态中任选一种状态的信号序列中等于每秒一个 3 位值。(3) 异步传输中调制速率的单位, 相当于每秒一个单位间隔; 例如, 如果单位间隔的宽度是 20 毫秒, 则调制速率是 50 波特。另见 *位/秒 (bps)*。

不活跃列表 (slow list)：多站网络上的辅站列表，因为辅站的非活动状态，比起主站，它们受到较不频繁的轮询。

不可访问 (inaccessible)：在 XOM 中，客户机未拥有有效指示符或句柄的上述某个对象。

不可预知的 (unpredictable)：对体系结构规则的一项违例，即实现不需要报告。结果可能包含一份来自线程调用、操作系统、或硬件的错误报告；程序的中止或死锁；或对程序进程不正确操作而没有错误指示。另见非法 (*illegal*)。

不能查看的 (unviewable)：指带有未映射的祖先的映射的窗口。

不受控小窗口 (unmanaged widget)：不能更改其大小的小窗口。

不透明结构 (opaque structure)：正在处理它的代码隐藏数据项或者数据类型的结构。

不透明数据结构 (opaque data structure)：数据结构通过一个功能的代码单元在内部使用，而不是导出在另一个功能代码单元中外部限制使用。外部使用必须避免访问结构的组成部分或者做一个关于它的构造的假设。

不透明数据类型 (opaque data type)：XDR 中，固定大小的字节不被解释为它们在计算机间传递通过的数据流。

不透明 (opaque)：请参阅不透明数据类型 (*opaque data type*)。

不完全类声明 (incomplete class declaration)：未定义任何类成员的 C++ 类。在完全声明或定义某个类之前，您只能使用类名，其中类的大小不是必需的。不完全类声明通常用作前向声明。

补丁程序 (patch)：(1) 在 SNA 中，两个网络可寻址单元 (NAU) 之间交换的信息遍历的路径控制网络组件的系列。路径由一系列路径控制元素、数据链路控制元素和链接组成。(2) 在网络中，两个节点之间的任何路由。(3) 在数据库中，从根段到单个段出现的段的序列。(4) 在 CDE 中，指定文件夹 (目录) 分层位置的文本字符串。

捕获键 (capture key)：一个切换控制键，启动或停止在活动连接期间保存显示在屏幕上的数据的过程。

捕获文件 (capture file)：一个通信程序用来俘获或记录来自连接 (与远程系统或设备) 的文件。

捕获 (capture)：将图像数字化到“M 视频”捕获适配器 (Capture Adapter) 的显存。

布尔 (Boolean)：以数学家 George Boole 命名的二进制编号系统，其中 0 或 1 是仅能返回两个值。传统上 0 值代表假而 1 值代表真。真值或假值，其中真 = 1，假 = 0。

布局 (layout)：在页面上安排打印事务，包含页边距、行间距、输入规范、页眉和页脚信息，缩进以及更多的信息。与几何 (*geometry*) 同义。

[C]

擦除字符 (erase character)：表示命令行上的前一个字符已擦除的字符。

擦除 (erase)：从数据介质中除去文本，使介质可用于记录新的文本。另见删除 (*delete*)。

裁纸页面 (burst pages)：连续格式纸中，输出页面可以在穿孔处分开。

彩色单元 (color cell)：“增强的 X-Windows”中，由基于红、绿和蓝三个色彩强度值构成的颜色映像图中的项。该值为 16 位无符号数字。零代表最小强度。服务器缩放该值以匹配正在使用的特定显示器。

彩色图形适配器 (color graphics adapter)：一个允许计算机使用彩色显示器的适配器。

彩色显示器 (color display)：一种能够显示两种以上颜色及两种色彩混合产生的变化色彩的显示设备 (与单色显示器相反)。

菜单窗格 (menu pane)：包含一个供用户选择的具有一组选项列表的弹出菜单的物理窗口。另见窗格 (*pane*)。

菜单光标 (menu cursor)：在 AIXwindows，为特定的菜单所定义的光标。每一种类型的光标由一个标识号来标识。

菜单条 (menu bar)：在一个窗口的客户机区域的上部的一个矩形区域，它包含该应用程序的标准下拉菜单。

菜单系统 (menu system)：一个通过用户加速浏览和选择的方式列出了相关的软件选项的交互式接口。

菜单 (menu)：一个操作员可以作出选择的一个显示的列表项。

参考位 (reference bit)：每个页帧表项中的一种位，它指示自上次操作系统清除该页以来对应的页已被访问 (读取或写入)。

参考小窗口 (reference widget)：搜索满足搜索条件的子的父小窗口。

参考 (refer) : 一种书目预处理器, 用于 **nroff** 和 **troff** 命令。“参考”旨在用于文字引用, 并且它支持数据输入、索引、排序、检索、打印、引用和注脚或尾注编号。

参数的双三次曲面 (parametric bicubic surface) : 由三个方程定义的曲面。 x 方程是: $x(u, v) = a_{11}u^3v^3 + a_{12}u^3v^2 + a_{13}u^3v + a_{14}u^3 + a_{21}u^2v^3 + a_{22}u^2v^2 + a_{23}u^2v + a_{24}u^2 + a_{31}uv^3 + a_{32}uv^2 + a_{33}uv + a_{34}u + a_{41}v^3 + a_{42}v^2 + a_{43}v + a_{44}$ 。方程 y 和 z 也类似。在双三次曲面上的点由从 0 到 1 的参数 u 和 v 来定义。如果一个参数为常量而另一个在 0 到 1 之间变化, 结果为三次曲线。对所有的 u, v , 如果 $w(u, v) = 1$, 双三次曲面被称为“普通的”, 但是如果 $w(u, v)$ 为 u, v 的函数, 那么曲面称为“有理的”。另见齐次坐标 (*homogeneous coordinates*)。

参数的双三次曲线 (parametric cubic curve) : 由方程: $x(t) = a_{x1}t^3 + b_{x1}t^2 + c_{x1}t + d_{x1}$; $y(t) = a_{y1}t^3 + b_{y1}t^2 + c_{y1}t + d_{y1}$; $z(t) = a_{z1}t^3 + b_{z1}t^2 + c_{z1}t + d_{z1}$; $w(t) = a_{w1}t^3 + b_{w1}t^2 + c_{w1}t + d_{w1}$ 定义的曲线。这里 x, y, z 和 w 是三次多项式。参数 t 通常在 0 到 1 之间变化。只有当 $a(w), b(w)$ 、或 $c(w)$ 不等于 0 时认为这样的曲线是有理的; 否则, 为仅仅是普通的参数曲线。另见 B 样条三次曲线 (*B-spline cubic curve*)、贝塞尔三次曲线 (*Bezier cubic curve*)、和基数样条三次曲线 (*cardinal spline cubic curve*)。

参数块 (parameter block) : 包含输入输出控制操作的指定参数的内存块。

参数列表 (argument list) : 一参数字符串。

参数声明 (parameter declaration) : 描述函数接收的值。参数声明确定了存储类和值的数据类型。

参数指针 (arg pointer) : 一个指向子例程参数自变量的指针。

参数字符串 (argument string) : 一个在程序或例程之间传递的有序参数列表。

参数组件 (parametric component) : 在 AIXwindows 中, 交付对大多数应用程序必要的所有功能的一种简单机制, 这也使构建更简单且耗费时间少。

参数 (argument) : (1) 扩展或更改命令工作方式的数字、字母或字。(2) 一个在调用例程和被调用例程之间传递的参数。(3) 独立变量。

参数 (parameter) : (1) 用户提供给面板、命令、或函数的信息。(2) 对指定的应用程序给出常量值的变量。(3) 数据在程序和过程间传递。(4) 在 Ada 语言中, 一个参数为跟子程序、项、或类属单元相关的某个指定项, 并用来与相应的子程序主体、接受语句或类属主体通讯。形式参数是用来代表在主体内的指定项的标识。实参是跟相应的子

程序调用、项调用、或类属实例化的形式参数相关的特定项。形式参数的方式指定相关的实参是对形式参数提供值、还是形式参数为实参提供值、还是两者都是。实参和形式参数的关联可用指定的关联、位置关联、或两者的联合来指定。另见形式参数 (*formal parameter*) 和方式 (*mode*)。

参与方 (participant) : 当应用程序初始化事务或接收事务的请求时即为参与方。

操作标签 (action label) : CDE 中显示在操作图标下的名称。另见操作名称 (*action name*)。

操作表 (action table) : 在 Xtoolkit 中, 表指定外部可用过程字符串到由小窗口类实现的相应过程的映射。所有小窗口类记录都包含操作表。

操作码 (op code) : 请参阅操作码 (*operation code*)。

操作码 (operation code, op code) : 告诉处理器执行哪个操作的数字代码。

操作名称 (action name) : 在 CDE 中, 与操作相关联的名称, 缺省显示在操作图标的下方。另见操作标签 (*action label*)。

操作数 (operand) : (1) 指令字段代表数据 (或者数据的位置) 来被操作或处理。不是所有的指令需要操作数字段。(2) 操作员把标识符、常量或者表达式分组。(3) 在实体上执行操作。(4) 用命令名称输入信息, 它定义了命令处理器操作和控制命令处理器运行的数据。

操作图标 (action icon) : 在 CDE 中, 图标代表文件管理器或应用程序管理器窗口或工作区里的操作。操作图标通过创建具有与其代表的操作同名的空可执行文件来创建。

操作系统 (operating system, OS) : 控制系统如何工作的程序集。控制程序的运行, 提供象资源分配、调度、输入和输出控制以及数据管理的服

操作语句 (action statement) : C 语言程序段定义生成的词法分析器如何与其认识的正则表达式响应。

操作 (action) : (1) 应用程序执行的一个已定义任务。修改对象属性或以某种方式操纵对象的操作。一种处理步骤或操作。(2) 在命令 **awk**、**lex** 和 **yacc** 中, C 语言程序段定义当程序认可输入时该如何操作。(3) 在 CDE 中, 为运行应用程序、执行命令和其它活动 (如打印、除去文件和更改目录) 提供方法的桌面构造。操作在配置文件数据库

操作 (operation) : (1) 计算机执行被请求的特定操作 (比如加法, 乘法或移位)。(2) NCS 中, 通过过程访问或操作对象。在语义上用它的名称也它的参数但不用它的实

现来定义操作。(3) 在 Ada 语言中, 与一个或多个类型有关的操作是基本操作。它或者是通过类型声明隐式的声明, 或者是个有参数或输入结果的子程序。

测试方式 (test mode): 用户测试新用户界面的方式, 其中编译的应用程序用于驱动此界面。允许构建、修改、测试和细化操作而无需编译、链接或调试。

层次结构 (hierarchy): 以某种有用的方式分级的一系列元素。在 AIXwindows 中, 对超过 40 种图形对象从最简单到最复杂进行顺序分级, 以确定外观资源和行为资源的继承的相对次序。

插槽 (slot): 系统设备内长形电插座, 其中安装了电子电路板 (卡)。

插入点 (insertion point): 在公共桌面环境中, 当您输入时, 文本插入的所在点。它通常显示为一个闪烁的垂直线或者下划线。

插入方式 (insert mode): (1) 一个在光标位置处将新文本放在已存在文本中的键盘操作。(2) 源项实用程序操作, 在它中间, 源语句被输入并在源成员中添加为新的记录。(3) 在令牌环网络中, 将一个附加设备作为一个环的活动部件。

插入光标位置 (insertion cursor position): 在该点处将会插入文本。

插入光标 (insert cursor): 标记当输入文本时添加新字符所在位置的光标的位置。

插头 (plug): 将电路导线连接到电源的设备。插头设计成能插入插座。

插针 (pin): (1) 为某些函数保留的内存区域。(2) 适配器插头中的一个连接器。

查看矩阵 (viewing matrix): GL 中用于描述相对于世界的查看器位置的矩阵 (虚拟眼睛看作场景)。另见视域 (field of view)、变换 (transformation) 和世界坐标 (world coordinates)。

查看器坐标 (viewer coordinates): 视觉坐标 (eye coordinates) 的同义词。

查看坐标 (viewing coordinates): 视觉坐标 (eye coordinates) 的同义词。

查询 (inquiry): (1) 一个对存储器中信息的请求。(2) 一个将显示站转换为查询模式的请求。(3) 在数据通信中, 来自于另一个系统的信息请求。

查询 (query): (1) 搜索数据, 查找期望信息的操作。(2) 在数据通信中, 主站要求从属站标识自己并给出其状态

的过程。(3) 在交互式系统中, 在工作站上从系统上得出响应的操作。(4) 在文件中请求基于特定条件的信息。

产品标识 (product ID): 一个整数, 它标识供应商的特许软件产品; 通过产品标识, 许可证服务器在同一供应商的产品中区别。

产品密码 (product password): 一种编码的字符串, 它有关于软件产品的许可证的信息。产品密码有两种类型: 许可证密码和复合密码。

产品 (product): 一种软件产品, 由可独立安装的软件包组成。

长常数 (long constant): 一个 4 字节整常数, 其后带有字母 “I” 或 “L”。

长度说明 (length specification): 一个变量所占字节数的源语言说明。

长队列状态 (long queue status): 长状态 (long status) 的同义词。

常量表达式 (constant expression): 一种表达式, 其值在编译过程中确定, 在程序运行过程中不改变。

常量传递 (pass-by-CONST): 在 Pascal 中, 通过传递变量的地址到调用例程的参数传递机制。调用例程不允许修改形式参数。与只读引用传递 (pass-by-read-only-reference) 同义。

常量叠加 (constant folding): 执行编译时操作数都是常量的操作, 并把结果作为常量。另见叠加 (fold)。

常量 (constant): 一种在程序运行过程中不改变值的数据项。对照变量 (variable)。对于 Ada 编程, 请另见对象 (object)。

长状态 (long status): 详细、多行的状态包含比普通短状态更多的有关每个作业的信息。与长队列状态 (long queue status) 同义。

抄本 (transcript): 远程通信中, 包含您在远程系统上输入的命令的写记录和远程系统对这些命令的响应的文件。

超级标量 (superscalar): 在一个给定时钟周期内执行多重指令的能力。

超级对象 (superobject): 是给定对象上级关系的对象。

超级客户机 (superclient): 具有对其他客户机使用的远程资源的读写许可以及根访问权限的无盘客户机。一台超级客户机安装并维护在非 V3.2 无盘服务器上的 V3.2 操作系统的可选软件。

超级块 (superblock)： 在文件系统布局中，指“块 1” (Block 1)，它用于跟踪文件系统，且是文件系统中至关重要的部分。它包含文件系统块的每次分配与解除分配的信息。另见 *i-list*。

超级链接 (hyperlink)： 在 CDE 中的“帮助”文本中，您单击用以显示另一帮助主题的文本或图形。

超级用户权限 (superuser authority)： 另见 *root 用户权限 (root user authority)*。

超级用户 (superuser)： 另见 *root 用户 (root user)*。

超界扫描线 (overscan)： 显示器的特征，表示许多行和水平像素被送到超出显示器的显示空间。

超类 (superclass)： AIXwindows 和“增强的 X-Windows”中将可继承资源传递给层次结构中的更低级子类的小窗口类。另见 *小窗口记录 (widget record)*。

超文本链接 (hypertext link)： 一条联机信息和另一条联机信息之间的连接。另见 *超文本 (hypertext)*。

超文本 (hypertext)： 通过一条信息与另一条信息之间的连接来提供联机信息的方式。这些连接称为超文本链接。数千个这样的超文本链接使您能在整个联机文档中搜索附加或相关信息。另见 *超文本链接 (hypertext link)*。

超整数 (hyper-integer)： 一种 XDR 标准，定义 64 位 (8 字节) 数作为整数和无符号整数的扩展。

程序初始化参数 (Program Initialization Parameters, PIP)： 当程序开始运行时传递给程序的数据。此数据修改程序进行的操作或改变程序运行的环境。

程序单元 (program unit)： 一种主程序或子程序。在 Ada 编程中，程序单元是一般单元、软件包、子程序或任务单元的其中一个。与 *模块 (module)* 同义。另见 *段单元 (segment unit)*。

程序断言 (program assertion)： 用于尝试验证程序校正的数学语句。在图形操作系统中，**assert** 子例程测试程序断言。

程序堆栈 (program stack)： *调用堆栈 (invocation stack)* 的同义词。

程序分级交互式图形系统 (Programmers' Hierarchical Interactive Graphics System, PHIGS)： 一种 ANSI 和 ISO 标准。PHIGS 通过使用保留的数据结构定义一个应用程序编程接口，该接口的设计用于交互式二维和三维图形应用程序。

程序故障管理 (program fault management, PFM)： NCS 的一个子系统，当应用程序不能成功完成时，它允许用户安装清除例程。

程序级 (program level)： 程序的版本、发行版、修订版和修正级别。另见 *修正号 (fix number)*、*修订号 (modification number)*、*发行版号 (release number)*、*版本 (version)*、*后台 (background)* 和 *版本号 (version number)*。

程序计数器 (program counter)： 处理单元中指导计算机通过程序的寄存器。*指令地址寄存器 (instruction address register)* 的同义词。

程序临时性修订 (program temporary fix, PTF)： 对许可程序的当前发行版中的缺陷的临时解决方案或旁路。

程序文本段 (program-text segment)： 一种虚拟内存段，它包含应用程序的可执行指令。程序文本段由在该段中发生的指令高速缓存未命中标识。

程序文本 (program text)： 能运行的程序的一部分。见 *文本 (text)*。

程序 (program)： (1) 一种文件，它包含一组符合特定编程语言语法的指令。(2) 适合于计算机处理的指令序列。处理可包含使用汇编程序、编译程序、解释程序或转换程序的运行作准备，以及运行程序。(3) 在编程语言中，一个或多个相关模块的逻辑组合。在 Ada 语言中，一个程序是由许多编译单元组成的，其中一个编译单元是调用主程序的一个子程序。程序执行由主程序执行组成，它可能调用在程序中其它编译单元中声明的子程序。(4) 设计、编写和测试计算机程序。

成员函数 (member function)： 被声明为一个类的成员的 C++ 操作符和函数。一个成员函数可以访问私有的、受保护的数据和它的类的一个对象的成员函数。成员函数也叫做方法。

成员 (member)： (1) 一个结构，一个联合，或者一个库的一个数据对象。(2) 元素 (*element*) 的同义词。(3) 一个结构，集合或者类中的一个 C++ 数据对象或者函数。成员可以是类，枚举，位字段和类型名。

成组异步平衡方式 (set asynchronous balance mode, SABM)： 一个链路控制帧。

持久段 (persistent segment)： 其页在磁盘上有永久位置，而不是在调页空间中有临时槽的段。

持久数据 (persistent data)： 跨多个事务应用程序运行，无论系统发生故障或重新启动都保留其值的数据。

持久性 (persistence)： 在 Display PostScript (DPS) 中，用于复合字符串中的所有后继文本段，直到遇到新字符集为止的指定字符集。

冲突查找请求 (colliding find request)： 当两台链接站试图同时相互呼叫对方时发生的情况。

冲突 (collision)：(1) 一种不希望出现的情况，因为在介质上并发传输而导致错乱数据。(2) X.25 通信中，当 DTE 和 DCE 在相同的逻辑通道中同时传送信息包（例如，清除请求信息包和清除指示信息包）时发生的情况。可能是清除冲突、呼叫冲突或复位冲突。另见清除冲突 (*clear collision*)。

重定父级 (reparent)：当在界面窗口间移动小窗口时，它被重新指定给新位置的父小窗口，并且自动获取新父的可继承属性。

重定向控制 (redirecting control)：传送某个操作到一个指定客户机。当窗口管理器或客户机程序实施窗口布局策略，以阻止更改窗口的大小或位置的尝试时使用。

重定向 (redirect)：从某个进程转移数据到一个文件，或从某个设备转移数据到它通常不进入的进程。

重分页故障 (repage fault)：虚拟内存页上的缺页故障，对已从磁盘“最近”读是已知的。

重复键 (typematic key)：当被按下时，多次重复其功能的键。

重复平铺操作 (repetitive tiling operation)：在 XGSL 中的一种操作，它由重复复制 16x16 像素阶梯矩形组成（由阶梯像素图数据地址指向）以填充该调用的 H 和 W 参数指定大小的矩形区域。在该阶梯像素图结构的标志字段中定义的格式定义该阶梯数据的格式。

重复因子 (repeat factor)：在 GL 中线型模式使用的扩大性。

重力 (gravity)：指“增强的 X-Windows”中窗口或子窗口的内容对窗口内的某个位置具有吸引力。这确定了窗口标识重新调整大小的方法。另见位引力 (*bit gravity*) 和窗口引力 (*window gravity*)。

重命名声明 (renaming declaration)：在 Ada 语言中，声明一个实体的另一个名称。

重试 (retry)：再次尝试导致设备错误消息的操作。

重新绑定 (rebinding)：在通信通道被关闭后为进行远程过程调用重新建立通信通道。

重新发送 (retransmit)：重复消息或消息段的传输。

重新启动标识包 (restart-indication packet)：在 X.25 通信中，DCE 发送的一种调用监视包，向 DTE 表明已接收重新启动请求。

重新启动请求包 (restart-request packet)：在 X.25 通信中，DTE 发送的一种呼叫监视包，请求重新启动一条链路。

重新启动确认包 (restart-confirmation packet)：在 X.25 通信中，DCE 发送的一种调用监视包，确认已重新启动链路。另见包 (*packet*)；信息包；分组。

重新引导 (reboot)：通过重复初始程序装入 (IPL) 操作来重新初始化程序的执行。

重载 (overloading)：(1) 在 Ada 语言中，一个标识符在程序文本中的给定点可以有几个可供选择的意思：该特性称作重载。例如，重载的枚举文字可以是两个或更多的枚举类型标识符中出现的标识符。重载的有效意义通过上下文来确定。子程序、聚集、分配算符和字符串文字也可被重载。(2) 在 C++ 里，当函数和运算符和类类型一起使用时，它具有能力允许您重新定义功能和大多数的标准 C++ 运算符。

抽象 (数据) (abstraction, data)：一种数据类型，具有私有表示和公有操作集。

抽象语法符号一 (Abstract Syntax Notation One, ASN.1)：一种能够同时定义复杂类型并给这些类型指定值的符号。

抽样率 (sample rate)：定位器抽样率 (*locator sample rate*) 的同义词。

出队 (dequeue)：从队列删除项目。对照加入队列 (*enqueue*)。

初始程序装入 (initial program load, IPL)：(1) 导致一个操作系统开始运作的初始化过程。与系统重新启动 (*system restart*) 和系统启动 (*system startup*) 同义。(2) 在开始时或者一个工作日或者一个系统故障时，通过该进程一个配置映像被装入存储器。(3) 装入系统程序和为一个系统运行作业准备的进程。

初始化程序装入设备 (initial program load device, IPL device)：通过输入/输出设备，通常是一个固定磁盘或者软盘，当计算机打开时系统软件被装入。

初始化算子 (initializer)：后面跟着一个表达式或多个变量聚集的表达式赋值操作符。

初始化 (initialize)：(1) 在编程语言中，设置一个数据对象的开始值。(2) 在一个计算机例程的开始、或中间预定义的点上、或操作上设置计数器、开关、地址、或者存储内容为零或其他开始值。(3) 准备使用，例如初始化一个软盘。(4) 准备系统操作。在将内核装入内存后，系统运行内部检测，初始化所有的内存和一些设备并分析根文件系统。

初始序列控制器 (initial sequence controller)：初始程序装入 (IPL) ROM 的三个控制程序之一。在硬件初始化和发送控制到核心序列控制器之后，初始序列控制器就接受了控制。

除去 (remove)： 对于软件选项，删除选项和其应用的或已提交的全部都从系统更新。软件重要产品数据 (SWVPD) 信息被更改以表明该选项已从该系统除去。根据选项，系统配置信息也被清理，尽管这不总是完全的。如果前一版本、发行版或选项的级别在系统上，系统将不恢复前一版本。只能除去一个选项及其更新。更新不能由自己除去。对照提交 (commit)。

处理程序 (handler)： 控制某个程序对特定外部事件的反应的软件例程，例如中断处理程序。对于 Ada 编程，请参阅异常 (exception)。

处理代理程序 (processing agent)： 应用程序服务器中的一种线程，它处理来自客户机的远程过程调用。

处理单元 (processing unit)： 计算机中的一种功能单元，它对处理的某个方面负责。

处理器亲缘性 (processor affinity)： 线程可能被分派到与上次执行所在的同一物理处理器的程度。

穿过方式 (pass-through mode)： VM/Pass-Through Facility 提供的使用方式，允许 VM 显示站用户交互地访问 VM 系统，包括终端连接的那个 VM 系统。它还允许用户访问非 PVM 系统，该系统支持远程 3270 二进制同步通信 (BSC) 显示站和有远程操作员控制台设施 (ROCF) 的 4300 处理器。用户可在定义的网络中访问、登录和使用另一个系统，就好像用户的本地终端直接连接到该系统一样。PVM 活动对已登录到目标系统的用户变得透明。

传播时间 (propagation time)： 信号在通信线路上从一点传送到另一点所需的时间。

传递功能 (pass-through function)： 通过程序透明地传递数据 (不更改) 的能力。

传输层 (Transport Layer)： 当隐藏通信网络的详细信息时，在两端提供端到端通信的网络服务。TCP 和 ISO TP4 传输协议提供全双工虚拟电路，此电路上的传输是可靠、无错、有序且无重复的。UDP 不提供保证 (无连接 RPC 协议在 UDP 顶层上提供一些保证)。

传输服务 (transmission services, TS)： SNA 中，要求特定对话支持传输控制 (TC) 协议的会话激活请求 (比如会话级别调拨会话级别请求的使用) 中的规范。每个定义的传输服务请求都由数字标识。

传输控制协议 / 网际协议 (Transmission Control Protocol/Internet Protocol, TCP/IP)： 一种属于行业标准、非私人所有的通信协议组，此协议允许来自不同生产商的设备之间的连接行为。“国防部高级研究计划署”支助此协议的开发。允许您建立局域网或广域网的通信子系统。

传输控制协议 (Transmission Control Protocol, TCP)： 用于 ARPA 因特网或任何遵守美国国防部网际协议标准的网络的通信协议。它在包交换通信网络以及这些网络的互连系统之间提供可靠的“主机至主机”协议。它假定“网际协议”是基础协议。

传输控制字符 (transmission control characters)： 包含在消息中用来控制数据链路上通信的特殊字符。例如发送站和接收站使用传输控制字符来交换信息。接收站使用传输控制字符来指定它接收到的数据中的错误。

传输脉冲串 (transmit burst)： 没有接收干扰和超时操作而发送的一组传输包。

传输协议 (transport protocol)： 来自 OSI 网络体系结构的传输层的通信协议，比如传输控制协议 (TCP) 或用户数据报协议 (UDP)。

传送 (transfer)： 从一个地方发送数据，并从另一个地方接收数据。

串行处理 (serial processing)： 指在单个设备中顺序或连续运行两个或更多进程，比如通道或处理单元。对照并行处理 (parallel processing)。

串行传输 (serial transmission)： 在同一条电路上分开传输每位数据字符。

串行化 (serialize)： (1) 将按位并行 (parallel-by-byte) 更改成按位串行 (serial-by-bit)。 (2) 在 XDR 中，指将特定机器表示转换为 XDR 格式。

串行设备 (serial device)： 顺序执行功能的设备，比如在一次打印一个字符的串行打印机。对照并行设备 (parallel device)。

串口端口 (serial port)： 用于串行设备的端口。另见串行设备 (serial device)。

窗格 (pane)： 在显示屏幕中，窗口的内部用来显示给用户的信息。一个窗口可以由一个或多个窗格组成。另见菜单窗格 (menu pane)。

创建操作 (Create Action)： CDE 中的软件应用程序，允许将图标与命令相关联，以便可以通过单击图标发出该命令。创建操作 (Create Action) 还用于定义应用程序数据文件的特定数据类型，并将图标与这些数据类型相关联。

创建日期 (creation date)： 文件创建日期。另见会话日期 (session date) 和系统日期 (system date)。

窗口标识 (window ID)： 与在 AIXwindows 或“增强的 X-Windows”环境中新打开窗口相关的唯一标识号码。

窗口菜单按钮 (Window menu button) : CDE 中在窗口左上角的控件, 位于标题栏的旁边。选择它就显示窗口菜单。

窗口菜单 (Window menu) : CDE 中通过选择“窗口”菜单按钮显示的菜单。此菜单提供了操作窗口位置或大小的选项, 比如移动、大小、最小化和最大化。

窗口管理器 (Window Manager) : CDE 中提供用户在工作区操作窗口的能力 (比如打开、移动、关闭窗口以及调整窗口大小) 的应用程序。

窗口管理器 (window manager) : 在公共桌面环境中显示和控制屏幕上窗口的程序。管理多个与 AIXwindows 和“增强的 X-Windows”相关的窗口的软件。在图形环境中, 指在屏幕上操作窗口并提供用户界面大部分的客户机。另见 *mwm*。

窗口图标 (window icon) : CDE 中最小化的窗口。

窗口引力 (window gravity) : 吸引子窗口到其父窗口某部分的引力。当重新调整子窗口大小时, 窗口引力使子窗口相对于边缘、角落或窗口中心自动重新定位。与小窗口引力 (*widget gravity*)。另见引力 (*gravity*)。

窗口 (window) : (1) AIXwindows 中屏幕上的矩形区域, 用户可以移动它, 将它放置于其它窗口之上或拉到其它窗口之下, 或将它图标化。(2) GL 中窗口内的所有绘制都由创建该窗口的 GL 进程进行, 且完全处于该进程的控制。然而, 窗口边界和窗口布局/图标的绘制由窗口管理器控制; 例如“AIXwindows 窗口管理器”。对于大多数简单的 GL 程序, 视口和屏幕屏蔽区设置为与窗口相同大小。不要将 AIXwindows 子例程和 GL 窗口子例程混淆, 后者定义自然空间中的立体。另见剪切 (*clipping*) 和当前窗口 (*current window*)。(3) curses 和扩展 curses 中有关屏幕某部分在某个时间点上如何显示的内部表示。窗口可以为任何大小, 从全屏到单个字符。(4) 数据通信中, 在 DTE 或 DCE 等到发送另一个数据包的授权之前, 它们可以在逻辑通道上发送的数据包的数量。窗口是包的调步或流控制的主要机制。另见帧窗口 (*frame window*) 和信息包窗口 (*packet window*)。

垂直回扫周期 (vertical retrace period) : 各次屏幕回扫之间经历的时间。所有的视频监视器表面使用一条电子光束来扫描监视器上的磷光体。因为磷光体仅发光很短的时间, 电子光束必须对整个屏幕进行定期的重扫。在大多数监视器上, 每秒 30 次 (30 Hz) 执行该操作。这样, 垂直回扫的周期为 1/30 秒。

垂直回扫 (vertical retrace) : (1) 监视器被刷新的速率。60 Hz 监视器每秒被重绘 60 次。与刷新率 (*refresh rate*) 同义。(2) 从屏幕底部向顶部移动电子光束的操作。

纯虚函数 (pure virtual function) : 声明虚拟函数为“纯”, 方法是用“=0;”替换函数定义。

磁带机 (tape drive) : 移动磁带并控制其移动的机械装置。

磁道 (track) : (1) 固定磁盘或软盘表面上的环形路径, 信息被磁性记录于这些路径上且从这些路径读取记录的信息。(2) 与单个读或写组件相关的数据介质上的路径, 此时数据介质移动经过此组件。

词法单元 (lexical element) : 在 Ada 语言中的一个标识符、文字、定界符、注释。

词法分析器 (lexical analyzer) : 一个分析输入并将输入分解成各类别比如数字、字母或运算符等的程序。

词法级别 (lexical level) : 一个例程嵌套在另一个例程中的嵌套深度, 它决定了那些例程中声明的标识符的作用域。

词法作用域 (lexical scope) : 程序或段单元种一个声明起作用的部分。一个在例程中声明的在该例程和其中的所有嵌套例程中已知的标识符。如果一个嵌套例程用同样的名称声明了一项, 则在该嵌套例程中外部项是不可用的。

词汇表窗口 (glossary window) : 包含词汇表术语的窗口。

磁盘操作系统 (Disk Operating System) : 请参阅 *DOS*。

磁盘驱动器 (disk drive) : 用来读写磁盘上信息的设备。

磁盘使用情况记帐 (disk-usage accounting) : 由用户文件占用的磁盘块的数目的记录。磁盘使用情况记帐是由 *acctdisk* 命令执行的。

磁盘适配器 (disk adapter) : 计算机用来访问并控制磁盘驱动器的硬件。

磁盘 I/O (disk I/O) : 固定磁盘的输入和输出。

磁盘 (disk) : 一种存储设备, 由一个或多个平面、有磁性表面的圆形盘片组成, 可以在上面存储信息。

次模式 (subpattern) : 正则表达式 (*regular expression*) 的离散元素。

次选表示方式 (secondary representation) : 第二格式, 它是主选表示方式的备用, 其中客户机可以为此服务提供属性值。

次要设备号 (minor device number) : 指明有关一个特定设备的各种类型信息的号。例如, 一个区分几台同类型打印机的号。另见主要设备号 (*major device number*)。

从属工作站 (dependent workstation)： 单机功能很少或没有单机功能的工作站，必须连接到主机或服务器，才能为用户提供任何实质性的功能。

粗略剪切 (gross clipping)： 在 GL 中也称为 3-D 剪切，指在标准化设备坐标中出现的剪切，与透视截锥体的侧面相对应。所有 3-D 图元都要经过这种剪切；尤其是文本字符串（但不是单个字母）的原点要以该方式剪切。另见 *剪切 (clipping)*、*精细剪切 (fine clipping)* 和 *剪切平面 (clipping plane)*。

粗体字 (boldface)： 加粗类型，通常为常规文本字体的加粗版本。也包括以此类型打印的处理。另见 *双重突出打印 (double-strike)* 和 *强调 (emphasized)*。

存储类说明符 (storage class specifier)： 存储类关键字。以下 C++ 关键字之一：**auto**、**register**、**static** 或 **extern**。

存储区计数 (bucket counts)： 元素数目大于或等于最低下限并且小于或等于较高上限。

存储区 (bucket)： 一个或多个累计操作结果的字段。

存储设备 (storage device)： (1) 用于存储和检索数据的功能单元。(2) 数据可以保留其中的设施。

存储 (storage) 存储器；存储量： (1) 保存的信息的位置。(2) 不同于内存，在物理设备（比如磁盘或磁带）上进行的信息的保存。另见 *内存 (memory)*。(3) 可以在其中输入、保留和处理已记录文本的单元，并可从此单元检索已记录文本。(4) 将数据放入存储设备的操作。

存储 (store)： 将信息放置于存储设备（在内存中或软盘、固定磁盘或磁带上），这样信息就可用于检索和更新。

存取 (access)： 从存储器获得数据或把数据放入存储器中。

错误标识符 (error ID)： 请参阅 *错误标识符 (error identifier)*。

错误标识符 (error identifier, error ID)： 一个用来标识特定错误的 8 字符代码。对每个错误记录模板都有一个唯一的错误标识符。

错误分析程序 (error analysis facility)： 提供有关错误可能的原因信息的程序。

错误记录模板 (error record template)： 描述错误日志项的错误类别、错误类型、错误描述、可能原因、推荐操作和错误数据。

错误计数器 (error counter)： 由设备驱动程序部件产生的一种错误项。如果一个操作在第一次尝试时不成功，某

些设备驱动程序能够产生重试操作。它们使用计数器来监视重试操作的次数和原因，并包含决定在什么时候应把计数器送到错误日志的算法。

错误检测和更正 (error-checking and correction, ECC)： 指在处理单元中，对所有单个位错误的检测和纠正，再加上对双位和某些多位错误的检测。

错误类别 (error class)： 标识一个错误日志项是属于硬件还是软件错误。

错误类型 (error type)： 标识一个错误日志项是属于永久故障、临时故障、性能降低、可用性的迅速损失还是未定错误。

错误日志项 (error log entry)： 系统错误日志中的记录，它描述了一个硬件或软件错误，并包含在错误发生时捕获的错误数据。

错误日志 (error log)： (1) 产品或系统中存储错误信息用于以后访问的数据集或文件。(2) 维护库中的一个表格用于记录有关产品或系统的错误信息。(3) 处理器中用于记录某些硬件和程序事件信息的数据集。(4) 机器检测、设备错误和卷统计数据记录。

错误设备驱动程序 (error device driver)： 一种由错误记录程序使用的特殊文件（伪设备）驱动程序。错误项由 **errlog()** 子例程和 **errsave()** 内核服务写到错误设备驱动程序。错误项由错误守护程序进程从错误设备驱动程序中读出并保存到系统错误日志文件。

错误输出 (error output)： *诊断输出 (diagnostic output)* 的同义词。

错误消息 (error message)： 检测到错误的指示信息。

错误状态 (error condition)： 由于试图在计算机程序中运行无效的指令或者对数据的操作非法而产生的状态。

[D]

打包 (packing)： 在 AIXwindows 和“增强的 X-Windows”中，在父容器对象内进行子对象分组。如果子对象紧密大包，在它们的边缘间的公共距离最小，而边缘到边缘的公共距离最大。

打开 (open)： 使文件对程序可用或可用于处理。对照关闭 (*close*)。

打印队列 (print queue)： 包含等待将打印的文件的名称列表的文件。

打印服务器 (print server)： 在 CDE 中，连接一个或多个打印机的主机计算机，或管理这些打印机的 UNIX 进程。

打印管理器 (Print Manager): 在 CDE 中, 显示系统上所有打印机的软件应用程序。

打印机会话 (printer session): 在与一个仿真 3286/87 打印机的主机计算机交互期间, 3270 主机连接程序 2.1 和 1.3.3 (HCON) 的操作方式。

打印机控件 (Printer control): 在 CDE 中, 用于启动打印机软件应用程序的前面板控件。在控件上放下一个文件将显示一个对话框, 您可用该对话框打印文件到缺省打印机。

打印机作业 (Printer Jobs): 在 CDE 中, 提供关于一个打印机上作业的信息的软件应用程序。

打印机 (printer): 从外部连接到系统单元的一种设备, 用于在纸上打印系统输出。

打印设备 (printing device): 任何打印机或打印的其它设备, 如类似打印机的设备或绘图仪。

打印输出 (printout): 计算机中打印机产生的信息。

大括号 (brace): (1) 字符 { (左括号) 和 } (右括号), 也称为“波形括号”。ASCII 字符是 C 编程语言 (编写操作系统的语言) 的主要符号。(2) Pascal 语言中的注释定义符。

大尾数法顺序 (big endian order): 整数值首先存储最重要字节的存储方法。另见小尾数法顺序 (*little endian order*)。

大尾数法 (big endian): 数据表示的一个属性, 反映了多个八位元 (multi-octet) 数据在内存中是如何存储的。在大尾数法表示中, 多个八位元数据项的最低地址八位元是最重要的。另见字节排列顺序 (*endian*) 和小尾数法 (*little endian*)。

大小字段 (size field): i-node 中, 指明与 i-node 相关联的文件大小 (以字节为单位) 的字段。

大小 (size): 更改窗口大小的屏幕管理操作。

大写字母 (caps): (1) 大写字母, 大写字体。(2) 一种打印样式, 对单一的大写字体使用两种字号尺寸。使用小号尺寸而不是小写字体。

大型机 (mainframe): 大型的计算机, 主要是和其它计算机相连、使它们共享大型机设施的计算机。此术语通常仅指硬件。

大字 (big word): 在 vi 编辑器中, 首尾由空格、制表符或换行指示符限定的相邻字母数字字符集。另见小字 (*small word*)。

带冲突检测的载波监听多路访问 (carrier sense multiple access with collision detection, CSMA/CD): 一类介质访问过程的通用术语, 允许多个站随意访问介质 (无需显式的事先协调), 通过载波监听和区分的方法避免争用。通过冲突检测和传输解决争用。

带符号的 (signed): 通过将其附加至信息的加密摘要中, 则信息带数字符号。这用来确保数据的完整性, 始发者的可靠性以及始发者与数据间的模棱两可的关系。

代号 (generation number): 用来区分对几个文件或同一文件的不同版本使用同一 inode 的号码。在文件创建时, inode 便分配给该文件, 并且新代号存储在该文件的 inode 中。任何对具有错误代号的 inode 的使用都表示此文件的用户视图已过时, 或是“失效”。

带宽 (bandwidth): 数据速率, 以千位、千字节、兆位和兆字节 / 秒传输。

代理 (broker): NCS 中管理有关对象和到该对象的接口的信息的服务器。希望成为接口客户机的程序可以使用代理获得有关导出接口的服务器的信息。另见位置代理 (*Location Broker*)。

代码点 (code point): 代码页中的字符。另见代码页 (*code page*) 和扩展字符 (*extended character*)。

代码段 (code segment): 请参阅段 (*segment*)。

代码服务器 (code server): 一个为网络上其它计算机提供代码服务的系统。

代码集 (code set): XPG4 系统接口中, 一个建立字符集的明确规则以及字符集中每个字符和其位表示之间的一对一关系。

代码页 (code page): (1) 一种图形字符指定, 控制所有代码点的功能含义。(2) 代码点阵列表示建立字符序数列 (数字顺序) 的字符。操作系统使用 256 字符的代码页。另见代码点 (*code point*) 和扩展字符 (*extended character*)。(3) 一个多达 256 个预定义显示符号的有序集。每种代码页的前 32 个码点是保留控制码, 对所有代码页都一样, 每种代码页剩下多达 224 个不同的显示符号。

代码 (code): (1) 计算机指令。(2) 为计算机写指令; 编程。(3) 条件陈述, 例如错误代码。

代 (generation); 生成: (1) 按照时间和祖先来引用项的方法, 使得没有组先的项被指定为第一 (第 n) 代, 而随后派生的项指定为 n-1、n-2 等等。(2) 对于某些远程系统, 指将配置转换成机器语言。

单处理器 SMP (single-processor SMP): 设计用来处理两个或更多处理器的系统, 它运行 SMP 版本的操作系统, 此系统已经通过单处理器配置。对照单处理器 (uniprocessor)。

单处理器 (uniprocessor): 包含单个处理器的系统。当用于此书时, 短语“可比单处理器”意指设计只具有单个处理器的系统, 此系统具有如讨论过的 SMP 系统一样的 CPU 时钟和高速缓存容量, 并且运行操作系统的单处理器版本。对照单处理器 SMP 系统 (single-processor SMP system)。

单缓冲区方式 (single buffer mode): 在 GL 中, 指帧缓冲区位平面组织成单个大型帧缓冲区的方式。此帧缓冲区是当前显示的缓冲区, 并且还是所有绘制发生的缓冲区。另见双缓冲区方式 (double buffer mode)。

单机工作站 (standalone workstation): 可以执行任务而不必连接至其它资源 (比如服务器和主机系统) 的工作站。

单机系统 (standalone system): 请参阅单机工作站 (standalone workstation)。

单机 shell (standalone shell): 用于系统维护的 shell 限制版。

单击 (click): CDE 中不移动鼠标指针按下并释放鼠标。除非另外指定, 否则假设为鼠标左键。

单机 (standalone): 网络安装环境中的一台机器, 它在本地访问所有所需资源。

单精度 (single-precision): (1) 依照所需精度, 使用一个计算机字来表示一个数字。(2) 引起在短格式中存储浮点值的规范。另见精度 (precision)。

单模光纤 (single-mode optical fiber): 一种光纤, 其中只有最低顺序绑定方式 (由一对正交极化区构成) 能够在长波上传播。对照多模光纤 (multimode optical fiber)。

单切换控制 (single-shift control): 在代码页切换中, 对于单个字符切换至另一页的控制代码; 无锁定切换。

单色的 (monochrome): 一个仅有两种颜色映射入口的特殊的静态灰度的情况。一些单色适配器可以在灰度适配器 (Gray Scale Adapter) 上显示灰色阴影。

单色显示器 (monochrome display): 仅有一种颜色的显示设备。另见灰度 (gray scale)。

单向打印 (unidirectional printing): 一种打印方法, 其中打印机上的打印头只在它向一个方向移动时打印, 而在它向相反方向移动时打印。这种打印方法通常可以产生更高质量的打印输出。

单向通道 (one-way channel): X.25 通信中, 逻辑通道允许单独进入呼叫或单独呼出呼叫。对照双向通道 (two-way channel)。

单选按钮 (radio button): 表明选项的固定设置。只能一次选定设置中的其中一个按钮。一种旁边有文本的圆圈, 当某个选项被选中时, 该圆圈被部分填充。

单选框 (radio box): 行列管理器小窗口内切换按钮的特殊配置。

单元标识 (unit identifier): FORTRAN 中指定外部的单元或内部文件的数字。此数字可以为以下数字之一: 值必须为零或正数的整数、对应单元 5 (输入) 或单元 6 (输出) 的 * (星号), 或内部文件的字符数组、字符数组元素或字符串的名称。

单元立方体 (unit cube): GL 中以下平面定义的体: $x = -1, x = 1, y = -1, y = 1, z = -1, z = 1$ 。另见规格化设备坐标 (normalized device coordinates)。

单元相关名称 (cell-relative name): 请参阅本地名称 (local name)。

单元 (cell): 水平行与垂直列的矩形结合点。例子有电子表格的单元和 AIX windows 图形界面中小窗口 XmRowColumn 利用的单元。在 Interleaf 中, 表格单元实际上是可以包含数据的已修改的框架。

单元 (unit): (1) FORTRAN 中, 表示使用输入/输出语句的文件的方法。一个单元可以连接或不连接至文件。如果连接了, 它就指该文件。连接是对称的; 也就是说, 如果一个单元连接至文件, 文件就连接至该单元。(2) 另见可编译单元 (compilable unit)。(3) 对于 Ada 编程, 它是编译单元 (compilation unit) 的同义词。

单字节控制代码 (single-byte control codes): ASCII 码 0 到 31 (0x00 到 0x1f) 和删除码 (0x7f)。

单 (ticket) 凭单; 票券: 一种将在启动中的主体的身份传输至其目标的应用程序透明机制。一张简易凭单包含主体身份、会话密钥、时间戳记和其它信息, 且使用目标密钥密封。特权凭单包含和简易凭单相同的信息, 还包含特权属性证书。凭单许可凭单是用于凭单许可服务的凭单; 服务凭单是用于除了凭单许可服务以外的服务的凭单。

当前变换矩阵 (current transformation matrix): 位于矩阵堆栈顶部的变换矩阵。所有通过图形流水线传递的点在传送都乘以当前变换矩阵。当前变换矩阵是当前建模和视图矩阵的串联。另见变换 (transformation) 和矩阵堆栈 (matrix stack)。

当前窗口 (current window): 系统要将图形例程输出定位至的窗口。另见窗口 (window)。

当前堆 (current heap)： Pascal 中，对 NEW 的调用所分配的动态变量驻留的存储区域。同时可存在其它堆，但只有一个是当前的。

当前工作目录 (current working directory)： 与当前目录 (current directory) 同义。

当前行 (current line)： 光标所在行。

当前会话 (current session)： CDE 中，注销时由“会话管理器”保存的会话。下次登录时，除非另行指定，否则该会话自动打开，使您可以在上次停止的地方继续工作。对照本地会话 (home session)。

当前记录指针 (current record pointer, CRP)： 当顺序处理 SFS 文件或从这些文件选择的记录范围时使用的逻辑指示符。CRP 跟踪选中范围内的哪个记录刚刚被处理，接着将处理哪个记录。

当前记录 (current record)： (1) 当前行指针所指向的记录。(2) 对程序当前可用的记录。

当前目录 (current directory)： 活动并且可用 pwd 命令显示的目录。从当前目录开始的相对路径名称解析。与当前工作目录 (current working directory) 和工作目录 (working directory) 同义。

当前时间 (currency time)： 用户读取新闻项的时间。news 命令仅考虑此时刻 (用户的当前时间) 之后投递的项。

当前图形窗口 (current graphics window)： 在 GL 中，系统要将图形例程输出定向到的窗口。

当前图形位置 (current graphics position)： 齐次三维点，几何绘制命令从该点开始绘制。当前图形位置无需可视。

当前文件夹 (current folder)： CDE 中，在活动的文件管理器视图中当前正在打开的文件夹。

当前文件 (current file)： (1) 正在编辑的文件。如果有多个窗口正在使用，则包含光标的文件是当前文件。(2) make 命令中，在一给定时刻 make 命令正在处理的文件。make 命令用宏 \$* 替代当前文件的名称。

当前选择 (current selection)： 一个突出显示的文本块或元素。

当前颜色 (current color)： 所有后继绘图图元采用的颜色。所有绘图图元以这种颜色绘制，直至颜色有更改。

当前主机 (current host)： 与本地主机 (local host) 同义。

当前字符位置 (current character position)： 下一个将要发生的字符串或像素读 / 写操作所在的二维屏幕坐标。

当前 (current)： 对于 Ada 语言编译单元，自单元自身编译之后，没有该单元的支持者 (supporter) 重新被编译的状态。这意味着为了编译程序所有支持者都要存在 (既然 Ada 语言要求这样)。当前意味着所有单元支持者以按 Ada 语言规则定义的正确次序编译。

导出 (export)： (1) 在 NCS 中，为了提供由接口定义的操作。服务器把接口导出到客户机。对照导入 (import)。(2) 把数据拷贝到可移动介质上。

导航键 (navigation keys)： 在 CDE 中，用于移动目前光标位置的键盘键。包括箭头键 (带或不带 Control 键)、Tab 键 (带或不带 Control 键或 shift 键)、Begin 键和 End 键 (带或不带 Control 键) 与 Page Up 和 Page Down 键。

倒排索引 (inverted index)： 书刊目录数据库的索引，允许直接访问。这可以与词典的翻页进行对比，即允许用户移到所要搜索字的准确位置。

导入项 (imports)： 在 Ada 语言中，命名在某个库单元 (在 with 子句中声明) 的上下文说明中的所有编译单元，以及由编译器引入的所有支持单元说明，例如标准软件包 SYSTEM。次要单元的导入项是在其上下文说明中命名的所有单元、由编译器引入的所有支持单元要求；其关联的父单元 (如果有) 以及包含在其它导入项中的所有类属说明的主体。

导入 (import)： (1) NCS 中请求由某个接口所定义的操作。客户机从服务器导入接口。(2) 从另一个系统引入数据。请与导出 (export) 对照。

登录会话 (login session)： 工作站用户和交互式系统进行通信的时间段，通常是登录和注销之间所消耗的时间。

登录名 (login name)： 由系统唯一识别一个用户的字符串。

登录目录 (login directory)： 当第一次登录系统所访问的目录。

登录 shell (login shell)： 用户登录到计算机系统时启动的 shell。为特定用户的登录 shell 决定于该用户的一个项 (位于 /etc/passwd 文件中)。另见 shell。

登录 (log in)： (1) 在一个显示站上开始一个会话。(2) 通过在工作站上输入标识和认证信息从而获得对一计算机系统的访问的操作。

登录 (log on)： 请参阅登陆 (log in)。

登台程序 (stage): 一系列步骤之一, 输入通常以 `< cr` (> (回车) 结束的 `ged` 命令。每个子命令由登台程序的子集 (包含命令行、文本、点、主元素和目标) 构成。

等待时间 (latency): 从一个操作的开始一直到某些实际事情开始发生 (例如, 数据传输开始) 的时间。

等价类 (equivalence class): 为了整理的目的, 被认为相同的字符或字符串组成的分组。例如, 为了整理的目的许多语言把大写字符与它的小写形式放在同一个等价类中, 但某些语言区分重音和非重音字符形式。

等宽字符 (constant-width characters): 一种字符集, 设计为每个字符和其它字符具有同等宽度。

低阶 (low-order): 最不重要的, 最右边的。示例, 在 32 位寄存器中, (从 0 到 31), 位 31 就是低阶位。

底端阴影 (bottom shadow): 在 AIXwindows 中, 一个穿过矩形图像对象 (小窗口或小配件) 底部的狭窄黑色带, 当操作对象时创建三维外观。

递归属性 (recurring attribute): 有几个属性值的一个属性。

递归 (recursion): (1) 使用某种函数定义自己。(2) 执行一种多步骤的操作, 其中每一步均利用上一步的输出。

递送者 (courier): DTS 中的一个本地服务器, 每次同步时从一个随机选中的全局服务器请求时间值。

地址解析协议 (ARP) (Address Resolution Protocol, ARP): TCP/IP 提供的协议之一, 在局域网里的因特网地址、基带适配器地址、X.25 地址和令牌环适配器地址之间进行动态映射。

地址空间 (address space): 进程可访问的代码、堆栈和数据。

地址扩展 (address extension): 在 X.25 通信中, 被叫和主叫地址扩展是可选的 CCITT 指定的设备, 可用于符合 X.25 (1984 版) 的网络。与扩展地址 (*extended address*) 同义。

地址列表 (address list): 命令 `xtalk` 使用此列表将用户名与网络用户地址和其它信息相关联, 使主叫无需知道地址就可进行呼出 X.25 呼叫。系统和每个用户各有一个地址列表。另见系统地址列表 (*system address list*) 和用户地址列表 (*user address list*)。

地址系列 (address family): 请参阅协议系列 (*protocol family*)。

地址转换寄存器 (ATR) (Address Translation Register, ATR): 一种把实际地址转换为虚拟地址的机制。

地址字段 (address field): 数据包的一部分, 其中包含寻址信息。另见包 (*packet*); 信息包; 分组。

地址 (address): (1) 远程系统用来呼叫该系统的电话号码。(2) 按其地址引用设备或数据项。(3) 在文字处理中, 由地址代码确定的记录介质或存储器的特定区域的位置。(4) 在数据通信中, 指定给连接到网络的每个设备或工作站的唯一代码。另见网络用户地址 (*network user address*)。(5) 网络通信中用于识别与其通信的特定网络或主机的编码系统。地址通常以点分十进制的格式表示。另见地址表示 (*presentation address*)。

点处理 (point-handle): 图形对象中的点。

点到点链路 (point-to-point link): 将单个远程链路站连接到一个节点或另一个站的交换式链路或非交换式链路。

点分十进制 (dotted decimal): 因特网主机地址的常用表示法, 将 32 位地址分成 4 个 8 位的段。每段的值用一个十进制数指定, 段与段之间用点号分隔 (例如, 010.002.000.052 或 10.2.0.52)。

点图 (stipple): 用于平铺显示某个区域的位图。点图模式用作用前景色进行填充操作的附加的剪切屏蔽区。

点阵 (dot matrix): 使用点阵打印字符的线式或其它方式的打印机。

点 (dot): 符号 (.) 在相对路径名中表示当前路径。

点 (point): (1) 在排版中, 用于度量的单位, 等于 0.01384 英寸 (0.3505 毫米) 或大约 1/72 英寸。1 pica 等于 12 点。(2) 在 CDE 中, 移动鼠标直到指针落在特定屏幕元素或区域上。

电磁兼容性 (electromagnetic compatibility): 对产品的设计和测试, 通过处理在射频频谱中的发射和可疑的频率来满足法律和公司的规范。电磁兼容性是各种电子设备预期的电磁环境正确运作的能力。

电缆 (cable): 传输信号的物理介质; 包含铜导体和光导纤维。

电路交换 (circuit switching): 一种按需要把两个或更多个数据终端设备 (DTE) 相连接并在连接释放之前允许它们之间独占使用数据电路的处理。与线路交换 (*line switching*) 同义。另见包交换技术 (*packet switching*)。

电路 (circuit): 请参阅虚拟电路 (*virtual circuit*)。

电源需求 (power requirement): 计算机系统消耗的实际能量, 用瓦度量。

电源因子 (power factor): 消耗功率与伏安培 (实际功率) 之比。

电源指示灯 (power-on light)： 操作程序屏面上的一种指示灯，表明系统单元的 DC 电源正在工作。

电源 (power source)： 电路的最小可接受的评级，提供功率给计算机系统，用伏安培 (kVA) 度量。

电子邮件 (electronic mail)： 与邮件 (mail) 同义。

调步响应 (pacing response)： 在 SNA Server 中，表示接收组件接受另一个步调组准备就绪的指示符。指示符携带在会话级别调步的响应报头 (RH)，以及虚拟路由调步的传送报头 (TH)。

调步 (pacing)： (1) 接收组件使用的技术，用来控制传送速率防止发送组件过速。(2) 有些系统要求的文件传输协议。通过等待特殊字符或等待行之间的特定秒数来控制数据传输。该协议防止在块太大或送往系统处理的数据速率太快时的数据丢失。另见行调步 (line pacing)、本地调步 (local pacing)、接收调步 (receive pacing) 和远程调步 (remote pacing)。

调度策略： 支配何时线程失去 CPU 控制以及随后哪个线程得到控制的一套规则。

调节 (conditioning)： (1) 使用指示符控制何时执行计算或输出操作。(2) 数据通信中向非交换语音通道添加设备以提供数据传输所需的线特征最小值。

调色板 (color palette)： 与色彩图 (color map) 同义。

调色板 (palette)： (1) 定位构建定制组件和与其它组件一起的生成。随后，通过拷贝和移动组件到其它接口来重用组件。(2) 在 CDE 中，图形的范围显示可以在应用程序中选定的选项，比如颜色或工具集。

调试方式 (debugging mode)： 一种特殊方式，其中程序提供自身活动的详尽输出，帮助用户检测和更正程序自身或程序或系统配置中的错误。

调试器 (debugger)： 计算机程序或软件中用于检测、跟踪及消除错误的一个或多个程序。

调试 (debug)： 在配置计算机系统或软件程序过程中检测、定位并更正错误。

调试 (debugging)： 在软件或系统配置中检测并更正错误。

调页空间 (paging space)： 驻留在虚拟内存中但当前不访问的信息的磁盘存储。

调页设备 (paging device)： 用来存储当前不在实际内存中的内存页面的磁盘。

调用标识 (invoke ID)： 一个整数用于区分从所有其它不同操作中的一个 (目录) 操作的整数。

调用堆栈 (invocation stack)： 链接在一起的程序列表，作为在相同的作业中程序调用其它程序的结果。与程序堆栈 (program stack) 同义。

调用树 (calling tree)： 程序控制树。主过程是调用树主干；任何由主过程引用的过程从主干分支，任何由另一个过程引用的过程是那一个过程的分支。不引用其它过程的分支称为叶。

调用 (call)；呼叫： (1) 通常通过指定入口条件然后跳转到入口点来激活程序或过程。另见装入 (load)。(2) 数据通信中，在交换线路上的两站之间建立连接的必要操作。(3) X.25 通信中两个用户之间的对话。

调用 (invoke)： 启动一个命令、过程或程序。

调整大小边框 (resize border)： 在 CDE 中的一种控件，用于更改窗口大小或窗口中窗格大小。

调整 (adjust)： 移动文本以适合左右页边距的过程。

调制解调器代用器 (modem eliminator)： 用于通过一个线路连接器和一个特定的引脚排列连接工作站和计算机端口的设备。当两个设备都用作 DTE (数据终端设备) 时，连接它们的电缆必须用调制解调器代用器来传输发送和接收信号。与零调制解调器 (null modem) 同义。

调制解调器 (调制器 - 解调器)

(modem, modulator-demodulator)： 将来自计算机的数字数据转化为电信线路上可以传输的模拟信号、将接收到的模拟信号转化为计算机上的数字数据的设备。

调制器 - 解调器 (modulator-demodulator)： 请参阅调制解调器 (modem)。

调制 (modulation)： 通过用一个信号的频率或大小来改变另一个信号的频率或大小。

叠加 (fold)： (1) 通过合并数据的组成部分来压缩数据，例如，通过添加这些字母的数字等价物，变换两个字的数字密钥。(2) 把字符串的小写字符转换为大写字符。另见常量折叠 (constant folding)。(3) 将在一行放不下的行的一部分放到下一行。对照截断 (truncate)。

叠印 (overstriking)： 一种生成特殊字符的方法，它输入一个字符而不把打印头移到下一个字符的位置，然后在第一个字符的顶上输入第二个字符。

顶层窗口 (top-level window)： AIX windows 和“增强的 X-Windows”中，包含所有与客户机应用程序相关的窗口的主窗口。

顶层事务 (top-level transaction)： 不在另一个事务作用域内执行的事务。换句话说，顶层事务是事务系列的根，即使它是族树中的唯一事务。

顶层小窗口 (top-level widget)： AIXwindows 和“增强的 X-Windows”中处于或邻近对象类层次结构的顶层的小窗口类，一般称作“核心类”。

顶层阴影 (top shadow)： 在 AIXwindows 中，一个穿过矩形图形对象（小窗口或小配件）顶部的狭窄浅色带，此对象在操作对象时创建三维外观。

顶层 (top-level)： AIXwindows 和“增强的 X-Windows”中指小窗口 / 小配件层次结构的最高级别的小窗口类。

定点设备 (pointing device)： 在“增强的 X-Windows”中，一种在有效尺度内移动的设备，它通常指一个鼠标。由 Core 协议定义一个可见光标；可见光标像指针一样，指示设备连接哪儿，它就跟踪到哪儿。

定点 (pointing)： 在显示对象上定位指点光标。排列鼠标指针的操作，以便指针在某些对象的顶部。

定界符 (delimiter)： (1) 一个字符或字符序列，标记字符串或数据单元的开始或结束。(2) 在一行输入内容中对单词或值进行分组或分隔的字符。

订口余白 (back margin)： 最接近装订边的页边距。通常指正面页面的左边距，反面页面的右边距。

定位器采样速率 (locator sample rate)： 来自定位设备的输入速率。与采样速率 (sample rate) 同义。

定位器分辨率 (locator resolution)： 一个定位设备上的点的密度。

定位器 (locator)： 在计算机图形学中，一个提供坐标数据的输入设备；例如，一个鼠标，图形输入卡，或者翻阅轮。

定位指针 (seek pointer)： 在字符文件或设备中包含当前位置偏移量的数据结构。

定向列表数据 (list-directed data)： 在 FORTRAN 中，依照 I/O 列表中的变量类型和长度，在主存储器和一个 I/O 设备之间传输的数据。另见格式化数据 (formatted data)。

定向列表 (list-directed)： 在 FORTRAN 中，一个使用一个数据列表来代替一个 FORMAT 规范的输入 / 输出规范。

定制概要文件 (customization profile)： 一个包含设备或软件程序的可选更改到缺省设置的描述的文件。另见概要文件 (profile)。

定制设备对象类 (Customized Devices Object Class)： 在每个设备实例（由唯一逻辑名区分）的 ODM 中的表示。定制设备对象类包含设备的基本信息，如设备状态及如何访问包含关于在其它对象类中信息。

定制数据库 (Customized Database)： ODM 中的实体，包含系统里已定义或可用设备的配置数据。另见设备配置数据库 (Device Configuration Database) 和预定义数据库 (Predefined Database)。

定制 (customization)： (1) 在 NIM 环境中，这是可选的软件安装。(2) 描述可选更改到软件程序（已安装到系统并配置为可用）缺省值的过程。对照配置 (configuration)。另见系统定制 (system customization)。

定制 (customize)： (1) 为系统描述用于特定数据处理系统或网络的设备、程序、用户和用户缺省值。(2) 在已安装并且已配置的软件程序中描述可选首选项或更改到缺省值。对照配置 (configure)。

动态变量 (dynamic variable)： 一个仅在需要时才分配的变量。需要显式的分配和取消分配。在 Pascal 里，预定义过程 NEW 和 DISPOSE 就是为这个目的提供的。

动态窗口 (dynamic window)： 一个可以在某个窗口或值域里动态改变的变量。

动态块头 (dynamic block header)： 一种数据结构，由编译器使用，在同一个堆里链接动态变量。

动态连接 (dynamic connection)： 当需要时才建立的连接，而不是在预先决定的或固定的。

动态链接 (dynamic linking)： 链接一个程序，其中库过程没有并入装入模块，而是在每一次程序装入时，动态地从它的库中装入。

动态路由 (dynamic routing)： 在主机，网络或两者之间设置路径的方法。通过使用守护程序在需要时更新路由表。

动态随机访问存储器 (dynamic random access memory)： 在存储器里，电池需要控制信号的可重复应用程序来保持存储的数据。这种控制信号的可重复应用程序叫做刷新操作。

动态许可证 (dynamic license)： 一个指定多个节点锁定的许可证。该许可证安装在许可证服务器上，然后许可证服务器设备从动态许可证获取许可证密码（每个密码指定了一个单独的节点锁定许可证）。服务器在用户节点自动的安装个人许可证，以调用许可的产品。

动态字符串 (dynamic string)： 请参阅字符串 (string)。

动态 (dynamic)： 创建弹出式菜单的一种样式。

抖动 (dithering)： 计算机图形中的一种交错处理明暗像素的技术，这样可以使得出的图像从一定距离看起来形状更平滑。

逗号表达式 (comma expression)：一中包含两个以逗号分开的操作数的表达式。虽然编译器对两个操作数都求值，但右操作数值才是表达式值。如果左操作数产生值，则编译器丢弃该值。

读队列 (read queue)：模块或包含移动上行流的消息的驱动程序中的一个消息队列。关联驱动程序的输入。

读访问 (read access)：授权查看 CDS 数据能力的访问权。

堵塞 (在“增强的 X-Windows”中) (occlude, In Enhanced X-Windows)：(1) 窗口存在的状态。如果一个窗口的视图完全被另一个遮住那么这个窗口是堵塞的。如果二者是映射的那么窗口 A 堵塞窗口 B，A 在全局堆栈中的次序是较高的，如果没有 B 的部分边界，那么就是可见的。在“堵塞”和“模糊”之间存在很细微的区别。窗口边界包含在计算中。InputOnly 窗口从不模糊其它 Windows 但能堵塞其它 Windows。(2) 当一个窗口完全的遮住另一个的视图时，它对另一个执行操作。对照小窗口可见性 (widget visibility)。

读/写引用传递 (pass-by-read/write-reference)：与变量传递 (pass-by-VAR) 同义。

端到端传输延迟 (end-to-end transit delay)：X.25 通信中，一可选 CCITT 特定设施。

端口 (port)：(1) 系统单元或远程控制器的一部件，它连接了用于外部设备（显示站、设备或打印机）的电缆。端口是一个访问点，用于数据进入（输入）计算机系统或从计算机系统退出（数据输出）。(2) 网络的入口或出口。(3) 为允许运行在一种计算机上的程序运行在另一种计算机上，对编程进行必需的更改。(4) 在 NCS 中，主机中的特定通信端点。端口用端口号标识。另见套接字 (socket) 和监听 (listening)。

短循环 (short circuiting)：带有 AND 和 OR 布尔表达式的求值，在这个表达式中，如果运算的结果可以由左边的操作数确定，则右边的操作数不求值。表达式的求值通常是从左到右。

短暂应用程序 (ephemeral application)：指不包含任何可恢复数据的应用程序。

短状态 (short status)：来自假脱机子系统的缩短形式（短格式）中的状态输出。

段标志 (segment flag)：fp_open 内核服务的 segflag 参数，指明了路径参数位于用户空间还是内核空间。

段单元 (segment unit)：在 Pascal 中，指一条包含与程序单元相链接的例程的独立的可编译的代码片断。另见程序单元 (program unit)。

断点 (breakpoint)：计算机程序中的某处（通常由指令指定），外部干预或监视器程序可能在此处中断执行。

段寄存器 (segment registers)：系统中拥有在当前使用中的内存段的实际地址的寄存器。

断言 (assertion)：请参阅程序断言 (program assertion)。

段 (paragraph)：(1) 通过空白行与其它文本隔开的文本。(2) 在字处理中，一个或多个句子可能由适当的指示符前导或跟随。

段 (segment)：(1) 分配给某个作业或系统任务的虚拟内存的邻近的区域。程序段可以自己运行，即使整个程序不在主存储中。(2) 虚拟内存分为多个段（每个段是由一个或多个 4KB 字节的页面组成的线性可寻址空间），这些段的最大大小可达 2 的 28 次方个字节。(3) 可以通过单个的、唯一的段寄存器 (256MB) 寻址的信息。(4) 计算机程序的一部分，可以作为实体运行而无需整个程序维持在系统内存中。(5) 一组显示元素。(6) “增强的 X-Windows”中绘制的一条或多条线，但不需要在端点处连接在一起。

堆叠磁带 (stacked tape)：具有多个软件映象的可引导磁带。

堆叠顺序 (stacking order)：互相堆叠于其上的兄弟窗口的关系。

堆栈缓冲区 (stack buffer)：按顺序存储可检索数据的存储区域。最后存储的文本是首先除去的文本。

堆栈回溯 (stack traceback)：指出进程到达其当前位置而选择的路径的调用顺序。

堆栈溢出 (stack overflow)：DOS 中的一种错误情况，由于缺少足够的 DOS 用来处理硬件中断的堆栈帧数而引起的错误。

堆栈指针 (stack pointer)：提供当前堆栈位置的寄存器。

堆栈 (stack)：(1) 存储器中的一个区域，它存储临时的寄存器信息并返回子例程的地址。(2) 一个构造后并维护的列表，这样存储的最后一个数据元素就是检索的第一个数据元素。(3) 内核方式中，和用户进程一起调页的区域。内核为每个进程保留一个堆栈。它保存进程信息，比如调用链和内核用于用户进程的本地变量。

堆 (heap)：动态分配变量的集合。

对称多处理器系统 (symmetrical multiprocessor (SMP) system)：包含多个本质上相同且执行相同功能的处理器的系统。

对等通信 (peer-to-peer communications)：指在交换中具有同等状态的两个节点间的数据通信。两个节点的任一节点都可开始这次对话。另见逻辑单元类型 6.2 (*Logical Unit Type 6.2*)。

对话方式 (conversation mode)：计算机系统的一种操作方式，其中用户和系统间的备用项和响应序列以与两人之间对话相似的方式进行。

对话框 (dialog box)：在需要用户进一步输入信息或系统需要显示信息时显示的窗口。结构上闭合的行，如包括特定的双向文本接口的小窗口或 shell。

对话密钥 (conversation key)：一个短期加密密钥，由认证服务提供给两个主体以确保它们之间的通信安全。另见会话密钥 (*session key*)。

对话相关器 (conversation correlator)：一个 LU 服务使用的内部 SNA 标识，用于跟踪哪个应用程序正在使用哪个对话。一个长为 1 到 8 字符的标识符，由连接函数分配，LU 服务维护。

对话小窗口 (dialog widget)：通过 Create 菜单的 Dialog 子菜单选择的任何小窗口类；对话小窗口的对话 shell 总与其隐式 shell 一样。另见对话 shell (*dialog shell*)。

对话 shell (dialog shell)：瞬时 shell 的一个子类，这种 shell 用于对话框。另见对话小窗口 (*dialog widget*)。

对话 (conversation)：(1) 在 SNA 中，事务程序对间的逻辑连接，用于在类型为 6.2 的逻辑单元间（从事务到事务）连续地共享会话。当对话处于活动状态时，它独占使用以明显 (distinct) 括弧定界的 LU-LU 对话；后续会话可以使用相同的对话。(2) 两个应用程序间的信息互换。(3) 一个允许两个应用程序间交换信息的路径。(4) 依靠键盘在计算机和用户之间的交互。

对话 (dialog)：在交互式系统中与两人之间的对话类似的一系列相关查询和响应。

队列节 (queue stanza)：定义作业可排队到的一个或多个设备的队列。

队列设备 (queue device)：一种逻辑设备，它定义连接到队列的物理设备的特征。

队列元素 (queue element)：数据块或队列中的项。

队列 (queue)：(1) 等待处理的项组成的行或列表。(2) 形成队列或排入队列。

对齐 (alignment)：在信息单元的完整边界上，一固定长度字段（如半字或双字）在主存储器中的位置。例如，一个字边界是可以被四除尽的存储器地址。

对齐 (justify)：打印具有偶数的右边和左边页边距的文档。

对数 (logarithm)：一种有关编号系统基数的数学运算。

对象标识 (object identifier)：值（区别于所有其它的这样的值）是与信息对象相关联的。（X.208）

对象程序 (object program)：完全编译的或者准备装入计算机的汇编程序。与目标程序 (*target program*) 同义。

对象定义 (object definition)：请参阅类 (*class*)。

对象管理 (object management)：创建、检查、修改和潜在复杂信息对象的删除。

对象句柄 (object handle)：在图形文件中，弧的开始点或者行对象。

对象空间 (object space)：定义的图形对象的空间。选择便利点作为原点，被定义的对象相对于这点。当对象通过调用 callobj 子例程而呈现时，它被呈现在模拟坐标中，对象空间成为（在那个时刻）与模拟空间一样。

对象库 (object library)：直接访问存储设备上的区域，一般用来存储对象程序和例程。

对象类表 (Object Class Table, OCT)：再现允许对象类描述的目录模式的属性。

对象类型 (object type)：在 CDE 中，桌面机制一般用相应的应用程序和操作关联特殊的数据文件。对象类型为输入文件（比如名称或内容）定义标准，外观（用于文件管理器的图标）和对象的行为（例如，当双击它时发生什么）。

对象类 (object class)：在 ODM 中，对象的存储集合有相同的定义，概念类似于结构数组。另见对象 (*object*)、结束描述符 (*terminal descriptor*) 和类 (*class*)。

对象模块 (object module)：(1) 输入链接编辑程序的相应目标程序的一部分。(2) 在由编译器从源程序产生的机器语言指令集。另见模块 (*module*)。

对象实例 (object instance)：请参阅实例 (*instance*) 和对象 (*object*)。

对象数据管理器 (Object Data Manager, ODM)：数据管理器用于系统数据的存储。许多系统管理函数使用 ODM。在许多命令中使用的信息，作为相关特征的对象而保存和维护在 ODM 中的 SMIT 函数。

对象 UUID (object UUID) : 标识特殊的对象的 UUID。RPC 运行时库和在 NCS 中定位代理程序都使用对象 UUID 来标识对象。另见通用的唯一标识符 (*Universal Unique Identifier, UUID*)。

对象 (object) : (1) 在网络安装管理环境 (NIM) 中, NIM 数据库中表示机器、网络或资源的项。(2) 在“增强的 X-Windows”中, 由私有数据和操作私有数据的公共例程构成的软件抽象。该抽象的用户可以仅通过对象的公共例程的调用和该对象相互作用。(3) 在 ODM 里, 对象类的实例或成员, 概念上类似于结构是结构的成员或数组。另见 *对象类 (object class)*。(4) 在 Pascal 里, 和数据对象 (*data object*) 是同义词。(5) 在 GL 中, 和显示列表 (*display list*) 是同义词。一序列的绘图命令被编译成一个单元。在概念上, 显示列表和宏相似: 通过提到它的名称可被多次调用。通过适当的使用转换矩阵, 可以在不同的位置、以不同的大小和方向对其进行实例化。例如, 按螺母形状排列的多边形序列可以被编译成为一个对象。于是通过调用该螺栓的显示列表, 就可多次绘制它。(6) 在 NCS 里, 正确定义的操作来操作一个实体。磁盘文件、打印机和阵列处理器都是对象的示例。通过接口来访问对象。每一个对象都有类型。(7) 在 Ada 语言中, 对象包含一个值。程序通过详细描述对象声明或评估分配算符来创建对象。声明或分配算符为对象指定一个类型: 对象只能包含该类型的值。另见 *常量 (constant)* 和 *变量 (variable)*。(8) 在 C++ 里, 一个是存储区域。在 C++ 中当定义了变量或调用了 **new**, 便创建了对象。当超出作用域时, 对象便破坏。(9) 在 XDS 里, 任何对象都在某一“域”中, 通常远程通信和信息处理或其一部分的域是可确认的 (可命名的), 对它来说 DIB 包含某些信息。(10) 在 XOM 里, 通过接口创建、修改、或破坏任何复杂信息对象。(11) 在 CDE 中, 任何数据的逻辑片具有关联的行为。例如, 文件管理器、文件、文件夹和应用程序都被认为是对象。每种类型的对象具有特定的关联操作。典型地, 每一个对象用一个图标表示。另见 *UUID 类型 (type UUID)*。

多重初始地址服务器 (multihomed server) : 有一个以上网络接口在运行的服务器。

多重初始地址机器 (multihomed machine) : 有一个以上配置的网络适配器和一个以上主机名的机器。

多重继承 (multiple inheritance) : C++ 中通过派生实现的一种面向对象的编程技术。

多重指示器 (polymarker) : 在计算机图形学中, 一系列标记。标记的定义包含像颜色、样式、宽度、高度、模式和原点这样的特定属性。

多点广播地址 (multicast address) : 对光纤分布式数据接口 (FDDIA) 来说, 多点广播地址是地址集合的高位字节中带有高位 (high order bit) 的地址。该类型的地址允许用户在环集合中对同一地址有多个站。如果一个包被发送

到一个多点广播地址, 所有具有该地址的站都可接收到该包。有时候一个多点广播地址用作分组地址。

多点链路 (multipoint link) : 互联几个站的电路。

多点 (multidrop) : (1) 在一个位置上连接到一个多点通道的所有站。(2) 在中心节点和端点节点之间的路径上有一个或多个中间节点的网络配置。

多点 (multipoint) : 属于在单电信线路上的两个以上的站之间的通信。

多路复用器 (multiplexer) : 请参阅 *多路复用设备 (multiplexed device)*。

多路复用设备 (multiplexed device) : (1) 接受几路输入信号并将它们复合为单个输出信号又能使每个输入信号可以恢复的设备。(2) 能够交错两个或多个活动事件或能够分配交错序列事件到各自活动的设备。

多路复用 (multiplex) : 在单个通道上间隔或同时传输两个或更多个报文。

多模光纤 (multimode optical fiber) : 允许一个以上绑定模式传播的渐变率 (graded-index) 光纤或级射率 (step-index) 光纤。对照单模光纤 (*single-mode optical fiber*)。

多任务方法 (multitasking) : 为并发性能或间隔处理两个或两个以上任务所提供的操作模式。任务分配法 (tasking) 的同义词。

多数数据位 (more-data bit) : 请参阅 *M-bit*。

多态性函数 (polymorphic functions) : 可应用于多个数据类型的对象的函数。C++ 以两种方法实现多形函数: 重载函数 (编译时解析调用); 虚拟函数 (运行时解析调用)。

多态性 (polymorphism) : 在不同系统中可能有不同意义的面向对象编程功能。在多态性的各种定义下, (a) 可使用各种类型自变量执行的方法或过程调用, 或 (b) 同一变量可在不同时间假定不同类型的值, 或 (c) 一个方法名可指示多个方法调用。

多线程应用程序 (threaded application) : 一个通过同时使用单个地址空间内的多重执行路径 (控件线程) 来执行其功能的应用程序。

多线路 (multiline) : 一个以上的通信线路。

多用户模式 (multiuser mode) : 使两个或更多用户在给定的时间段内使用一个处理器服务的操作模式。除非另外指定, 使用通常是连续的。

多字节控制 (multibyte control)： 在一个字符流数据中，两类有效控制之一。转义序列 (*escape sequence*) 的同义词。

[E]

二叉搜索树 (binary search tree)： 一种搜索结构，其中每一步搜索都用二去除数据元素集；当数据元素为奇数时采取适当的操作。

二级中断处理程序 (second-level interrupt handler, SLIH)： 取决于设备的例程，它处理来自特定适配器的中断的进程。SLIH 由与该中断级别相关的一级中断处理程序 (FLIH) 调用。

二进制表达式 (binary expression)： 包含两个操作数和一个运算符的表达式。

二进制补码 (twos complement)： 纯二进制计数系统中的基补码。通过取一个二进制反码，然后在结果数字上加 1，派生出二进制补码。在二进制补码系统中，数字 n 的二进制补码是 $-n$ 。另见二进制反码 (*ones complement*)。

二进制常量 (binary constant)： 由一个或多个二进制数字组成的常量。

二进制反码 (ones complement)： 在纯二进制记数系统中减小的基补码。在二进制数中通过用一全部替换零和用零全部替换一派生二进制反码。在二进制反码系统中，数字 n 的二进制反码是 $-n$ 。另见二进制补码 (*twos complement*)。

二进制数字 (binary digit)： 与位 (*bit*) 同义。

二进制同步通信 (BSC) (binary synchronous communication, BSC)： 一种电信线路控制形式，在站与站之间，二进制编码数据的二进制同步传输使用标准传输控制字符集和控制字符序列。对照同步数据链路控制 (*synchronous data link control*)。

二进制文件 (binary file)： 一个包含非 ASCII 字符集代码的文件。二进制文件可以利用文件中每个字节的所有 256 个可能值。

二进制映像传送 (binary-image transfer)： 请参阅位块传送 (*bit block transfer*)。

二进制运算符 (binary operator)： (1) 一个表示对两个数据项、数组或表达式执行操作的符号。二进制运算符的四种类型为：数字、字符、逻辑和关系。对照一元运算符 (*unary operator*)。 (2) 具有两个术语的算术运算符。与二元运算符 (*dyadic operator*) 同义。

二进制 (binary)： (1) 属于基数为二的数字系统。二进制数字为 0 和 1。 (2) 涉及两种状态的选项，例如开关或是

否。 (3) ODM 中用于将变量定义为不以 null 结束的位字符串的结束描述符。另见结束描述符 (*terminal descriptor*)。

二维半 (two and one-half dimension)： 请参阅 2-1/2 维 (*2-1/2 dimension*)。

二义性派生 (ambiguous derivation)： 如果一个 C++ 类派生自两个或更多个具有相同名称的基类，那么该派生是二义性的。

[F]

发光二极管 (light-emitting diodes, LED)： 激活时可以发出可见光或红外光的半导体芯片。LED 经常用于显示操作员信息。

发光可编程功能键盘 (lighted programmable function keyboard, LPFK)： 一个主要用于图形应用程序的输入设备，它可以在应用程序的控制下点亮各键。

发行版更新软件包 (release update package)： 一组可选择的增强功能、文件集和自上次操作系统发布以来的可用可选软件产品的新版本。还包含每个软件包的维护包软件包。

发行版号 (release number)： 程序的一种发行版级别，它是对程序的外部编程接口更改的一个指示符。版本、发布、修订和修正级别一起组成程序的程序级别或版本。另见修正号 (*fix number*)、修订版号 (*modification number*)、版本号 (*version number*)、程序级 (*program level*) 和版本 (*version*)。

发送调步 (send pacing)： 在 SNA 中，指组件发送的消息单元的调步。对照接收调步 (*receive pacing*)。

返回码 (return code)： (1) 返回给程序的一个值，它表明该程序所发出的操作的结果。 (2) 用来影响后继指令运行的代码。与返回值 (*return value*) 同义。

返回语句 (return statement)： 一种编程语言控制语句，它包含字 return 后跟一个可选表达式和一个分号。

返回值 (return value)： 同返回码 (*return code*)。

返回字段 (field return)： 在逆向从字段到字段移动数据光标的操作，由面板布局确定。

反馈 (flyback)： 类似于逆向换行的移动。

反向计费 (reverse charging)： 在 X.25 通信中的一种可选设施，它允许 DTE 请求对通信会话的计费由被呼叫的 DTE 方偿付。另见可选的设施 (*optional facilities*)。

反转视频 (reverse video) : 将字符、字段或光标颜色同它们的背景颜色反相, 以加强字符、字段或光标亮度的一种显示形式。例如, 将黑底红字反相为红底黑字。

范围约束 (range constraint) : 在 Ada 语言中, 一个类型的范围约束指定一个范围, 从而确定属于该范围的类型的值的子集。

范围 (extent) : 在磁盘或者软盘上的连续空间, 它由特定的数据集、数据空间或文件占据或保留。

范围 (range) : 在 Ada 语言中的一组相连的标量类型值。该范围通过给值较低和较高界限来指定。该范围内的值被称为属于该范围。

方法描述符 (method descriptor) : 在 ODM 中, 输入的一个命名的变量 **method** 用来定义一种方法或者一个对象的操作。该方法可以是一个命令, 程序, 或者 shell 脚本。参阅方法 (*method*) 和描述符 (*descriptor*)。

方法 (method) : (1) 在“增强的 X-Windows”, 小窗口自己执行的函数或者过程。(2) 在 ODM 中, 和一个对象有关的并为该对象定义为一种方法描述符的值的可执行代码。该方法可以是一个命令, 程序, 或者 shell 脚本。另见方法描述符 (*method descriptor*)。(3) 一个面向对象编程术语同义与成员函数 (*member function*)。

方式名称 (mode name) : (1) 登录方式表中的项名称。(2) 在 SNA 中, 标识用于会话的规则和协议集。

方式字 (mode word) : 描述 i-node 类型和状态的一个 i-node 字段。

方式 (mode) : (1) 一种操作方法。(2) 在 SNA 数据通信中, 用于会话的规则和协议的集合。(3) 在 M-Video 抓取适配器 (M-Video Capture Adapter) 中, 类似有效或覆盖层抓取的操作方法。(4) 对于 Ada 编程, 请参阅参数 (*parameter*)。

方位角 (azimuthal angle) : 在 GL 中, 如果一图元放置在背景中, 其 *z* 坐标垂直向上, 方位角视图角就是观察者用 *y* 轴在 *x-y* 平面里所做的角。如果观察者以图元为中心围绕图元行走, 方位角是唯一变化的东西。

方向键 (arrow keys) : CDE 中, 键盘上的四个方向键。

方向性组件 (directional component) : AIXwindows 中用给定值指定方向的复合字符串的一部分。方向性组件由 **XmStringDirectionCreate** 函数创建。

防火墙 (firewall) : 一个系统或者机器, 它控制在外部网络和专用网间的访问。

访问过程 (access procedure) : 用于获取对共享资源访问的过程或协议。在局域网中, 共享资源是传输介质。IEEE 802 标准指定的介质访问过程是 CSMA/CD 令牌总线和令牌环。

访问级别 (access level) : 在计算机安全性中, 用户访问安全文件或库时所拥有的权限级别。

访问解析 (access resolution) : 确定某个特定类成员的可访问性的过程。

访问控制表 (ACL) (access control list, ACL) : (1) 一种文件属性, 包含文件访问控制的基本和扩展权限。(2) 可以访问客户机程序的主机列表 (由“增强的 X-Windows”维护)。缺省情况下, 只有本地主机上的和该列表 (即大家所知的“访问列表”) 中的程序可以使用屏幕。该列表可由本地主机的客户端程序更改; 一些服务器实现也可以修改该列表。授权协议名称和服务器在连接设置时接收到的数据也可能影响该列表。另见任意访问控制 (*discretionary access control*)。

访问控制列表项 (access control list entry) : 访问控制列表中的数据用以指定一个权限集。在 **principal** 或 **group** 项的情况下, 权限集是授权给拥有在项中指定的特权属性的 **principal**; 在 **mask** 项的情况下, 权限集是对 **principal** 或 **group** 项中的权限集执行掩码操作。

访问权 (access right) : 请参阅许可 (*permission*)。

访问声明 (access declaration) : 用来恢复对基类成员的访问。

访问许可 (access permission) : 一组指定, 用来确定谁可以访问某个特别文件及用户如何能够访问该文件。另见许可代码 (*permission code*)。

仿真器 (emulator) : 一个程序, 使一台计算机作为连接到另一个系统的工作站。

仿真 (emulation) : (1) 使用编程技术和特殊机器功能部件来允许一个计算系统运行为其它系统写的程序。(2) 仿真。比如, 当一个计算机模拟另一个计算机的特征。

放入信息 (drop-in) : 一些应用程序能够检测, 某个图标已经在它们的窗口中放下, 来执行适当的操作, 比如打开一个文档。

放下目标 (drop target) : 在 CED 中, 一个矩形的图形代表了在应用程序里按下的区域。

放下区域 (drop zone) : 在 CED 中, 工作区的一个区域, 包括“垃圾桶”, “打印管理器”和“邮件前台控制面版控制”, 它接受一个放下的图标。为了快速访问, 图标也可以在工作区里放下。

放下页码 (dropped folio)：一种页记数样式，其中页码数打印在页脚。请参阅页 (folio)、盲页码 (blind folio) 和快速页码 (expressed folio)。

放下 (drop)：在 CDE 里，在开始拖动一个对象以后释放鼠标按钮的操作。如果在一个适当的位置放下该对象，操作就会开始。

非打印字符 (non-printing character)：控制字符 (control character) 的同义词。

非法 (illegal)：要求报告某种实现的体系结构规则的违例。另见不可预知的 (unpredictable)。

非归零 (NRZ)：一种二进制代码系统，该系统中，在全部时间间隔中必须维持信号状态而且不回复到信号元素间的备用或静止状态。根据 Nyquist 的理论，使用 NRZ 允许通道里的最大数据信号率应该是带宽的两倍。

非活动 (inactive)：描述无输入焦点的窗口。

非交换网络 (nonswitched network)：网络中，不必拨号建立的计算机或设备之间的连接。

非交换线路 (nonswitched line)：(1) 不必通过拨号建立的计算机或设备连接。(2) 一种专线。租用设施 (leased facility) 的同义词。对照交换线路 (switched line)。

非可变随机存取存储器 (NVRAM)：在机器的电源关闭后能保留其内容的随机存取 (存储) 存储器。NVRAM 的特定部分留出供系统 ROS 的启动设备列表使用。

非可执行程序单元 (nonexecutable program unit)：FORTRAN 中的块数据子程序。

非扩展结果 (non-extended result)：在 dlc_getx_arg 结构的 result_ext 文件里没有定义任何数据的异常通知。

非零 (nonzero)：不等于零的值。

非模态对话框 (modeless dialog)：在 AIXwindows 中的一个对话框 (Dialog) 小窗口，它请求来自用户的输入但是不中断工作会话。

非模态弹出 (modeless pop-up)：在增强的 X-Windows 中的一个窗口，它正常时是可见的并由窗口管理器控制。

非确定性程序 (nondeterministic program)：通过分析源代码不能确定结果的程序。在程序第一次使用之前没有初始化变量，它就是非确定性程序，因为在程序运行中未初始化的变量的值可以改变。

非生产性的 (non-productive)：仅由重复控制信息组成而不包含最终用户数据的介质上的数据流量。

非输入区段 (no-input zone)：无用区段 (dead zone) 的同义词。

非锁定锁定 (nolock lock)：当必须提供锁定模式而实际上并不要求锁定数据时所使用的哑元模式。指定这种锁定模式的函数可以读取数据，即使它这时也被其它的操作或事务锁定，使得脏读得以进行。这种锁定在操作上是一致的，这意味着使用它锁定和读取一个值时将返回一个立即及时更正的真实值，但在处理上是不一致的。

非同步管道消息 (broken pipe message)：管道变成非同步时出现的消息。

非一个数字 (NaN)：在二进制浮点计算中的一个值，它包含掩码状态和二进制数字序列并没有被解释为机器值。

非引用的指针 (dereferenced pointer)：在 Pascal 中使用 -> 或 @ 运算符的表达式，用来定位来自指针的动态变量。

非执行语句 (nonexecutable statement)：描述程序单元、数据、编辑信息或语句功能的特征的语句，但并不产生任何操作。

非终端符号 (nonterminal symbol)：解析器识别的结构。

分贝 (decibel, dB)：(1) 一贝尔的十分之一。(2) 信号强度单位，如数据通信通道中的信号。(3) 用于测试相对功率的单位。分贝数量等于测得功率水平的比率的对数 (底为 10) 的 10 倍。

分辨率 (resolution)：一种进程或能力，使对象的各个部分在屏幕上可区分，特别与字体生成有关。在屏幕上可显示的点数。用于确定图形图像的清晰度。

分布式功能终端 (Distributed Function Terminal, DFT)：执行原来由处理单元完成的操作的工作站，诸如管理数据链路、控制设备和格式化数据。

分布式计算环境 (Distributed Computing Environment, DCE)：资源和数据可位于不同的处理器上的计算环境。

分布式介质 (distribution medium)：操作系统软件、许可程序或应用程序分发给用户所用的介质。分布式介质可以是硬件所支持的几种不同介质其中任何一种，如流式盒式磁带、9 磁道磁带或 3.5 英寸软盘。

分布式可用空间 (distributed free space)：可用空间 (free space) 的同义词。

分布式事务 (distributed transaction)：可以更新处于很多台机器上的很多用户进程中的数据的事务。

分布式文件系统 (distributed file system)：在通信网络种由物理上驻留在多台计算机上的文件或目录组成的文件系统。

分布式 (distributed) : 组成计算环境的程序和计算机化的信息源, 在物理上可以位于不同计算机系统上, 但仍作为单个逻辑单元协同工作。事务处理系统很容易就可以移到一个分布式计算环境中, 传统上, 这些系统涉及通过从远程终端提交修改或信息请求来修改处于中央位置的信息源。

分层网络 (hierarchical network) : 其中主机由树形结构的管理机构来管理的网络。该网络结构减轻了中央管理机构的管理负担。

分层文件系统 (hierarchical file system) : 在 CDE 中, 使用容器层次结构在计算机系统上组织数据的方式, 通常称为文件夹 (目录) 和文件。在这种模式下, 文件夹可能包含其它文件夹和文件。文件夹内的逐个文件夹容器创建了组织级别, 即为层次结构。

分段故障 (segmentation fault) : 内存寻址异常。当引用未被分配至引用进程的内存位置时, 发生此异常。出界数组引用或指针的不正确使用会引起分段故障。

分隔符 (separator) : 分隔命令或文件各部分或定界字符串的标点字符。

分块调用 (blocking call) : 一种调用程序一直暂挂, 直到被调用过程完成的调用。

分块因子 (blocking factor) : 内部循环的迭代次数, 每次相应分块循环的传递都会执行。

分块 (blocking) : 一涉及更改访问大数组循环的访问顺序的优化, 以便尽可能少地访问每个数组元素。

分类机制 (classification mechanism) : 工作负荷管理 (Workload Management) 类指定规则集, 决定指定给进程的类。

分类 (sort) : 重新安排一组项的一些或全部, 基于那些项的内容或特征来安排。

分派方法 (dispatch method) :

分派器 (dispatcher) : XOM 中使用工作区接口函数实现服务接口函数的软件。

分派 (dispatch) : 对即将运行的作业在特定的设备上的分配处理时间。

分配 (allocate) : (1) 分配资源 (如磁盘文件或软盘文件) 以执行特定任务。对照解除分配 (deallocate)。(2) 在本地 LU 和远程 LU 之间分配会话的请求。(3.) 在 NCS 中, 创建远程过程调用 (RPC) 句柄来识别对象。

分区 (partition) : (1) 在硬盘上的逻辑存储分区。(2) 一个固定大小的存储分区。

分页 (pagination) : (1) 调整文本到页边距内的过程。(2) 在字处理时, 根据预设的页面布局参数的数字来自动排列文本。

分支处理单元 (branch processing unit) : 一个处理单元, 处理分支指令和分派、整型 / 浮点指令到整型 / 浮点但愿。

分支 (branch) : 计算机程序中, 选择许多可供选择的指令集之一的指令。仅满足指定条件时出现的条件分支。选择许多可供选择的指令集之一的指令。

分组 (tab group) : AIXwindows 中将 **XmPrimitive** 小窗口组织成组以实现组中或组之间更有效遍历的方法。

峰值速率 (peak rate) : 设备可在理想条件下操作的最大速度, 如果其设计者选择工作负载。

封装 (encapsulation) : 隐藏数据对象的内部表示并从客户机程序实现功能的细节。这可以使最终用户将注意力集中在数据对象和功能的使用, 而无需了解它们的表示或实现。

否定响应 (negative response) : 数据通信中的一个应答, 表示数据未正确地接收到或命令不正确或不可接受。

浮点常量 (float constant) : 包含小数点、指数或两者的数。指数包含 “e” 或 “E”、可选的符号 (+ 或 -) 与一或多位数字 (从 0 到 9)。

浮点常量 (floating-point constant) : 一种浮点数量表示法, 把它表示为一个可选的符号后跟一位或多位数字包括小数点。另见浮点数 (floating-point number)。

浮点类型 (floating point type) : 请参阅实类型 (real type)。

浮点数 (floating-point number) : 由有区别的成对的数字表示的实数。实数是小数部分、一个数与通过提升隐式的浮点底数到由第二个数表示的乘幂获得的一个值的乘积。另见浮点常量 (floating-point constant)。

浮点异常 (floating-point exception) : 请参阅异常 (exception)。

浮点 (floating point) : 以 32 位或 64 位表示实数的一种方法 (也即, 有分数或小数的值)。浮点表示法对于描述非常大或非常小的数字有用。

浮动保持 (floating keep) : 通过比较文本块的大小与该页剩余空间的大小, 确保围住的文本穿越页边界时不被破坏的一种保持。如果这两者不符合, 以第一个可能的机会在下一页打印该块。固定的保持把该页的剩余保留为空白, 并且浮动保持移动后继文本到那页。

浮动显示 (floating display)：在文本格式编排中，**nroff** 命令保存在某一页的文本块。然而，如果没有空间用于在当前页的浮动显示，**nroff** 命令取消该显示，并完成用输入文件的跟随该显示的文本填充该页。当该页满后，**nroff** 命令把该显示放在下一页的顶端，然后继续处理前一页的文本。另见**静态显示 (static display)**。

浮动许可证 (floating license)：请参阅**并行使用许可证 (concurrent-use license)**。

符号表 (symbol table)：请参阅**分析 (parse)**；**语法分析**。

符号地址 (symbolic address)：唯一的行地址，比如 . (句号) 或 \$ (美元符号)，用于代替行编号地址来识别数据位置。另见**模式地址 (pattern address)**。

符号调试器 (symbolic debugger, sdb)：辅助以某个高级语言写的调试程序的工具。

符号链接 (symbolic link)：包含另一个文件或目录路径名称并作为指向另一个文件或目录的一种文件。

符号名 (symbolic name)：用于表示实体 (比如文件或数据项) 的唯一名称。另见**名称 (name)**。

服务方式 (service mode)：维护方式 (*maintenance mode*) 的同义词。

服务访问点 (service access point, SAP)：在以太网逻辑链路概要文件中，指在本地系统上的事务处理程序的地址。此地址为十六进制值。

服务更新 (service update)：更正“基本操作系统” (BOS) 中的缺陷，或向 BOS 或可选软件产品添加新功能的软件。另见**维护包更新 (maintenance level update)**。

服务控件 (service controls)：适用于所有目录操作的一组参数，它指挥或限制服务的供应。

服务器报告 (server reporting)：服务器的一个协议，用于因要注册而向先前未知客户机联系的单元管理器报告。

服务器抓取 (server grabbing)：当客户机捕获服务器获得独有使用权以防止来自其它客户机连接的请求直到抓取结束。对于橡皮筋式 (rubber-banding)、弹出式菜单或非可视运行请求，这通常是瞬时状态。

服务器 (server)：(1) 通常在后台 (守护程序) 运行且由“系统程序控制器”控制的应用程序。(2) 在网络上，包含数据或提供网络上其它计算机访问的设施的计算机。(3) 处理协议、队列、路由以及在计算机系统的设备中传送数据所需的其它任务。(4) 在“增强的 X-Windows”中提供了基本窗口机制。它处理来自客户机的 IPC 连接，多路分解图形请求到屏幕上，并多路复用输入返回到客户机。(5) 在

NCS 中，指向一个或更多对象导出一个或更多接口的进程，可以从远程主机调用其过程。

服务请求号 (service request number, SRN)：服务技术人员用来确定系统出故障区域的一组编号。

服务事务处理程序 (service transaction program)：(1) 向 SNA Server 内部提供功能的程序。(2) 由事务处理系统执行的事务处理程序。服务事务处理程序执行各功能，如提供对远程数据库和远程队列的访问。另见**应用程序事务处理程序 (application transaction program)** 和 **事务处理程序 (transaction program)**。

服务质量协商 (quality of service negotiation)：一个可选的 CCITT 指定的设施。

父标识 (parent ID)：标识控制小图形对象 (称为孩子) 的图形对象的字符序列。

父窗口 (parent window)：在“增强的 X-Windows”中，控制其子窗口的大小和位置的窗口。如果窗口有子窗口，它就是父窗口。

辅键 (secondary key)：记录的一个键字段，它定义了辅助索引。

父类型 (parent type)：对于 Ada 编程，请参阅**派生类型 (derived type)**。

父目录 (parent directory)：当前目录的高一级目录。另见**父文件夹 (parent folder)**。

父设备 (parent device)：一个分级位置术语。它表示您连接的设备连接到的设备。例如，SCSI 磁盘的父设备可能是 SCSI 适配器。

父文件夹 (parent folder)：在 CDE 中，包含子文件夹和文件的文件夹。当讨论命令行活动时，这也可以被称为**父目录 (parent directory)**。另见**子文件夹 (subfolder)**。

辅站 (secondary station)：根据主站指示，运行数据链路控制功能的数据站。它解释接收到的命令并对传输生成响应。

辅助窗口 (secondary window)：持续时间短暂的窗口，比如对话框。此窗口只显示短暂的时间，通常在传递一些信息或让您获取一些操作性指示后就关闭。

辅助单元 (secondary unit)：在 Ada 语言中，库单元的主体 (比如子程序主体、包主体或类属主体) 或子单元。不是库单元的所有编译单元是辅助单元。辅助单元不受到其它独立编译单元的引用的支配，且能够看作是库单元的**隐含实现或单独声明**。

父 (parent)：(1) 使用派生原语生成子进程的进程。(2) 属于保护资源，文件或库的用户列表被一个或多个文件或

库共享。对照子 (*child*)。 (3) 在 AIXwindows 和 “增强的 X-Windows” 中, 控制连接的一个或多个小图形的对象的图形的对象。小图形对象称为孩子, 并且它们在删除其父亲时自动删除。 (4) 在 Ada 语言中, 数据包主体或子程序主体的相关的规范。子单元的父亲是它声明的主体。

副本链接 (copy-link): 一个建立在目标文件和相同文件的副本版本间的链接。任何作用于目标文件的副本版本的更改都会自动作用于原始目标文件。

覆盖层平面 (overlay planes): 使用显示器缓冲区中的一个或更多的位平面来创建可视数据 (文本, 图形等等), 以非破坏性方式中覆盖在帧缓冲区中的视觉数据。不是背景色, 覆盖层平面是透明的。

覆盖层 (overlay): (1) 写在 (因此破坏) 现有的文件上。 (2) 把程序段装入主存储器, 替换全部或部分以前装入的程序段。 (3) 在不同程序状态期间重复使用相同的内部存储区域。 (4) 在 DPS 中, 预定义的数据的集合比如行、色调、文本或在页面上打印时可以与变量数据合并的徽标。 (5) 在 “M 视频” 捕获适配器中, 一个源的指定像素和来自于另一个源的像素的替换, 比如覆盖在活动图像上的内存像素。

覆盖模式 (overwrite mode): 一种系统操作形式, 通过键盘输入的字符替换现有的字符。

覆盖 (override): (1) 替换以前的参数或值的参数或值。 (2) 替换参数或值。

覆盖 (overwrite): 记录到一个存储区域使以前该区域的数据被破坏。

复合对象 (compound object): 在 AIXwindows 中, 一个由在单一容器小窗口中收集的多个小窗口和小配件构成的图形对象。

复合字符串 (compound string): 一种为简化国外语言支持而设计的字符串类型, 通过允许无需对依赖于语言的属性 (字符集、文本和方向) 硬编码来显示文本。

副面板 (subpanel): CDE 中 “前面板” 的扩展, 它向上滑动以提供对附加元素的访问。副面板通常包含几组相关元素。

复数常量 (complex constant): FORTRAN 中, 用括号扩起来, 以逗号隔开的有序实型或整型常量对。有序对的第一个常量代表复数的实数部分, 第二个常量代表虚数部分。

复数类型 (complex type): FORTRAN 中表示复数值的数据类型。用括号扩起来, 以逗号隔开的有序实数数据项对方式表示的值。第一项代表复数的实数部分, 第二项代表虚数部分。

复数数学库 (Complex Mathematics Library): 一个 C++ 类库, 提供工具以生成复数及对其执行标准数学运算。

复数 (complex number): 一个由有序实数对构成的数字, 以 $a+bi$ 格式表示, 其中 a 和 b 为实数, i 的平方等于 -1 。复数由两部分组成: 实数部分和虚数部分, 其中 a 是实数部分的值, b 是虚数部分的值, i 是 -1 的平方根。

复位包 (reset packet): 用于使在数据终端设备 (DTE) 和数据电路终端设备之间接口处的虚拟电路复位的一种信息包。

复位冲突 (reset collision): 请参阅冲突 (*collision*)。

复位请求包 (reset-request packet): 在 X.25 通信中的一种信息包, 它用于在 DTE/DCE 接口处复位虚拟电路。另见包 (*packet*); 信息包; 分组。

复位确认包 (reset-confirmation packet): 在 X.25 通信中, 由数据终端设备 (DTE) 发送的一种包, 通知数据电路端接设备 (DCE) 一个复位操作已被处理。另见分组 (*packet*)。

复位原因 (reset cause): 请参阅原因代码 (*cause code*)。

复位诊断 (reset diagnostic): 请参阅诊断代码 (*diagnostic code*)。

复位 (reset): (1) 使一个计数器处于对应于某指定初始数的状态。 (2) 使某数据处理设备的全部或部分再回到一个规定的状态。 (3) 在一个虚拟电路中, 数据流控制的重新初始化。 (4) 将设备或电路返回到清除状态。 (5) 在 X.25 通信中, 重新初始化虚拟电路上的控制流, 复位时, 传送中的所有数据将为虚拟电路被消除。

复选框 (checkbox): 一个可以打开或关闭, 表示选项状态的小正方形框。

负载存储绑定 (load-store bound): 这里产生一系列计算的延迟, 因为大量数据必须装入到注册表中或者内存中。

负载级别 (load level): 在处理单元、通道或者设备工作之间的平衡。

负载 (load): (1) 从存储器传输程序或者数据到一个内存区域, 在该区程序可以运行, 数据也可以操纵。 (2) 将一张软盘放置到一个软盘驱动器中。 (3) 把纸插入到一个打印机中。另见调用 (*call*); 呼叫。

赋值表达式 (assignment expression): C 语言中, 把右操作数表达式的值赋给左操作数变量, 并使其变量值为右操作数值的表达式。

赋值兼容性 (assignment compatible)：表示值类型是否允许赋给某个变量。另见兼容类型 (*compatible types*)。

赋值语句 (assignment statement)：编程语言中，把表达式值赋给变量的语句。

赋值转换 (assignment conversion)：更改右操作数的形式，使右操作数与左操作数具有相同的数据类型。

复制构造函数 (copy constructor)：一种 C++ 构造函数，用于从另一个具有相同类类型的类对象制作副本。

复制 (copy)；**副本**：(1) 用户对已经存在的数据对象制作一个完整或部分副本的操作。(2) 一个通过双边协定存储在其它 DSA 中的项的副本，或是来自于请求 (高速缓存副本) 的本地或动态存储副本。

副作用 (side effect)：因通过某过程或功能改变非本地变量的值而引起的不期望的结果。

[G]

概要文件 (profile)：(1) 一种文件，它包含为系统或用户定制的设置。(2) 一种数据，它描述用户、程序或设备的重要特征。(3) 在安全性中，一种实体的特征描述，对其的访问是受控的。(4) 对特定网络操作员可用的控制描述。另见定制概要文件 (*customization profile*) 和映射 (*mapping*)。

高级程序到程序通信 (APPC) (Advanced Program-to-Program Communication, APPC)：一种通信体系，允许事务程序在对等的基础上交换信息。SNA LU 6.2 允许 APPC 体系运行在 SNA 网络。

高级数据链路控制 (high-level data link control, HDLC)：在数据通信中，使用指定的位串来控制数据链路，以与 HDLC: ISO 3309 Frame Structure 和 ISO 4335 Elements of Procedures 国际标准一致。

高宽比 (aspect ratio)：图元的高宽比。一个宽 10 英寸、高 5 英寸的四边形的高宽比为 10/5 或 2。

高速缓存查询 (cache lookup)：确定高速缓存是否包含满足存储引用的必要信息的过程。一被引用的定义好的地址位集，标识询问行。

高速缓存命中 (cache hit)：满足信息从高速缓存中可用的处理器存储引用。

高速缓存未命中 (cache miss)：(1) 当 CPU 引用数据高速缓存或指令高速缓存中不存在的数据或指令时发生的延迟。(2) 无法从高速缓存获得满足因而需要 RAM 访问的处理器存储引用。

高速缓存线标记 (cache line tag)：每一高速缓存线保存的信息，标识其包含的虚拟存储器部分。

高速缓存线 (cache line)：高速缓存查询期间，通常装入、存储及询问的高速缓存组件。另见组相关性 (*set associativity*)。

高速缓存一致 (cache coherency)：确保多个处理器上的多个线程更改单一的高速缓存线不会造成在不同高速缓存里的高速缓存线的不一致版本的需要。

高速缓存 (cache)：(1) 包含频繁访问到的数据或指令的高速缓冲存储区，可以比信息通常驻留的存储介质更快地递送数据或指令。高速缓存用于减少访问次数。另见回写式高速缓存 (*write back cache*)。(2) 在高速缓存放置、隐藏或存储。一种借此高速存储器为了其它目的不是马上需要的、用于保留曾经装入的数据的技术 (通常基于软件)，希望在高速存储器必须重新分配之前发生该数据的另一个请求。

高位 (high-order)：最左边的有效位。例如，寄存器中的位 0。

格式化程序 (formatter)：(1) 一个为源文档的打印作准备的计算机程序。(2) 文本处理器的组成部分，它格式化输入行用于打印或在特定类型的设备上显示。

格式 (format)：(1) 对于诸如字符、字段与行等的定义的布置，通常用于显示、打印输出或文件。(2) 确定如何记录数据的模式。(3) 布置诸如字符、字段与行等。(4) 在编程语言中的一种语言构造，它指定文件中数据对象以字符格式表示方法。另见已格式化的数据 (*formatted data*)。

隔行扫描技术 (interlacing)：一种双倍垂直分辨率的方法，是通过显示 1/2 扫描行的奇数视频帧实现。

隔离级别 (isolation level)：开放文件描述符 (OFD) 的说明字段，决定哪一个 OFD 运行的操作和其它 OFD 运行操作相分离的程度。操作分离是通过采用每个操作的内部锁来完成的。这个字段的值确定了如何获得和保持 OFD 所执行的操作的锁。

根窗口 (root window)：在图形环境中，每个屏幕有覆盖它的一个根窗口。这种窗口不能被重新配置或取消映射，但除此之外则像任何其它窗口一样执行。根窗口没有父窗口。

根段 (root segment)：(1) 在覆盖层操作中，运行其它覆盖层段时必须保留在主存储器中的程序部分；也就是覆盖程序中的第一段。当程序正在运行时，根段一直保留在主存储器中。(2) 在层次数据库中，树结构中的最高段。

根卷组 (root volume group, rootvg)：包含基本操作系统 (BOS) 的卷组。另见迁移安装 (*migration installation*)。

根目录 (root directory) : 含有系统中所有其它目录的一种目录 (/)。

根设备 (root device) : 根文件系统驻留其上的设备。

根文件系统 (root file system) : 基本文件系统, 所有其它文件系统可以挂装在该文件系统上。根文件系统含有使系统的其余部分运行所需的操作系统文件。

跟踪标识 (trace ID) : 被跟踪事件的唯一标识。

跟踪表 (trace table) : 包含计算机程序指令的性能的记录存储区域。

跟踪模板 (trace template) : 跟踪格式化程序使用此模板来确定如何格式化包含在跟踪条目中的数据。

跟踪日志 (trace log) : 记录跟踪事件于其中的文件。

跟踪守护程序 (trace daemon) : 从跟踪设备读取, 并写至跟踪日志文件。

跟踪条目 (trace entry) : 从跟踪事件记录的数据。

跟踪 (trace) : (1) 记录提供系统中发生的历史事件的数据。(2) 正运行的计算机程序的记录。它显示了运行的指令的顺序。(3) 监视系统性能或辅助调试程序。

根 (root) : (1) 具有最高权限的系统用户的用户名。(2) 在“增强的 X-Windows”中, (a.) 在其上创建窗口的屏幕。像素图或 GContext 的根与该像素图或 GContext 被创建时使用的画面的根相同。像素图或图形上下文的根与该像素图或图形上下文被创建时使用的画面的根相同。某个窗口的根就是在其下创建该窗口的根窗口。(b.) 在小窗口树上, Shell 小窗口由 XtInitialize 或 XtCreateApplicationShell 子例程返回。(3) 在 AIXwindows 中, Shell 小窗口是小窗口树层次结构的根, 该层次结构在任何给定应用程序界面中维护。另见小窗口树 (widget tree)。

更改位 (change bit) : 在每个页帧表项里的一位, 指示自上次操作系统清除页之后, 相应页已经被写。

更新传播 (update propagation) : 将更改应用于刚进行更改的 CDS 目录中的所有副本的快速尝试。更新传播比藏匿 (整个更改集合定期分发) 更有效更及时地传递更改。

更新脚本 (update script) : 由某个应用程序开发者创建用来更新程序的 shell 脚本或可执行文件。脚本文件必须遵照特定的准则以使其与操作系统提供的程序更新工具兼容。

更新时间戳记 (Update Timestamp, UTS) : 标识对特定 CDS 名称的任何属性进行最新修改的时间的属性。对于目录, UTS 反映了对整个目录 (不是其中的一个副本) 的属性进行的更改。

更新 (update) : (1) 修改存在于计算机大容量存储介质上程序或程序选项的过程, 它使程序可执行, 并确保已修改的程序可以与系统中其它受到影响的程序正确的交互。(2) 系统某部分的改进。(3) 添加、更改或删除项。(4) 根据指定的过程用当前信息修改主文件。

公布的事件 (posted event) : 一种通知, 通过 e_post 系统调用的方法, 由其连接的设备处理程序发送到 DLC。

公布 (post) : 使得弹出菜单或下拉菜单出现的一种操作。这种操作通常是单击或按压鼠标按钮上的一个按钮。

公告牌 (bulletin board) : 一模拟现实生活公告牌的图像对象, 从正在运行的客户机应用程序, 把文本和图像信息以消息的格式显示给用户。

公共块 (common block) : FORTRAN 中一个可以由调用程序和一个或多个子程序引用的存储区域。

公共目录 (public directory) : 在 BNU 中, 对所有 BNU 用户打开的目录 (/var/spool/uucppublic)。公用目录用于在系统之间传送文件和程序, 这些系统由 BNU 链接或由其它版本的 UNIX 至 UNIX 复制程序 (UUCP) 链接。

公共桌面环境: 一种可视化操作系统用户界面, 包含 AIXwindows 窗口里的图标和菜单。文件系统的图标化视图, 提供用户窗口、图标和菜单以执行程序和管理任务。

公共子表达式增强 (common subexpression enhancement) : 一种优化, 使编译器能够检测到一个表达式 (或循环) 中两个 (或多个) 截然不同的子表达式是等价的, 只需计算一次。

公共 (public) : C++ 类的公共成员对所有函数都是可访问的。

工具箱 (Toolkit) : AIXwindows 和“增强的 X-Windows”中 C 语言数据结构和子例程的集合, 总的说来, 它加速了以 C 语言编写的兼容应用程序的图形用户界面的开发。

功能部件 (feature) : 编程或硬件选项, 通常在需额外开支才可获得。

功能等价性 (functional equivalence) : 一种情况, 其中给定问题域的不同算法产生同样的结果。

功能键 (function key) : 指请求操作但不显示或打印字符的键。这包含那些通常产生一个打印字符, 但在与代码键配合使用时产生一种功能作为替代的键。另见字符键 (character key)。

共享产品对象树 (Shared Product Object Tree, SPOT) : (1) `/usr` 文件系统的一个版本, 无盘客户机将此系统作为自己的 `/usr` 目录装入。(2) 对于 NIM, 指 `/usr` 文件系统或在 NIM 环境中由服务器导出用于远程客户机的等价的文件系统。

共享端口 (shared port) : 通信应用程序使用的端口 (例如 UUCP), 用于确保对某端口的专有访问。

共享库 (shared library) : 由 `ld` 命令创建的库, 它包含至少一个多个进程可以使用的子例程。程序与子例程如以前那样链接, 但是不同子例程的公共代码并入一个库文件中, 此文件可在运行时装入, 且由许多程序共享。识别共享库文件的密钥保留在每个子例程报头中。

共享内存标识 (shared memory ID, shmid) : 指定给用于某个特定进程内的共享段的标识。另见 *文件描述符 (file descriptor)*。

共享内存 (shared memory) : 可以同时被一个以上的协同进程访问的内存区域。

共享锁 (shared locks) : 共享锁是一种在其中多个事务可以同时锁定要读取的数据项的锁。另见 *排它锁 (exclusive lock)*。

供应商标识 (vendor ID) : License Use Management 中许可产品的供应商的标识。通过供应商标识, 许可证服务器可以区分网络中建立的任何数量的供应商。供应商标识是一个“网络计算系统通用唯一标识”的特定于 License Use Management 的用法。

供应商密码 (vendor password) : 在 License Use Management 中, 指具有供应商信息编码的字符串以及一个供应商标识, 它在许可证数据库中建立了某许可证产品的供应商。

公用电话网 (public telephone network, PTN) : 用交换线路或非交互线路提供声音和数据通信服务的公共通信载波网络。

公用交换电话网 (public switched telephone network, PSTN) : 用交换线路提供声音和数据通信服务的公共通信载波网络。

公用交换网 (public switched network, PSN) : 一种通信服务, 通过它用户通过拨号特定服务地址号连接。

公用节点 (public node) : 不运行监视器系统组件或服务器的任何节点。例如, 在公用节点上运行的客户机应用程序。

公用数据网络 (public data network, PDN) : 用交换或非交互线路提供数据通信服务的公共通信载波网络。

公众通信公司 (common carrier) : 提供通信服务给公众的任何由政府管理的公司。

工作段 (working segment) : 其页面由磁盘调页空间的槽而非磁盘上永久位置支持的段。

工作负荷并发 (workload concurrency) : 系统达到当有处理器时始终有许多可分派线程的理想状态的程度。

工作负荷 (workload) : 代替系统正在做的工作的请求序列, 比如命令、I/O 操作以及子命令库的调用。此术语通常指已经捕获的可重复的工作负荷 (通过 shell 脚本、远程终端仿真器), 这样它可以用来评估系统更改的性能影响。

工作集 (working set) : 程序的可执行代码和 / 或数据区的那部分, 它被集中使用, 所以对于保持尽可能快的存储器类型非常重要。这样程序指令的高速缓存工作集就是当程序以最高速度运行时, 需要保存在指令高速缓存内的程序高速缓存线路的集合。

工作目录 (working directory) : *当前目录 (current directory)* 的同义词。

工作区背景 (workspace background) : CDE 中, 显示器上没有被窗口和图标的覆盖的部分。

工作区菜单 (Workspace menu) : CDE 中, 通过指向工作区的未占用区域并单击鼠标 3 键而显示的菜单。

工作区对象 (workspace object) : CDE 中, 驻留在工作区而非窗口查看器中的对象。工作区对象包含窗口、图标 (最小化窗口) 以及从“文件管理器”和“应用程序管理器”拖放到工作区的对象。

工作区管理器 (Workspace Manager) : CDE 中, 在多个工作区中控制窗口大小、布局 and 操作的软件应用程序。

工作区界面 (workspace interface) : 每个工作区独立为帮助分派器的而显现的界面。

工作区切换器 (workspace switch) : CDE 中使您能够从多个工作区选择一个工作区的控件。

工作区切换区域 (workspace switch area) : 在 CDE 中, 指“前面板”中间的矩形区域, 它包含工作区切换器、“锁定”控件、“退出”按钮和忙指示灯。

工作区图标 (workspace icon) : CDE 中从“文件管理器”复制到工作区的图标。

工作区 (work area) : CDE 中控件和文本出现的窗口部分。

工作区 (workspace) : (1) XDS/XOM 中, 其中可以创建某 OM 类的 OM 对象, 以及实现支持那些 OM 类的对

象管理功能的空。 (2) CDE 中当前的屏幕显示, 它包含图标和窗口以及放置显示图标的未被占据的屏幕区域。

工作文件 (work file): 用于临时存储正在处理的数据的文件。

工作站公共链路访问 (Common Link Access to Workstation, CLAW): 定义通道命令 (用于主机和通道连接适配器的之间) 的体系结构。

工作站 (workstation): (1) 输入 / 输出设备的配置, 操作员在这些设备上工作。 (2) 一个终端或微型计算机, 通常连接至大型机或网络, 用户可在它上面执行应用程序。另见终端 (terminal)。

构建代码 (build code): 运行构建过程把源文件转换为超文本数据库的程序。构建代码定义了什么项无法构建成超文本, 什么项不行。例如, 非法字体不会构建, 因为它没有包含在构建代码里。

构造函数 (constructor): 一种与类具有相同名称的特殊 C++ 成员函数。它用于构造类对象并可初始化它们。

构造字符串 (conformant string): 在 Pascal 中, 一种字符串, 其声明长度与形式参数长度不匹配。另见形式参数 (formal parameter)。

孤立文件 (orphaned files): 用 fsck 命令不能到达文件。

固定存储器空间 (fixed storage space): 在系统配置期间定义的任何存储设备, 是 DASD 系统的组成部分。如果在正常操作的某些时候, 一个固定的存储设备不可用, 那么操作系统检测到一个错误。

固定的部分 (记录的): 在 Pascal 中, 对特定记录类型的所有实例都为公共的记录的部分。

固定的图标 (fixed icon): 固定在桌面上的图标。固定的图标通常用于那些需要使用公共桌面环境的文件, 因此经常显示于桌面上, 如主目录。

固定点类型 (fixed point type): 请参阅实类型 (real type)。

固定框 (fixed box): (1) 几何学管理技术。 (2) 一类边界框, 有由父框创建的固定的子框数目。这些受管子边界框不存在几何学管理器请求。

顾客信息控制系统 (Customer Information Control System, CICS): 一许可程序, 使输入远程工作站的事务可由用户编写的应用程序并行处理。它包含编译、使用和维护数据库的工具。

挂钩例程 (hooking routine): 与库连接但保留在库外的函数; 协议扩展过程。stub 的同义词。

挂机 (on-hook): 取消激活, 与电话设置有关。电话不正处于使用中的情况是“挂机”。对照 摘机 (off-hook)。

关闭连接 (connection close): 废弃客户机产生的所有事件, 服务器复位自身状态为无连接。

关闭 (close): (1) 结束活动并从屏幕除去窗口。 (2) 一个结束文件和程序间连接的数据操作函数。对照 打开 (open)。

关键资源 (critical resource): 一系统资源, 其速度和 / 或大小限制可以处理的特殊工作负荷的速度。

关键字 (keyword): (1) 一个在编程语言中预定义的字。一个保留字。 (2) 在编程语言中, 表征语言构造的一个词法单元。一个关键字通常有标识符的格式。 (3) 一个标识参数的名称或符号。 (4) 由一个特定字符串组成的命令行操作数部分。

关联变换 (associative transformation): 一中涉及到更改表达式顺序的优化。关联变换产生数学上完全相同的结果, 但不总是按位相同的结果。

关联子 (correlator): 一个在两个或更多程序之间传递的值, 允许共有资源的关联和识别。

关联 (associativity): 使用运算符将操作数分组的顺序 (从左到右或从右到左)。

关系表达式 (relational expression): (1) 一种描述两个算术表达式或两个数据项间关系 (如大于或等于) 的逻辑语句。 (2) 一种用一个关系运算符连接起来的前后两个算术表达式构成的表达式, 其结果只能是“真”值或“假”值。

关系运算符 (relational operator): (1) 用于表示关系条件或关系表达式的一个保留字或符号。 (2) 一种运算符, 对两个操作数进行比较并产生一个布尔值。

关系转换 (relational transformation): 一种转换, 它更改算术关系表达式为其逻辑补码的非。例如, $A > B$ 的关系变换是 $\text{NOT}(A \leq B)$ 。

关于条目帮助控件 (On Item Help control): CDE 中, 前面板上的“前面板”控件一般是访问条目帮助信息。

关于条目帮助 (On Item help): CDE 中, 当从帮助菜单中选择关于条目时, 光标更改为问号然后可以通过单击对象获得关于特定的命令、操作、对话框, 或者控件的信息。

管道 (pipe): (1) 引导数据以使一个进程的输出成为另一个进程的输入。一个命令的标准输出可以用管道运算符 (|) 连接到另一个命令的标准输入。像这样连接的两个命令构成一个流水线。 (2) 发送进程和接收进程之间的一个单向通信路径。另见流水线 (pipeline)。

管理程序调用 (supervisor call, SVC) : 中断正在运行的程序并将控制传递给管理程序的指令, 这样管理程序就可以执行指令指示的特定服务。

管理程序 (supervisor) : 操作系统控制程序的一部分, 它协调资源的使用并维护处理单元操作的流动。

管理环境 (administration environment) : 易于系统配置和管理的这部分监视器。

管理器类 (Manager class) : 在 AIXwindows 中, 一个提供资源和功能以实现某种功能的元类, 比如键盘接口和遍历机制。它构建于 **Core**、**Composite**、和 **Constraint** 类。

管理器 (manager) : 请参阅设备管理器 (device manager)。

光标标识 (cursor ID) : 增强的 X-Windows 中, 与每种唯一类型的光标相关联的唯一识别号码。

光标轮廓 (cursor glyph) : GL 中一个 16x16 或 32x32 光栅模式 (确定光标形状的位图)。GL 光标轮廓可以是一或二位深; 因而 GL 光标可使用多达三种颜色。颜色 0 始终是透明的。

光标稳定性 (cursor stability) : 在文件系统 (而不是由 SFS 管理) 中, 在单一文件系统内同时维护多个上下文的能力。SFS 的等价光标由在单一文件中同时获得多个 OFD 的能力提供。

光标移动键 (cursor movement keys) : 用来移动光标 (不改变文本) 的方向键。

光标 (cursor) : (1) 显示器上的可移动符号 (如下划线), 向用户指明下一个输入字符会放在哪里或下一个操作指向哪里。(2) “增强的 X-Windows” 中屏幕上指针的可视形状。由热点、源位图和色对构成的光标。(3) 一个可以通过输入设备 (通常为鼠标) 的方法在屏幕上移动的图元 (如箭头)。

光纤 (fiber optics) : 光学技术的分支, 研究放射性能源通过光纤进行传输, 光纤是由诸如玻璃, 复合硅, 塑料等透明资料制造的。

光栅图形 (raster graphics) : 计算机图形学中, 显示的图像由按行列排列的像素数组组成。

光栅镶嵌 (raster mosaics) : 注释文本字体中的一个区域, 它包含对字体中每个字符的定义。

光栅子系统 (raster subsystem) : 有关图像被转换和缩放到屏幕坐标后的系统部分。它包含扫描转换和显示。

光栅字体 (raster font) : 一种字体, 其中字符由光栅位图直接定义。另见字体 (font) 和图元字体 (primitive font)。

光栅 (raster) : 视频显示的区域, 通过水平和垂直扫过显示的电子束来覆盖它。通常, 在垂直回扫间隔期间, 显示的电子水平地从上到下扫过每一行, 然后返回到顶部。另见帧缓冲区 (frame buffer)。

广播拓扑 (broadcast topology) : 所有站都并行地与介质连接, 并且能够并发地接收由连接到介质的任何其它站传输的信号拓扑结构。

广播 (broadcast) : 同时向多于一个目的地传输数据。

广域网 (wide area network, WAN) : 在比局域网提供的服务的区域更大的地理区域内提供数据通信能力的网络。包含散布在巨大地理区域上计算机的网络, 此区域涉及多个城市、状态或国家或地区。

规程 (discipline) : 用来排列打印任务的排序方法, 可以是 FCFS (先来先服务, first-come-first-served) 或 SJN (最短作业优先)。另见先来先服务 (first-come-first-served) 和最短作业优先。

归档库 (archive library) : 一个存储程序以保管的地方。

归档 (archive) : (1) 存储程序和数据以保管。(2) 单个或多个文件的副本或数据库副本, 保存该副本以备原始数据损坏或丢失。

规范处理 (canonical processing) : 按照已定义规则集处理发生的情况。这是 shell 和简单命令典型使用的输入风格。

规格化设备坐标 (NDC) : 在 GL 中, 坐标的范围是 -1 到 1。所有绘制在单元立方体内的图元出现在屏幕上 (除非被屏幕掩码掩盖)。另见转换 (transformation) 和单元立方体 (unit cube)。

规则文件 (rule file) : 包含规则的一种文件, 它确定公共桌面环境的外观和行为。

滚动列表 (scrolled list) : 滚动的列表。另见滚动 (scroll)。

滚动球 (roller ball) : 鼠标器内的球, 它与桌面或其它坚硬表面相接触。

滚动区域 (scroll region) : 在 AIXwindows 中, 指包含两个箭头和一个滚动块的滚动条 (ScrollBar) 小窗口的矩形区域。

滚动条 (scroll bar) : 目录窗口边上的垂直或水平条, 它允许您滚动文件来查看窗口边界外的内容。它是由滑动

块、滚动区域和滚动箭头构成的图形设备。用户可以通过在滚动区域内向上或向下滑动滑动块或按滚动箭头之一来变换视图。这使靠近滚动条的视图在窗口中向上或向下滚动。

滚动文本 (scrolled text)： 滚动的文本。另见 *滚动 (scroll)*。

滚动 (scroll)： 垂直或水平移动文本，以便查看屏幕或窗格边界外的信息。

滚动 (scrolling)： 显示器屏幕上图形或文本信息的水平或垂直的移动。

过程地址 (procedure address)： AIXwindows 工具箱中特定程序过程的位置。

过程间分析 (interprocedural analysis)： 为变量、返回值和全局数据之间信息关系指定引用过的进程。

过程 (procedure)： (1) 请参阅 *shell 步骤 (shell procedure)*。(2) 在编程语言中带有形式参数或没有形式参数的块，其初始化通过过程调用的方法实现。(3) 为解决一个问题所采取的行动历程的描述。(4) 使得一个或多个程序被执行的一组相关的控制语句。(5) *函数 (function)*；功能的同义词。(6) 对于 Ada 编程，请参阅 *子程序 (subprogram)*。

过滤器元 (filter primitive)： 一个按照指定的准则分隔数据的程序。

过滤器 (filter)： (1) 读取标准输入数据、修改数据和发送数据到显示屏幕的命令。(2) 一个设备或程序，用于分离数据，信号或资料以同指定的标准一致。

国际标准化组织 (International Standards Organization, ISO)： 一个使商品和服务标准化的国际化团体。对于“增强的 X-Windows”，标准化与的字符集和字体有关。

国际电话电报咨询委员会 (Consultative Committee on International Telegraphy and Telephone, CCITT)： 一个联合国专门标准团体，其成员包括公众通信公司，负责为国际电信表示字母表、图形、控制信息和其它基础信息交换问题设计和建议推荐。

国际字符支持 (international character support)： 与本地语言支持 (*national language support*) 同义。

国家或地区代码 (country code)： 在 X.25 通信中，网络用户地址 (对公用网络) 中国家终端号前的 3 位的数字。

[H]

哈夫曼编码 (Huffman coding)： 用来压缩数据的字符编码技术。

涵盖外围设备节点 (converged peripheral node)： SNA 中，一种具有有限寻址和路径控制路由能力的物理单元类型。它提供到其它 SNA 节点的常规连接性，并支持并行会话、每 LU 多会话、主和从属 LU 和每节点多线路。

函数调用 (function call)： 一个表达式，它使执行路径从当前函数转至指定函数，并求值所调用函数提供的返回值。函数调用包含控制要转至的函数的名称，以及用括号括起来的值列表。

函数定义 (function definition)： 函数的完整描述。函数定义包含一个可选的存储类说明符、一个可选的类型说明符、函数声明符、可选的参数声明和块语句 (函数体)。

函数模板 (function template)： 提供如何构造一组相互关联而又各自独立的 C++ 函数的蓝图描述。

函数声明符 (function declarator)： 指对函数进行命名的函数定义的一部分，它提供有关函数返回值的附加信息，并列出函数的参数。

函数引用 (function reference)： 某个表达式中内在函数名称或用户函数名称的外观。

函数指针 (function pointer)： 给出函数或过程的位置的标识符。

函数子程序 (function subprogram)： 请参阅 *函数 (function)*；功能。

函数作用域 (function scope)： 在函数中声明的 C++ 标签具有函数作用域，可用于该函数中的任何地方。

函数 (function)；功能： (1) 特定用途的实体，或其特征操作。(2) 诸如回车或换行之类的机器操作。(3) 返回单个变量的值的子例程，通常有单一的出口，例如计算数学函数的子例程。*过程 (procedure)* 的同义词。对于 Ada 编程，请另见 *子程序 (subprogram)*。

涵义名 (purported name)： 依照句法是一个名称的一种构造，但是它还没有被显示为一个有效名称。

行编辑器 (line editor)： 一个一次仅显示一行数据，仅允许通过输入命令读取和修改数据的编辑器。

行打印机 (line printer)： 一次一行字符作为打印输出的打印机。行打印机的输出是不变宽度的字符。

行调步 (line pacing)： 继续传输前，发送一行后跟一个等待间隔。另见 *调步 (pacing)*。

行高 (line height)： 对一行文本的垂直度量，该度量从一行的底部到下一行的底部。行高度通常用点 (point) 来表示。

行号 (line number)： 对于 Ada 语言调试器，与包含指定编译单元的 Ada 编译器有关的行。

行宽 (linewidth)： 以像素表示的行的宽度。

行列表管理器 (row column manager)： 包含切换按钮或按钮的一种小窗口。它更新其说明历史帐户属性，该属性允许它确定最后选定哪个切换按钮。

行业标准基准 (industry-standard benchmark)： 为一致采用的或者为一些 (假定为中性的) 发起组织采用的一个基准，用来指定部分计算机系统性能的一个有意义的方式。对于假定的用户设定的性能提高相对应的行业标准基准性能提高，有很多反面例子。

行 (line)；线路： (1) 屏幕上的一个水平显示。(2) 连接到数据电路终端设备 (DCE) 或数据转换交换 (DSE)、或连接几个 DCE 或 DSE 的数据电路部分。(3) 由系统作为一个从工作站的输入单个块所接受的字符串，比如在执行回车前输入的所有字符。(4) 请参阅 X.25 线路 (X.25 line)。

行 (row)： 字符或其它表达式的一种水平排列。

毫秒 (millisecond)： 一个时间度量单位，1/1000 秒。

合成视频 (composite video)： 按照美国的 NTSC 标准和欧洲的 PAL 标准合成声音、色度和同步信号。也称为 NTSC 或 PAL (欧洲)。

合法的 (valid)；有效的： (1) 允许的。(2) 真实的或依照适当标准或权威。

核心顺序控制器 (core sequence controller)： 用于初始程序装入 (IPL) ROM 的三个控制程序之一。核心顺序控制器接受来自初始顺序控制器的控制，并把控制传递给 IPL 控制器。

核心小窗口 (core widget)： 在“增强的 X-Windows”中，包含对所有小窗口都共有的字段的定义。所有小窗口是核心小窗口的子类。另见小窗口 (widget)。

赫兹 (hertz, Hz)： 频率单位，等于每秒一周。

横向显示 (landscape display)： 一个宽度大于高度矩形显示。请参阅图像显示 (portrait display)。

横向向右 (landscape right)： 在从打印机打印出来时打印图像的右边位于纸张的尾部边缘的页面方向。

横向向左 (landscape left)： 在从打印机打印出来时打印图像的左边位于纸张的尾部边缘的页面方向。

宏处理器 (macro processor)： 将宏指令转化为指定值的程序。

宏调用 (macro call)： 当执行、导致在相同源语言中的预定义指令序列执行时的一个单独指令。

宏指令 (macro instruction)： 请参阅宏调用 (macro call) 和宏 (macro)。

宏 (macro)： (1) 程序或文件开始处声明的标号。这个标号可以用来表示分配给声明中的标号的值。(2) 可用于替代其它许多名称的一个名称或标号。(3) 指令或语态的序列，当替代宏指令时由宏生成程序运行。(4) 语句的集合为从单独的源语句中生成汇编程序序列语句，定义汇编程序语句的名称、格式和条件。(5) 当调用宏时，一系列 Ada 语言调试器命令得到执行。采用宏选项，就可以定义和操作新的调试器命令。另见例程 (routine)、语句函数 (statement function) 和子例程 (subroutine)。

后备存储 (backing store)： 由“增强的 X-Windows”服务器维护的离屏和保存的像素集合。

后处理器 (post processor)： 一种计算机程序，它影响某个最后计算或组织。在文本格式中，后处理器命令转换 **nroff** 和 **troff** 命令的输出，以用于某些打印机、排版机或照相排版机。

后代 (descendant)： 请参阅孩子 (child)。

后端程序 (backend program)： 请参阅后端 (backend)。

后端 (backend)： 发送输出到特定设备的程序。有两种类型的后端：友好的和非友好的。

后台活动 (background activity)： 请参阅后台进程 (background process)。

后台进程 (background process)： (1) 一种无需操作员干预但在工作站做其它工作时可以由计算机运行的进程。(2) 一种程序执行方式，此方式中 shell 提示用户输入下一个命令之前无需等待程序结束。对照前台进程 (foreground process)。

后台 (background)： (1) 在多道程序设计中，低优先级、非交互式程序运行下的环境。对照前台 (foreground)。另见程序级别 (program level) (2) 在 CDE 中，窗口的底层区域，其中显示各种元素 (如按钮和列表)。

后退 (Backtrack)： 在 CDE 中的帮助管理器中，一个用于跟随向后链接 (之前遍历的逆向顺序) 的按钮和导航。

后缀 (suffix)： (1) 连接至文件名末端的字符字符串，它帮助识别其文件类型。(2) 已经处于通话中的呼叫者拨的代

码。(3) 文件名的一部分, 添加在末尾, 通过某种标点符号 (比如点号(.)) 和其它后缀或基本文件名称隔开来。

呼出表 (callout table): 一张内核表, 保持所有睡眠状态进程和处于等待状态的通道的跟踪。

呼出呼叫 (outgoing call): 在 X.25 通信学中, 由另一个数据终端设备 (DTE) 发起的呼叫。

呼出 (callout): 一个内核参数, 确定可同时暂挂的最大已调度活动数。

呼叫冲突 (call collision): 请参阅冲突 (collision)。

呼叫重定向通知 (call redirection notification): X.25 通信中, 一可选 CCITT 指定设施通知呼叫方: 呼叫已经重定向到另一个 DTE。

呼叫接受信息包 (call-accepted packet): X.25 通信中, 呼叫方 DTE 发送的通知呼叫接收方 DCE 的管理分组。另见包 (packet); 信息包; 分组。

呼叫连接信息包 (call-connected packet): X.25 通信中, DCE 发送的通知呼叫方 DTE 呼叫建立结束的管理信息包。另见包 (packet); 信息包; 分组。

呼叫请求信息包 (call-request packet): X.25 通信中, DTE 发送的请求通过网络建立呼叫的管理信息包。另见包 (packet); 信息包; 分组。

呼叫请求 (call request, CRQ): 计算机发送的信号, 请求数据通信设备与网络中的另一台计算机建立通信连接。

呼叫用户数据 (call user data, CUD): X.25 通信中, 用户应用程序可选地把数据包含在呼叫请求分组。

忽略 (ignore): DOS 中忽略设备错误并继续处理的选项。

互操作性 (interoperability): 一种系统和产品与其它系统或产品相互作用而不需要客户方面特殊努力的能力。随着“网络就是计算机”这一概念现实, 互操作性已成为信息技术产品提高质量的重要性。

互斥机制 (mutual exclusion mechanism): 一种阻止两个单个执行代码片段相互干扰对方使用特定数据对象的方法。例如, 如果一个线程正执行修改一个共享数据结构的函数, 那么应用程序可能需要阻止其他线程在修改完成之前试图同时读取数据。

互斥锁定 (exclusive lock): 锁定的一种, 只有持有锁的事务才能以任何方式访问数据。另见共享锁定 (shared locks)。

互斥锁 (mutex): 术语称为相互排斥锁 (mutual exclusion lock)。该类型锁的使用排斥除锁定所有者以外的其他所有线程访问任何锁定的资源。

互连网络 (internetwork): 任一连接多个网络的广域网。

滑动块 (slider): (1) AIXwindows 中, 连接至 XmScrollBar 条小窗口的小型交互式图形对象。此滑动块控制文本信息或图形在显示屏上的垂直或水平的移动。(2) 使用磁道和臂从可用设置值中选择值来设置的控件。臂的位置 (或单独的指示符) 给出了当前的设置值。

坏块 (bad block): 磁盘上永远无法可靠使用的部分。

环境变量 (environment variable): (1) 描述进程操作环境的变量。公共环境变量描述了主目录、命令搜索路径、正在使用的终端和当前时区 (分别为 HOME、PATH、TERM 和 TZ 变量)。(2) 包含在当前软件环境中的变量并因此对任何请求它的调用函数都可用。

环境光线 (ambient light): 三维图形中, 光线到达目标表面之前在场景里的一个或多个表面上的反射光。假定环境光是无方向的, 经过反射面均一地反射到各个方向。在 GL 中, 光方程式使用环境术语模拟环境光线, 而不是实际地计算反射。

环境 (environment): (1) 用户登录时 shell 变量和路径集的设置。这些变量可由用户在以后修改。(2) 用于支持函数运行的逻辑和物理资源的命名集合。(3) 公共桌面环境中, 规则文件、资源和消息文件的集合, 它们定义了一个特定桌面配置的外观和行为。

环绕舍入 (wrap around): (1) 文件中引用点从一行的末尾到下一行的开始或从文件的末尾到另一个文件的移动。(2) 在基于显示器的字处理设备中, 由于显示器水平限制迫使文本的一行可打印行到两行或更多显示行的自动存放。(3) 从存储器中最大可寻址位置到第一个可寻址位置的操作的继续。(4) 寄存器从最高到最低寄存器地址寻址的继续。

环形缓冲区 (ring buffer): 一种应用程序定义的缓冲区, 监视器方式输入放置在其中。应用程序将来自输入设备的数据放置在该缓冲区中。环形缓冲区机制显著性地缩短了从输入设备到应用程序的输入数据路径。

环形网络 (ring network): (1) 一种网络, 其中每个节点有两个连接到节点的分支。(2) 一种网络配置, 其中设备由单向传输链路连接而形成闭合路径。另见环形 (ring)。

环形 (ring): (1) 用于在 LAN 中分布数据的一种方法。(2) 在 FDDI 中, 由物理介质连接的两个或多个站, 其中信

息在活动的站之间按顺序传递，每个站都轮流检查和复制和重复该信息，最后将它返回到起始的站。另见 *环形网络 (ring network)*。

缓冲区长度 (buffer length)：可以存储在一个给定缓冲区中的数据段最大长度。

缓冲区 (buffer)：(1) 一临时存储单元，特别是以一定速率接受信息又以另一种速率传送信息的临时存储单元。(2) 一可调整的内存存储空间，临时为执行输入或输出保留，从其中读取或写入数据。

换行字符 (new-line character, NL)：使打印或显示位置移动到下一行的控制字符。该字符在 C 语言中用 '\n' 表示。通常回车符和 NL 隐式相关联。

换行 (linefeed)：一个使得输出设备向前移动一行的 ASCII 字符。

幻数 (magic number)：表明文件类型的数字或字符串常量。

幻影工具 (ghost instrument)：在 Performance Toolbox 中，指控制台上的空白空间。当为一个系统设计的控制台包含的工具不可用于当前系统时，通常会使得某个工具成为幻影工具。幻影工具占用某个空间并阻止新工具定义在该相同空间中，或阻止其它工具移动或调整大小以使用该空间。

幻影图标 (ghost icon)：表示文件丢失的图标。

灰度 (gray scale)：(1) 指“增强的 X-Windows”中一类退化的伪彩色，其中任何给定色彩表项中的红色、绿色和蓝色值是相等的，因而产生灰色阴影。灰度值可以动态更改。(2) 也称为 *GrayScale*，是一个值。(3) 在灰度适配器中，与所产生灰色阴影对应的不同亮度级别。另见 *单色显示 (monochrome display)*。

恢复过程 (recovery procedure)：(1) 当显示屏幕上出现一条错误消息时操作员执行的一种操作。该操作通常允许程序继续或允许操作员运行下一个作业。(2) 一种方法，它将系统返回到主要系统错误发生的点，并再次运行最近的关键作业。(3) 一种进程，其中指定的数据站尝试解析传送数据期间出现的错误条件冲突。

恢复 (restore)：返回到原始值或原始图像；例如，从软盘恢复库。

回车符 (carriage return)：(1) 文本数据中，表示在下一行左页边距处继续打印的操作。回车符等同于打字机的回车符。(2) 通常表示一个命令行结束的击键。

回调函数 (callback function)：与 *回调 (callback)* 同义。

回调例程地址 (callback routine address)：给定回调例程的内部地址。

回调例程 (callback routines)：如果（当）某个指定条件满足时调用的过程。另见 *回调 (callback)*。

回调列表 (callback list)：(1) 如果（当）某个指定条件满足时调用的过程列表。(2) 在 AIXwindows 中，单独小窗口可以在需要时定义回调列表。

回调原因 (callback reason)：如果条件满足，则调用回调过程。

回调 (callback)：如果（当）某个指定条件满足时调用的过程。通过在回调列表里指定过程来实现。与 *回调函数 (callback function)* 同义。另见 *回调例程 (callback routines)*。

回放 (playing back)：在 Encina 中，当重新启动一个使用可恢复数据的 TP 系统时的恢复服务的操作。当启动时，恢复服务回故事务的日志记录，该日志记录已准备好但实际上没有提交，这就保证该可恢复的数据的状态反映 TP 系统维护的记录。

回滚 (backscrolling)：(1) 纸张通过打印机时，颠倒正常的流动方向（从上到下）。(2) 如果引用的是视频显示器，则指从上到下移动文本穿过视域。

回滚 (rolled back)：撤销代表未完成（是异常终止的）事务执行的任何修改。不管是什么原因，异常终止的事务所做的任何更改必须被撤销。一旦事务回滚，没有证据表明事务曾经被尝试保留在事务处理系统日志的记录之外。另见 *异常终止 (abort)*。

回呼 (call-back)：UUCP 文件 USERFILE 的一个特征，告诉远程系统它要访问的本地系统是否回拨检查它的身份。

回绕 (rewind)：从自变量列表选择一个以前的项，而不是选择下一个或当前的项。

回溯 (traceback)：对于 Ada 调试器，指例程的列表，这些例程位于您在调试的代码上的调用链中。例如，如果您在一个 Ada 过程内设置断点，您将看到所有调用您的 Ada 过程的过程列表，以这些过程的调用顺序排列。所有调用链中的调用过程列表到（但不包括）调用开始最高级别列表的操作系统这一项。

回显 (echo)：(1) 在通讯信道中的反射信号。在通讯终端，每个信号显示两次，一次是在进入本地终端时，第二次在返回通讯链路时。允许校验信号的准确性。(2) 在计算机图形学里，当前值的直接通知，它由操作员的输入设备在显示控制台提供。(3) 在字处理中，在输入的同时，打印或显示每个字或行。

回写式高速缓存 (write back cache)：“增强的 X-Windows”中，由库高速缓存 GC 以允许将各独立的更改请求合并成一个协议请求。另见**高速缓存 (cache)**。

汇编程序 (assembler)：把汇编语言指令转换为目标代码的计算机程序。与**汇编程序 (assembly program)**同义。

汇编程序 (assembly program)：与**汇编程序 (assembler)**同义。

汇编语言 (assembler language)：一种符号编程语言，其中指令集包含机器指令，它的数据结构与机器的存储器和寄存器直接对应。

汇编 (assemble)：把汇编程序翻译为计算机语言。汇编通常通过把计算机语言操作码替换为汇编语言操作码，把绝对地址、即时地址、可重分配地址或虚拟地址替换为符号地址来完成。

会话服务器 (session server)：CDE 中提供联网会话的系统。会话文件驻留在会话服务器上并且只要您登录到网络中的服务器上，就能使用这些文件。

会话概要文件 (session profile)：对于 3270 主机连接程序 2.1 和 1.3.3，描述客户机系统与 System/370 主机计算机之间会话特征的概要文件。另见 *3270 主机连接程序 (3270 Host Connection Program)* 和概要文件 (*profile*)。

会话管理器 (Session Manager)：CDE 中的一个软件应用程序，它控制保存会话、恢复会话、屏幕锁定和解锁以及屏幕保护程序的使用。会话保存后，桌面环境的状态（图标位置、打开窗口的大小与位置、应用程序的打开/关闭状态以及当前颜色选用板等等）被保留下来这样它可以在下一次登录时恢复。

会话级别调步 (session-level pacing)：SNA 中的流向控制技术，它接收会话控制数据传送率（它在此速率接收请求单元）。当发送方可以比接收方处理请求更快地发送请求时，它用于防止接收方负荷过多未经处理的请求。

会话记录 (session records)：在记帐系统中连接时间和连接显示站的线路使用的记录（从登录和注销记录产生）。

会话密钥 (session key)：用于 Kerberos 规范。另见**对话密钥 (conversation key)**。

会话日期 (session date)：与会话相关的日期。另见**创建日期 (creation date)** 和**系统日期 (system date)**。

会话 (session)：(1) 程序或设备能互相通信的一段时间。(2) 控制本地 LU、远程 LU、方式和连接的某类型资源的名称。(3) 在网络体系结构中，在站之间为通信而建立、维护和释放连接的设施的关联。(4) 工作站的用户能和交互式系统通信的一段时间，通常指在登录和注销间经过

的时间。(5) 在 SNA 中，两个网络可寻址单元 (NAU) 之间的逻辑连接，可以激活、定制此连接以提供各种协议，或应要求取消激活此协议。(6) 在远程通信中，与远程系统或主机系统通信的一段时间。

绘图仪 (plotter)：打印二维图和图表的，用电缆连接到系统的硬拷贝设备。

绘制按钮 (drawn button)：在 AIXwindows 中，一个图形对象，它用一个符号或其它在其表面上的绘制的图象来模拟一个真实世界的按钮。

混合许可证 (compound license)：在 License Use Management 中，一种允许系统管理员为给定许可证个数生成许可证密码的许可证类型。当管理员需要某个数量的许可证，但还不知道什么机器或谁会使用它们时，这样的许可证是很有价值的。混合许可证可生成节点锁定或非节点锁定的两种许可证之一，而不是两者。

混合字符串 (mixed string)：一个由 DBCS 字符和单字节字符混合组成的字符串。

活动的 (active)：(1) 文本光标当前所处的长方形窗口称为是“活动的”。(2) 事务生命周期中的状态之一。在此状态期间可访问或修改数据。

活动网关 (active gateway)：可如网络接口一样对待，用于交换路由信息的网关。如果网关已经有一段时间没有如所述的去做，那么删除与网关关联的路由。对照**被动网关 (passive gateway)**。另见**网关 (gateway)**。

活动抓取 (active grab)：在“增强的 X-Windows”中，取实际上由抓取客户机拥有。对照**被动抓取 (passive grab)**。另见**按键抓取 (button grabbing)** 和**抓取 (grab)**。

伙伴 (partner)：在数据通信中，远程应用程序或远程计算机。

获得许可的产品 (licensed product)：一个已经被软件供应商启用的软件产品，以便使用 License Use Management 系统。启用允许一个供应商强制最终用户遵守他们的许可证协议。

[J]

基本编码规则 (Basic Encoding Rules, BER)：一种用来把 ASN.1 值编码成八位字符串的规则集。

基本操作系统 (BOS) 安装 (Base Operating System (BOS) installation)：使机器处于运行状态所需的最少数量软件的安装和配置过程。

基本对话 (basic conversation)：两事务程序之间的连接，允许它们交换包含 2 字节前缀（用于指定记录长度）的逻辑记录。LU 1、2 和 3 不使用 2 字节前缀；但是 LU

1、2 和 3 对话必须是基本对话。服务事务和 LU 1、2 和 3 应用程序事务程序使用该对话类型。对照映射对话 (*mapped conversation*)。

基本名称 (base name)： (1) 个完整路径名称右边的最后元素。 (2) 没有指定父目录的文件名。 (3) 在 Ada 语言中，没有指定类型限定符 **lib/** 或 **sec/** 的编译单元名称。 (4) 在 CDE 中，图标文件的文件名减去文件名后缀可获得大小 (.l、.m、.s、.t) 和类型 (.bm、.pm)。例如，名称为 myicon.m.pm 的图标文件的基本名称是 myicon。

基本权限 (base permission)： 分配给文件所有者、文件或组或其它要访问此文件的人的访问方式。访问方式包含读 (**r**) 权限、写 (**w**) 权限和执行 / 搜索 (**x**) 权限。另见任意访问控制 (*discretionary access control*)。

基本实型常量 (basic real constant)： 一个包含小数点的十进制数字字符串，表达一个实型值。

基本输入 / 输出系统 (Basic Input/Output System)： 请参阅 BIOS。

基本网络实用程序 (Basic Networking Utilities, BNU)： 一组程序和文件，有时称为 UNIX 到 UNIX 复制程序 (UUCP)，提供基本网络实用程序，例如命令 **uucp**。BNU 包含一组目录、文件、程序和命令，允许用户通过专线或电话线与远程 UNIX 系统通信。另见 UNIX 到 UNIX 复制程序 (*UNIX-to-UNIX Copy Program*) 和 UUCP 登录标识 (*UUCP login ID*)。

基本增量 (basic increment)： 设备能达到的最小动作单元。

基标量类型 (base scalar type)： 在 Pascal 中，从中可派生出集合类型或子界类型的类型。另见子界标量类型 (*subrange scalar type*)。

基础编辑器 (underlying editor)： 一种大型编辑器程序 (比如 vi)，它们的另一种编辑器程序 (比如 vedit) 是一个有限子集 (*limited subset*)。

基带系统 (baseband system)： 一种通信系统，信息经编码、调制然后在传输介质上传输而无需替换或改变信息信号的频率。在介质上的任何一点，一个时刻仅存在一个信息信号。

基地址寄存器 (base address register)： 与基寄存器 (*base register*) 同义。

基地址 (base address)： 把符号引用解析为存储器地址的起始地址。

几何管理 (geometry management)： 几何 (*geometry*) 的同义词。

几何文本 (geometric text)： 其字符字体由需要用来绘制字符的笔划的数学描述定义的文本，而不是光栅图像。也称为可编程字符集 (*programmable character set*) 或笔划文本 (*stroke text*)。

几何 (geometry)： 在“增强的 X-Windows”中，(1.) 屏幕上的窗口大小和位置。(2.) 小窗口的大小可使用几何管理例程进行更改。布局 (*layout*) 和几何管理 (*geometry management*) 的同义词。

激活 (activate)： 用鼠标指针指向并双击，成功地引起某事发生。

基寄存器 (base register)： 一个程序员选择以包含地址的通用寄存器。与基地址寄存器 (*base address register*) 同义。另见索引 (*index*)。

基类 (base class)： 一种 C++ 类，从其中可派生出其它类。基类本身也可以从另一个基类派生。

奇偶错 (parity error)： 当接收到的数据通不过接收系统的预期奇偶性校验时发生的传输错误。这个错误通常是由于发送和接收系统有不同的奇偶性校验设置造成的。

奇偶 (校验) 位 (parity bit)： 用来使组中的所有数字的总数总是奇数 (奇校验) 或总是偶数 (偶校验) 的附加到一组二进制数字的二进制数字 (位)。

奇偶校验 (parity check)： 确定在二进制数字数组中的 1 (或 0) 的数目是奇数还是偶数的测试。

机器对象 (machine object)： 网络安装管理数据库的项，表示一个机器的配置。

机器语言 (machine language)： 一种可由计算机直接使用的语言，不需要中间处理。最终的编译处理输出是一个装入模块包含了机器语言指令。计算机语言 (*computer language*) 的同义词。

机器指令 (machine instruction)： (1) 指导处理器运作的二进制码。编译器和汇编程序将源指令转化为机器指令。(2) 机器语言指令。计算机指令 (*computer instruction*) 和计算机语言 (*computer language*) 的同义词。

机器执行状态 (machine execution state)： 表明机器关闭、引导或运行的状态。该状态是两个机器状态之一。

机器状态 (machine state)： 表示每个机器执行状态和控制状态的一种状态。

机器字 (machine word)： 字 (*word*) 的同义词。另见计算机语言 (*computer language*)。

基数样条三次曲线 (cardinal spline cubic curve)： 计算机图形学中，其端点为四个控制点中的第二和第三个的三次曲线。一系列具有连续斜度并通过除第一个和最后一

个控制点外所有控制点的基数样条曲线。另见参数三次曲线 (*parametric cubic curve*)。

基数 (base number)：自我校验字段部分，从中可计算出校验位。

基线 (base line)：请参阅基线 (*baseline*)。

基线 (baseline)：在字体里，一条虚构的线，每个字符的底端与之对齐。

基于 Web 的系统管理器 (Web-based System Manager)：用于管理操作系统中图形用户界面 (GUI) 工具。基于 OO (面向对象, Object Oriented) 方式，基于 Web 的系统管理器使用户能够通过操作表示系统中对象的图标执行管理任务，以此作为学习和记住复杂命令的另一种方法。

基准测试程序 (benchmark)：一个程序，为测试不同体系结构或相同体系结构的不同实现的计算机的相对性能而设计。一严格指定的工作负荷和处理该工作负荷时量化系统性能的方法的结合。性能度量通常源自处理工作负荷所需的时间。

基 (basis)：在 GL 中，曲线或 patch basis 是控制控制点和渐进样条曲线间关系的 4x4 矩阵。B 样条曲线、贝赛尔曲线和基数样条曲线在这里都是不同的，它们有不同的基。

级别 1 (level 1)：物理级 (*physical level*) 的同义词。

级别 2 (level 2)：帧级 (*frame level*) 的同义词。

级别 3 (level 3)：信息包级 (*packet level*) 的同义词。

级别 (level)：(1) 一个软件应用程序的版本。(2) 另见更高的层 (*higher layer*)。(3) 在 X.25 通信中，另见物理级 (*physical level*)、信息包级 (*packet level*)、和帧级 (*frame level*)。

集合点 (rendezvous)：在 Ada 语言中，当一个任务调用其它任务的一个条目时在两个并行任务间发生的交互作用，并且对应接受语句由代表调用任务的其它任务执行。

集合 (collection)：Ada 语言中由访问类型分配程序求值创建的整个对象集合。

级联按钮：AIXwindows 中的一种矩形图形控件，可从另一个图形控件背后显示以提供附加选项或选项范围。

级联菜单 (cascading menu)：一个相关选项的子菜单，当选中文项时调用。通常一个提供级联菜单的选项通过该选项的一个向右箭头来指定。类似于上下文行 (*context line*)。

集线器 (concentrator)：一个 FDDI 节点，为自身能够连接到 FDDI 网络而拥有除那些必需部件之外的附加部件。这些附加部件 (类型为 M) 用于在树型拓扑结构中连接到其它 FDDI 节点 (类型 S)。集线器主要用于允许多于两台单连接站 (SAS) 通信。它也可以连接多个 SAS 到双连接站 (DAS) 环。

吉字节 (gigabyte, GB)：当指内存容量时为十进制记数法表示的 1 073 741 824；在所有其它情况下，将它定义为 1 000 000 000。

极坐标 (polar coordinates)：用自原点的距离和自某参考方向 (通常是自 *x* 轴逆时针) 的角度来度量位置的一种坐标系统。

继承 (inherit)：从父中拷贝资源或者属性到子。

继承 (inheritance)：(1) 在 AIXwindows 和“增强的 X-Windows”，从类层次结构的一个对象超类下行传送类资源到一个对象子类。(2) 一种允许把已存在的类作创建其它类的基类的面向对象编程技术。

寄存器 (register)：(1) 一种存储设备，有指定存储容量，如位、字节或计算机字，它通常用于特殊用途。另见通用寄存器 (*general purpose register*)。(2) 在 NCS 中，对 RPC 运行时库已知的一种接口，从而通过 RPC 机制对客户机可用。**rpc_\$register** 调用寄存器一个接口。(3) 在 NCS 中，在位置代理数据库中输入一个对象及其位置。**lb_\$register** 调用寄存器一个有位置代理的对象。程序可使用位置代理查找调用来确定已注册对象的位置。

记录的范围 (range of records)：将要按顺序处理的多个记录。记录的范围被选定，通过指定绑定要选定的记录的键值，或通过为其应该在非唯一索引中被选定所有匹配记录而指定单个键值。

记录工具 (recording instrument)：在 Performance Toolbox 中，有定期显示统计信息给系统资源能力的一种工具。记录工具通常有一个时间刻度，将当前时间放在右边。当接收到新的读数时将绘图的值移到左边。用于绘图这些记录的图类型包含行、区域、地平线和条形图。对照状态工具 (*state instrument*)。

记录类型 (record type)：(1) 文件中各记录的分类。(2) 在 Ada 语言中，由组件组成的一种记录类型值通常有不同的类型或子类型。对于记录值或记录对象的每个组件，记录类型的定义指定唯一确定记录中组件的一个标识。

记录名 (record name)：一种记录的数据名，在记录描述项中描述。

记录器 (logger)：(1) 记录事件和物理环境的函数单元，通常是关于时间的。(2) 使用户实体登录 (示例，识别本身、自己的用途和进入的时间) 和带着相关数据注销的程序。这使适当的记帐过程按照操作系统来执行。

记录锁定 (record lock): 防止对某文件的某些部分或全部进行写或读的一种锁。另见 **锁 (lock)**。

记录 (record): (1) 在编程语言中, 由数据对象组成的一个聚集, 这些对象可能具有不同的属性, 它们通常都附加有标识。(2) 作为一个单元来处理的一组数据。(3) 作为一个单元来处理的字段集合。另见 **类记录 (class record)**、**小窗口记录 (widget record)** 和 **实例记录 (instance record)**。

计数器标识 (counter identifier): X.25 API 中的计数器的名称。

计数器 (counter): (1) 一个用于累计事件发生次数的寄存器或存储位置。(2) X.25 API 中的一个变量, 当一个信息包到达时加一, 当收到一个信息包时减一。

计算机辅助教学 (computer aided instruction, CAI): 一种数据处理应用程序, 其中计算系统用于辅助学生教学信息。

计算机环境的可移植操作系统接口 (POSIX) (Portable Operating System Interface For Computer Environments, POSIX): 计算机操作系统的 IEEE 标准。

计算机图形元文件 (Computer Graphics Metafile): 一个用于存储面向对象的图形, 独立于设备的图形文件格式。

计算机语言 (computer language): 与 **机器语言 (machine language)** 和 **机器指令 (machine instruction)** 同义。

计算机指令 (computer instruction): 一个可由计算机处理单元识别的指令 (为计算机而设计)。与 **机器指令 (machine instruction)** 同义。

计算机字 (computer word): 与 **字 (word)** 同义。

计算内存 (computational memory): 所有实际内存中的虚拟内存页面的集合是工作存储区或程序文本段的一部分。

计算器 (Calculator): CDE 中一个模拟手持计算器功能的软件应用程序。

计算时间 (computed time): 同步处理的结果—职员程序或服务器按照从多个服务器接收到的值处理计算所需的时间值。

记帐系统 (accounting system): 一个监视系统运作的各个方面的实用程序; 它收集每个事务的详细数据, 并提供工具处理该数据以产生各种报表。

加电复位 (power-on reset, POR): 一种按键序列, 它不用关闭系统的电源就能重新启动操作系统 (或其它程序)。

加电自测 (power-on self-test, POST): 每次系统加电激活的一系列内部诊断测试。

加方式 (add mode): 在加减运算中, 参考末位数, 把小数点记号放在预定的位置的方式。在选择操作中, 允许选择不受键盘导航影响的方式。

伽玛校正 (gamma correction): 指 GL 中对明暗处理应用程序的查找表项进行亮度的对数赋值。由于人眼觉察的亮度是呈对数的而非线性的, 因此伽玛校正必不可少。另见 **伽玛斜面 (gamma ramp)**。

伽玛斜面 (gamma ramp): 指 GL 中三个查找表 (红、绿、蓝每种颜色各一个) 的集合, 与监视器的电子枪连接。伽玛查找表中的项可设置成随不同品牌的监视器之间荧光粉的不同而调整。伽玛查找表中通常装入的是对数曲线。另见 **伽玛校正 (gamma correction)**。伽玛查找表不是色彩映射表的子集, 而是一个独立实体。

加密密钥 (encryption key): 由 **makekey** 命令生成的密钥, 与执行加密的程序一起使用。它的输入与输出通常是管道。

加密 (encrypt): 将明文数据转换成密文文本。

加入队列 (enqueue): 向队列中插入一项。对照出队 (**dequeue**)。

加速键 (accelerator): 在 AIXwindows 中, 用键盘替代鼠标按键操作。例如, 按住键盘上的 <Shift> 和 <M> 键可以和鼠标按键操作所做的相同的方式完成菜单操作。加速键通常提供增强的输入速度和更多方便。

加速数据传送 (expedited data transfer): X.25 通信中, 一可选 CCITT 指定设施。

加速数据协商 (expedited data negotiation): X.25 通信中, 一可选 CCITT 指定设施。

加载程序 (loader): 一个程序读取运行文件到主存储器中, 以至于该文件可以被运行。

假目标 nam (fake target nam): 一个用于 **makefile** 文件的控制名称, 它看起来象一个目标名称, 但是实际上它通知 **make** 命令执行某些不同的操作。

假脱机程序 (spooler): 排队系统 (**queueing system**) 的同义词, 用于将打印作业排队。

假脱机文件 (spool file): (1) 包含为随后打印而保存的输出的磁盘文件。(2) 用于在设备间进行数据传输的文件。

假脱机 (spooling, 外部设备同时联机操作)： (1) 将辅助存储器用作缓冲区存储器。当在外围设备和计算机处理器之间传送数据时，这减少了处理延迟。 (2) 中间设备上的读和写输入以及输出流，它们的格式易于随后的处理。 (3) 当计算机正忙于其它工作时，执行外围操作，比如打印。

间接的块 (indirect block)： 一个包含其它块的指针的块。

间距 (pitch)： 一种打字机输入的宽度单元，它基于水平方向每英寸可设置的字母次数。例如，10 间距输入表示 10 字符 / 英寸。

兼容单元 (compilable unit)： 在 Pascal 中与编译单元 (compilation unit) 同义。另见单元 (unit)。

兼容的 (compatible)： 无需显著更改就能适合计算机并在其上运行相同的程序。

兼容类型 (compatible types)： 不同的数据类型可作为相同运算的操作数。

兼容性 (compatibility)： (1) 无需较大的更改就可可在不同环境中同等地执行任务的能力。 (2) 满足指定接口要求的功能部件能力。

监视调度守护程序 (monitor scheduling daemon)： 运行在应用程序服务器上、为客户机提供到所需的代理程序的链接的一个进程。

监视器方式 (monitor mode, MOM)： 应用程序可以在其中直接访问显示适配器的模式。

监视器 (monitor)： (1) 一台用于观察和验证数据处理系统的操作的设备。 (2) 数据处理系统中一个观察和记录选择的活动以便分析的功能单元。可能用于显示相对正常值重要的偏差，或用于决定实用程序或特定功能单元的级别。 (3) 显示器 (display)；显示同义。

监听标识符 (listen identifier)： 在 X.25 API 中，一个用来监听和接收一个进入呼叫的标识符。

监听 (listen)： 在 X.25 API 中，准备通过指定的 X.25 端口来接收在路由列表中指定的一个项的，满足标准进入的呼叫。

监听 (listening)： 在给定的套接字上，等待网络通信的程序正在监听该套接字。另见套接字 (socket) 和端口 (port)。

检测码 (sense code)： 发送或接收的值，或指明发生何种错误的否定响应。

检查点 (checkpoint)： 一个正在由应用程序使用的可恢复数据的当前状态的快照。检查点用来捕获在备份之间的

可恢复数据的状态，通过提供更新的数据映像从而使重新启动系统（会使用那些数据）所需时间最少。

简单名 (simple name)： 对于 Ada 编程，请参阅声明 (declaration) 和名称 (name)。

简单邮件传输协议 (Simple Mail Transfer Protocol, SMTP)： 一种协议，通常用于网络，其目标为传送邮件。sendmail 命令使用 SMTP 来接受和接收邮件。

剪切列表 (clip list)： “增强的 X-Windows”中，一指定给剪切的矩形列表。

剪切面 (clipping planes)： GL 中，剪切发生之前，图元空间是映射到规格化设备坐标的。剪切面 $x=\pm w$ ； $y=\pm w$ 或 $z=\pm w$ 对应于约束视图平截面的左、右、顶部、底部、近和远平面。另见粗略剪切 (gross clipping)、剪切 (clipping) 和立体 (frustum)。

剪切区域 (clipping region)： “增强的 X-Windows”中，一种图形输出类型。图形上下文中，由位图或矩形定义的图像，用来限制输出到窗口的特殊区域。

剪切 (clip)： 计算机图形学中，除去位于给定边界之外的那部分显示图象。

剪切 (clipping)： GL 中如果图元与窗口边界重叠，则称为剪切。显示出现在窗口中的这部分图元，忽略剩余部分。系统有许多种剪切类型。三维绘制图元剪切为平截面（透视变换）或菱截面（正交投影）的边界。三维剪切也应用于原始字符串，但不是字符本身。二维剪切也会执行，其中所有图形被剪切为窗口 AIXwindows 的边界。二维剪切的区域可以用屏幕屏蔽区控制。另见剪切面 (clipping planes)、精细剪切 (fine clipping)、粗略剪切 (gross clipping)、屏幕屏蔽区 (screenmask)、边界框 (bounding box)、挑选 (culling)、变换 (transformation) 和窗口 (window)。

检索时间 (retrieval time)： 在存储器中定位数据和为后继处理读取它而需要的时间间隔。

剪贴板选择 (clipboard selection)： 选择粘贴到剪贴板并可以粘贴或传输给函数的数据。数据可以包含如文本、图形和小窗口等元素。

剪贴板 (clipboard)： 一存储空间，设在一边用于临时存储和检索在剪切粘贴操作期间的文本或图形。剪贴板中的数据对其它应用程序是可用的。

简易安装 (Easy Install)： 一个用于以软件包的形式或服务更新的应用程序。

键单击 (key click)： 请参阅键盘单击 (keyboard click)。

键范围 (key range)： 表示将顺序处理的记录范围的两个键字段。记录范围的选择通过指定键值（此键值绑定要选择记录的）或者是通过指定单独的键值（与其相匹配的所有记录应被选取在非唯一的索引中）。

键符号 (keysym)： 在键盘的键帽上的一个编码符号。

键盘遍历 (keyboard traversal)： 一个允许用户移动键盘焦点并用一个键控顺序而不是鼠标激活用户界面组件的 X 小窗口资源。

键盘单击 (键单击) (keyboard click, key click)： 由按键电路触点的打开或关闭产生的在传输线路上的瞬时脉冲或波动。

键盘的发送 / 接收 (keyboard send-receive)： 请参阅键盘的发送 / 接收方式 (keyboard send-receive mode)。

键盘发送 / 接收 (KSR) 方式 (keyboard send-receive (KSR) mode)： 在输入和输出功能上，图形显示仿真一个标准 ASCII 终端的方式。

键盘映射 (keyboard mapping)： 一个列表，通常位于概要文件中，在键盘的每个键与显示屏幕上显示的字符、键压下时程序采取的操作之间建立对应关系。另见映射 (mapping)。

键盘抓取 (keyboard grabbing)： 在“增强的 X-Windows”中，客户机可以主动地抓取键盘控制，按键事件将会被发送到该客户机而不是事件应该正常被发送到的那台客户机。

键盘 (keyboard)： 一种有多种的键组成的输入设备，这些键允许用户输入数据、控制光标和指针位置以及控制与工作站的对话。

箭头按钮 (arrow button)： 一模拟带有方向箭头按钮的图形控件。用户使用指针和鼠标按压按钮并开始一些与方向关联的操作。

建议书 X.25 (Recommendation X.25)： 请参阅 X.25。

键抓取 (key grabbing)： 在“增强的 X-Windows”中，可由客户机被动抓取的键盘上的键。或当键按下时，键盘可主动地由客户机抓取。另见抓取 (grab)、按钮抓取 (button grabbing)、指针抓取 (pointer grabbing)、活动抓取 (active grab) 和被动抓取 (passive grab)。

键 (key)： (1) 一个或多个字符用于标识记录并在索引文件中建立记录的顺序。(2) 一个单独的标识符（属于类型 **key_t**），命名了特定的进程间的通信成员。(3) 标识共享库文本映象的名称。(4) 在数据元素集合中的标识符。(5) 与键表中的定义相匹配的字符串。

降序按键序列： 从键字段的最高值到键字段的最低值的顺序安排数据的方法。

交叉引用程序 (cross-referencer)： Ada 语言中的一个工具，提供编译单元内符号声明、分配或引用的所有位置的列表。符号以名称、类和封装单元标识，符号引用通过源文件行号标识。

焦点窗口 (focus window)： 同义于输入焦点 (input focus)。

交付 (回调) (deliver (callback))： 交付回调或上行调用意味着导致其调用。

交付 - 确认位 (delivery-confirmation bit)： 请参阅 D 位 (D-bit)。

交互式处理 (interactive processing)： 一种处理方法，在这种方法中每个系统用户的操作引起程序或系统的响应。对照批处理 (batch processin)。

交互式 (interactive)： 属于一种包含请求和应答的活动，例如一个系统用户和一个程序之间或两个程序之间的活动。

交互作用 (interaction)： 一个用户和一个或多个软件应用程序之间通过输入设备（键盘、鼠标、文件服务器等）和输出设备（显示器、打印机和文件服务器等）进行连续的闭合的相互操作。

交换标识帧 (exchange identification (XID) frame)： 在逻辑链路控制 (LLC) 报头中，传递发送主机特征的帧。

交换标识 (exchange identification, XID)： 当连接第一次建立的时候，与远程物理单元交换的 ID。

交换间隔 (swap interval)： GL 中帧缓冲区的交换间经过的时间量。在系统发出交换前后缓冲区请求之前，它至少会等待 **swap interval** 子例程中指定的时间量。交换间隔以垂直回扫单元度量，大多数系统上三十分之一秒发生一次回扫。交换间隔对于完成平滑流动的动画非常有用。

交换式虚拟电路 (switched virtual circuit, SVC)： X.25 通信中，虚拟呼叫请求的虚拟电路。当清除了虚拟呼叫后，释放此电路。对照永久虚拟电路 (permanent virtual circuit)。另见虚拟电路 (virtual circuit)。

交换网络备份 (switched network backup, SNBU)： 数据通信中，当非交换线路失败使，提供交换线路连接的技术。

交换线路 (switched line)： 数据通信中，计算机或设备间通过拨号建立的连接。对照非交换线路 (nonswitched line)。

交换 (swapping)： (1) 将活动作业临时从主存储器除去，把它保存在磁盘上，并且在先前被第一个工作占据的主存储器区域中处理另一个工作。(2) 在具有虚拟内存的系

统中，将活动页面写至辅助存储器并从辅助存储器将另一个作业的页面读至实存的页面调度技术。

脚本文件：在 Ada 调试器中，指包含用于驱动调试器的一系列命令的文件。当您不能在一个会期内完成调试会话时，脚本文件对所有调试大而复杂的程序非常有用。

脚注文本文 (footnote text)：脚注内的文本。

脚注 (footnote)：放在一栏或一页的文本下面的引用、解释与注释，但在页脚上部的页面主体内。

较高层或较高级 (higher layer or level)：存在于站的分层结构中的控制或处理逻辑的概念层，位于链路层之上，且数据链路功能的实现随其而定（例如，设备控制、缓冲区分配和站管理）。另见级别 (level)。

校验和 (checksum)：(1) 与组关联的组数据和，用于校验目的。(2) 写到软盘上一个扇区，用于错误检测用途的数据。

节点标识 (node ID)：网络中用来标识节点的唯一的字符串。

节点锁定许可证 (nodelocked license)：锁定到特定节点的许可证，使该产品仅可用于该节点。许可证服务器不管理节点锁定许可证。

节点验证 (node verification)：超出网络寻址方案提供的额外的安全级别。节点验证有助于确保连接到正确的远程站点上。仅可用于 LU6.2 连接。另见 BIND 密码 (BIND password)。

节点 (node)：(1) 连接到网络的计算机。(2) 链路的结束点，或者连接到网络的节点结点，通常是两个或更多。节点可以是处理器、控制器或工作站，他们可以通过路由和其它功能加以区别。(3) 在系统网络体系结构中，硬件组成部分连同相关的软件部分，实现了七层结构的功能 (SNA)。(4) 在树结构中的点，下级数据项从它起源。

节点 (nodes)：在网络中连接以形成一个监视单元的系统。

阶段 (phase)：(1) fsck 命令执行的文件系统检查和修复的几个阶段之一。(2) 一个过程的不同部分，在其中执行相关的操作。(3) 排序和合并程序中的一部分，如排序阶段和合并阶段。(4) 数据调用的一部分。

结构标记 (structure tag)：命名结构数据类型的标识。

结构规则表 (Structure Rule Table, SRT)：目录模式的复发属性，此目录具有对专有名称的许可结构的说明。

结构化编程 (structured programming)：在层次模块中组织计算机程序的技术，它使程序更易于调试、修改和替代。通常所有的模块都有一个单独的入口点和一个单独

的出口点。通过在结构中向下传递控制，此结构不具有到结构更高层次的无条件分支。

结构化类型 (structured type)：定义具有多个值的变量的几种数据类型中的任何一个；例如记录和数组。每个值都是结构化类型的组成部分。对照标量类型 (scalar type)。

结构化文件系统 (structured file system)：由单个结构化文件服务器 (SFS) 管理的数据收集。所有对结构化文件系统的访问是通过单个服务器，此服务器使用标识文件系统及其组织的一种特殊类型的文件描述符。

结构化文件 (structured file)：(1) INed 文件的一种特殊类型，它包含专用数据，比如有关文件中数据结构的信息，以及有关对文件进行更改的历史信息。结构化文件可以包含使用表单来显示和编辑的层次数据。(2) Encina 中，其数据组织成特定格式（通常面向记录格式）的文件。

结构化字段 (structured field)：允许为数据流中的传输而将变量长度数据进行编码的机制。另见字段 (field)。

结构 (structure)：包含有序数据对象组的变量。与数组不同的是，结构中的数据对象可以有不同的数据类型。

结果 (result)：由某个操作产生的一种实体。

接口图标 (interface icon)：界面中出现的对选择选项的图形表示。

接口，扩展的 (interface, extended)：请参阅扩展接口 (extended interface)。

接口，限制的 (interface, limited)：请参阅限制接口 (limited interface)。

接口 (interface)：(1) 一个公共边界，但不是内部连接。一个接口可以是一个硬件组件，以链接两个设备或两个甚至更多计算机程序访问的存储器或寄存器的一部分。与命令解释器 (command interpreter) 同义。(2) 两个功能单元之间的共享边界，是由函数特征、公共物理互连特征、信号特征和其它特征所定义的。(3) 链接系统、程序或设备的硬件或软件或两者兼有。(4) 与 shell 同义。(5) 操作的集合。网络计算体系结构指定了用于接口的网络接口定义语言。另见扩展接口 (extended interface) 和限制接口 (limited interface)。

接收超时 (receive time out)：在数据通信中，当在给定的时间周期内未接收到数据时发生的条件。

接收调步 (receive pacing)：在 SNA 中，正在被一个部件接收的消息单位进行的调步处理。对照发送调步 (send pacing)。另见调步 (pacing)。

接收 (receive) : 在 X.25 通信中, 接收来自缓冲区的一个进入包 (如进入呼叫包或数据包)。

结束描述符 (terminal descriptor) : ODM 中, 命名的变量类型 **short**、**long**、**binary**、**char** 或 **vchar**, 用于定义 ODM 对象类定义中的基本数据类型。另见 *short*、*long*、*binary* (二进制)、*char*、描述符 (*descriptor*)、*vchar* 和对象类 (*object class*)。

结束信号 (end signal) : 在在线会议中, 一个互相认可的字符, 该字符显示了参与者建议的结束。常用结束信号是 **o** 和 **oo**。

结束页 (trailer page) : 跟随在打印的文件或打印作业后的结束页。

节 (section) : 在 vi 编辑器中, 跟在由 **sect=** 选项定义的小节标题后的文本。

节 (stanza) : 文件中一组行, 它们一起有一个公共功能或定义系统的一部分。节通常以空白行或冒号隔开, 且每个节都有名称。

截断 (truncate) ; 截取 : (1) 根据一些规则结束一个计算过程; 例如在特定项中结束幂级数求值。(2) 除去字符串的起始或结束元素。(3) 删除不能在指定或可用行宽度打印或显示的数据。对照叠加 (*fold*)。(4) 将字段或语句缩短至特定长度。

解块 (unblocked) : 在“企业系统连接导向器”中的一种属性, 当它设置了以后, 就为特定的端口建立通信能力。对照块 (*block*)。

解码 (decode) : (1) 通过逆向一些以前编码的效果来转换数据。(2) 解释代码。另见 *DCD*。

解码 (demangling) : 将编码后的 C++ 名称转换回它们最初的源代码名称。在编译期间, 诸如函数和静态类成员名这样的标识符用类型和分类信息进行编码, 以确保链接时的类型是安全的。这些编码后的名称将出现在目标文件和最终的可执行文件中。解码将把这些名称转换回它们最初名称, 从而使程序更容易调试。

解释例程 (interpreted routine) : 一个用于解码由伪代码写的指令, 并立即执行这些指令的例程。另见 *编译 (compile)*。

解释器 (interpreter) : 一种允许程序立即运行而不需要重新编译或重新连接的工具。

解锁 (dropping locks) : 释放事务在数据上保持的锁。

解析器例程 (resolver routine) : 一种内核进程, 用于将符号化的主机名称解析为因特网地址。一种例程方法, 用于根据本地网络是组织为平面网络还是分层网络来解析名称。

解析器 (parser) : 解释用户输入并确定如何处理输入的程序。另见语法规则 (*grammar rules*)。

解序列化 (deserialize) : (1) 从按位串行改为按字节并行。(2) 在 XDR 中用来从 XDR 格式改为具体的机器表示。

介质访问控制 (medium access control) : 在一个局域网上, 控制传输介质上的通信, 而不需要考虑介质的物理特征的协议。然而, 它要考虑网络拓扑方面, 来使得数据站之间的数据交换成为可能。另见 *逻辑链路控制 (logical link control)*。

禁用 (disable) : (1) 使其不起作用。在交互通信中断开与子系统的连接或停止子系统。对照启用 (*enable*)。(2) 使队列或连接到队列的设备离线, 从而使打印任务能够发送给它。

紧急切断电源警告 (EPOW) : 紧急切断电源警告 (Emergency power off warning)。

仅输入窗口 (Input Only window) : 在图形环境下, 一个用来控制诸如光标, 输入事件生成, 和抓取的不可见窗口。该窗口不能用于图形请求。

进程标识 (process ID, PID) : 指定给正在运行的一个进程的一个唯一号。

进程表 (process table) : 一种内核数据结构, 它包含关于系统中所有进程的相关信息。

进程并发 (process concurrency) : 一个给定进程一直有多个可分派线程的程度。

进程调步 (process pacing) : 请参阅 *调步 (pacing)*。

进程记帐 (process accounting) : 每个进程如何使用处理单元、内存和 I/O 资源的一个分析。

进程属性值 (process attribute value) : 在工作负载管理中, 进程属性值包含用户标识、组标识和应用程序路径名。

进程锁定 (process lock) : 允许调用进程同时锁定或解锁其文本和数据段到内存。

进程映像 (process image) : 请参阅 *新进程映像 (new-process image)*。

进程组 (process group) : 系统中的每个进程是由进程组标识标识的进程组的一个成员。此分组允许相关的进程组的信号化。新创建的进程加入到其创建程序所在的进程组。

进程 (process) : (1) 要产生期望结果需要的操作序列。(2) 一种实体, 它接收用于运行程序的处理器时间的部分。

(3) 一种系统内的活动，它由一个命令、shell 程序或另一个进程启动。当一个程序正在运行时，它被称为一个进程。
(4) 在计算机系统中，一种唯一的、有限的事件过程，它在给定条件下完成并由其自己的用途定义，或由其效果定义。
(5) 任何操作或数据上操作的组合。
(6) 在操作系统中，运行的程序的当前状态。这包含一个内存映象、程序数据、使用的变量、通用寄存器值、使用的打开文件状态和当前目录。在进程中运行的程序必须是操作系统程序或用户程序。另见作业 (*job*)。

进入呼叫信息包 (incoming-call packet)：在 X.25 通信中，指由 DCE 传送的呼叫管理信息包，以便向某个 DTE 通知另一个 DTE 请求的呼叫。另见包 (*packet*)；信息包；分组。

进入呼叫 (incoming call)：在 X.25 通信中，到达数据终端设备 (DTE) 的呼叫。

精度 (precision)：(1) 区分接近相等的值的能力的一种度量。另见单精度 (*single precision*) 和双精度 (*double precision*)。 (2) 说明一个量的辨别程度。例如，一个三位数字可以辨别 1000 个不同的数。 (3) 在 GL 中，打印或显示的位数。 (4) 用于近似一个齿条的一段直线段数。

精减指令集计算机 (Reduced Instruction Set Computer, RISC)：一种使用相对小的、简化了的在一个周期执行的常用指令集的计算机，以加速计算机的处理。

精细剪切 (fine clipping)：在 GL 中，精细剪切屏蔽所有绘制命令到屏幕上的矩形区域。除了字符串大小写，它可能是不必要。原来的字符串在转换后可能被粗心地剪切掉，或者对 3-D，剪切和字符串不绘制。通过视口粗剪切和使用屏幕掩码的精细的剪切，字符串可以很平滑移出屏幕到左侧或底端。另见剪切 (*clipping*) 和粗略剪切 (*gross clipping*)。

警报 (alarm)：工作站或打印机上的可听见信号，用来引起操作员的注意。

警告 (alert)：在 SNA 中，发送到主机系统上系统服务控制点 (SSCP) 的错误消息。

静态绑定 (static binding)：在编译时发生基于重载函数的解决方案的绑定。

静态变量 (static variable)：程序一开始运行就分配的变量，它保留已分配状态直到程序停止。标准作用域规则适用此变量。对照自动变量 (*automatic variable*)。

静态调试器陷阱 (static debugger trap, SDT)：置于代码中预先定义点的陷阱指令，它调用调试程序。陷阱指令引起运行时的程序检查，其结果就是激活调试程序。

静态链接 (static linking)：程序的链接，其中各库过程并入负载模块，而不是每次程序运行时从它们的库中动态装入。

静态路由 (static routing)：在主机、网络间或主机和网络间通过将路由由手工输入路由表设置路径的方法。静态路由不受路由守护程序的影响，且必须手工更新。

静态内存 (static memory)：具有固定大小的已分配的内存。

静态显示 (static display)：格式化文本过程中，当 **nroff** 命令在已指定为静态显示的输入文件中找到文本块时，只要有容纳整个块的足够空间，它就将文本放置于当前页面上。如果没有足够的空间，**nroff** 命令打开新的页面并将文本置于其上。另见浮点显示 (*floating display*)。

静态 (static)：(1) 创建弹出式窗口的一种风格。 (2) C++ 中用于定义变量和函数的作用域以及链接指令的关键词。对于内部变量，变量具有块作用域并在函数调用时保留其值。对于外部变量，变量具有文件作用域并在源文件中保留其值。对于类变量，变量由所有类对象共享且在整个程序中保留其值。

居中点 (centered dot)：作为标志使用的加粗分隔符，用来区分段落或列表项。

局域网 (local area network, LAN)：(1) 通信被限制在合适的地理区域 (1 到 10 km) 的一个网络，如：作为单一的办公楼、仓库、或者校园。一个本地网络服务于一个设施，不需要利用公共载体通信公司，尽管它们可能利用普通发送来进行相互连接。一个本地网络依赖于一个适合高数据速率的通信介质 (每秒 1 到 20 M 字节)，而且在操作过程中出错率一直很低。 (2) 在一个数据网络中的一系列传输用来在数据站之间直接数据通信。

矩形 (rectangle)：由 $[x,y,w,h]$ 指定的矩形，有无限地细轮廓路径，四个角为 $[x,y]$ $[x+w,y]$ $[x+w,y+h]$ 和 $[x,y+h]$ 。在 XGSL 中，当填充一个矩形时，右下边缘不被绘制。例如，如果 $w=h=0$ ，则不绘制任何东西；如果 $w=h=1$ ，则将绘制一个像素。

矩阵堆栈 (matrix stack)：在 GL 中，一个带有硬件和软件支持的矩阵堆栈。该堆栈的顶部的矩阵是当前的转换矩阵，所有通过图形流水线的点都乘以该矩阵。它是当前模型和视图转换的串联 (*concatenation*)。另见当前转换矩阵 (*current transformation matrix*)。

矩阵 (matrix)：(1) 一个按行和列排列的矩形元素数组，可以基于矩阵代数规则进行操作。 (2) 在计算机中，输入输出为阵列格式的神经网络，该网络以在某些交集上连接的元素开头。 (3) 广义上讲，指一个任意维的数组。

句柄 (handle): 作为某个对象的临时局部标识符的数据结构。一个句柄可通过分配而创建。可以通过绑定句柄而使其识别特定位置的对象。

聚集 (aggregate): (1) 数组、结构或联合。(2) 在编程语言中, 某一数据类型的数据对象的结构化集合。(3) 一种发射载波信号, 包含在传输电路上发送的 12 个单边带。

聚集 (gather): 对于输入/输出操作, 从非连续内存位置读取数据以写入设备。对照散射 (*scatter*)。

拒绝 (reject.): 根据测试期间的结果, 使已应用更新的部分不成为产品的永久部分。当您拒绝某个应用服务更新时, 该更新文件被删除, 并且软件重要产品数据 (SWVPD) 信息被更改以表明该更新不再在系统上。如果有软件的前一个版本, 则它被恢复并成为软件的活动版本。对照应用 (*apply*) 和提交 (*commit*)。

巨量消失 (catastrophic cancellation): 一种编程错误, 其中会一直加上或乘以非常大的负指数值, 直到产生零值, 然后该值传给后续计算。

具体类 (concrete class): 允许实例的 OM 类。

句子 (sentence): 在 vi 编辑器中, 指通过 . (句号)、! (感叹号) 或 ? (问号) 与其它文本分隔开的文本, 每个标点后紧跟两个空格。

卷标识 (Volume ID, Vol ID): 记录在软盘上一系列字符, 用于为用户和系统标识软盘。

卷标 (volume label): 磁带或磁盘上用于标识磁带卷及其所有者的区域。

卷服务 (Volume Service, VOL): 寻址存储器的 Encina Base 的组成部分。

卷组 (volume group, VG): 一个或多个物理卷的集合, 从此集合空间可分配给一个或多个逻辑卷。具有不同大小和类型的 1 到 32 个物理卷 (读/写固定磁盘驱动器) 的集合。另见逻辑卷 (*logical volume*)。

卷 (volume); 音量: (1) 数据某部分, 它和数据载体一起可用作为一个单元方便地处理。(2) 系统的音响级别。(3) 文件系统物理存储位置。另见记录卷 (*log volume*)。

绝对地址 (absolute address): 无需额外求值就可确定存储位置或设备的地址。

绝对设备 (absolute device): 一种定位设备 (如平板), 以坐标系统上数集的形式向操作系统报告其位置。

绝对时间 (absolute time): 时间刻度上的一点。

绝对值 (absolute value): 实数的数值, 不管其代数符号 (正号或负号)。

[K]

卡 (card): 插入系统部件插槽的电子线路板。另见适配器 (*adapter*)。

开放系统互连 (Open Systems Interconnection, OSI): (1) 开放系统的互连与特定 ISO 标准一致。(2) 标准化程序的使用启用数据处理系统的互连。

开关表 (switch table): 文件系统用来定位字符设备入口点的表。

开关 (switch): 一个命令行选项。

可编程输入/输出操作 (programmable input/output operation): 一种数据传送, 作为 I/O 指令的部分在处理器和一个 I/O 设备或内存地址空间间进行。I/O 指令指定控制逻辑的地址, 将执行的一个命令, 和传送数据至的处理器寄存器位置或从中传送数据的处理器寄存器位置。

可编程终端 (programmable terminal): (1) 有计算能力的一种用户工作站。(2) 一种工作站, 它可被编程以执行用户确定的功能。

可编程字符集 (programmable character set, PCS): 一种几何文本字体。与笔划文本 (*stroke text*) 同义。另见几何文本 (*geometric text*)。

可辨认的编码方式 (distinguished encoding): 对“基本编码规则” (Basic Encoding Rules) 的限制以确保对每个 ASN.1 值有唯一的编码方式, 这定义在“X.500 目录标准” (Directory Standards, CCITT X.509) 中。

可擦可编程只读存储器 (EPROM): 可擦可编程只读存储器 (Erasable programmable read-only memory)。

可重定位 (relocatable): (1) 当程序重定位时不需要更改的值、表达式或地址。(2) 一组代码的属性, 其地址常数的改变可以补偿起始点的变化。

可重入服务 (reentrant service): 一种服务, 它对从并行多个线程中调用是安全的。如果服务是可重入的, 则调用例程时没有负担来序列化其访问或进行其它显式防范。另见线程序列服务 (*thread-serial service*) 和线程同步服务 (*thread-synchronous service*)。

可串行性 (serializability): 事务处理系统的基本属性, 这指一种概念, 即通过事务进行的交换和修改必须能同步, 并使其看起来多个同时进行的事务实际上是一系列顺序请求。数据被事务更改, 或事务依赖的对象必须在第一个事务完成之前与其它事务隔离开来。

可访问的 (accessible): 指客户机拥有对一个对象的有效指定者或句柄。

可恢复的数据 (recoverable data)： 其值经历系统关机和故障都能存留下来的数据。对可恢复的数据进行的更改是永久的，与系统问题无关。对可恢复的数据的记录更改是用于确保永久性的最常用方法。对记录在日志中的数据所做的更改总可被重播，以使该数据成为有效状态。

可绘制的 (drawable)： 在“增强的 X-Windows”，对于窗口和象素映射的一种集合术语，在图形操作中作为目的地。然而，只输入窗口不能在图形操作中作为可绘制的源或目的地。

可见部分 (visible part)： 对于 Ada 编程，请参阅软件包 (package)。

可见的 (visible)： (1) 被映射且未被屏幕上的另一个窗口遮挡的窗口区域。(2) 基于作用域规则并独立于存取的 C++ 标识符的可见性。

可见性 (visibility)： Ada 语言中，在程序文本中给定点上，具有特定标识符的实体的声明，如果此实体对于在标识符点上的具体值是可接受含义，则它是可见的。在选定部件的选择器位置或命名的关联的名称位置，声明可见于选择。否则，声明直接可见，也就是说，如果仅标识符就具有该含义。

可靠流传递 (reliable stream delivery)： 一种包传递，它允许一个机器上的一个应用程序连接到另一个机器上的一个应用程序。流实际包含许多数据的包，它被一次发送一个到接收机器。

可链接的 (linkable)： 当所有编译单元相关性被解析时，一个 Ada 语言程序的状态。要产生一个可执行文件，编译器不需要翻译任何 Ada 源；仅仅需要调用链接编辑器。

可伸缩性 (scalability)： 工作负荷从多处理器环境中获益的能力。

可识别专用操作代理 (recognized private operating agency, RPOA)： 一种专用 X.25 网络，可由用户在调用设置时间可选地选定来传递 X.25 流量。

可视的 (viewable)： 指所有祖先都已经映射过的映射窗口；不必可见。当图形不可视时，可以在窗口上执行图形请求，但是除非服务器保持后备存储，才保留输出。

可推模块 (pushable module)： 流头和驱动程序之间的一种模块。驱动程序是一种非可推模块，流头包含非可推模块。

可信环境 (trusted environment)： 一种清洁环境，其中为了确保用户和操作系统之间的安全性，所有不可信的过程都被杀死。

可信计算基 (trusted computing base, TCB)： 系统中包含所有支持该系统安全性策略的元素的部分。此可信计算基础包含所有保护系统上信息的硬件、微码和软件。

可信进程 (trusted process)： 满足安全性特定标准的进程。

可信路径 (trusted path)： 请参阅可信环境 (trusted environment)。

可信设备 (trusted device)： 一种 IPL 设备，比如固定磁盘或软盘驱动器，其中 IPL ROM 代码可以找到 IPL 记录和 IPL 代码。

可信同级 (trust peer)： 一个单元相对于另一个单元的特征，通过该特征，此单元保留了共同的认证代理。

可选的设施 (optional facilities)： 在 X.25 通信中，网络供应商可能或可能不提供设施给那些是否选择订阅的客户。另见闭合用户组 (closed user group)、快速选择 (fast select)、逆向收费 (reverse charging) 和吞吐量级别协定 (throughput-class negotiation)。

可选软件 (optional software)： 也可参考可选软件产品 (optional software products)。当安装基本操作系统 (BOS) 时，软件是不自动地安装到系统上。可选软件可以是封装的产品和 BOS 一起销售。也可以单独购买可选软件，软件产品单独的订购而不作为 BOS 的一部分销售。在任一情况中，在可以安装可选软件前必须把 BOS 安装到系统上。

可移动存储设备 (removable storage device)： 系统配置成为一个系统 DASD 的可选部件期间定义的任何存储设备。一种可移动存储设备，它可在正常操作期间的任何时候从系统中卸下。

可移植文件名称符集 (portable file name character set)： 在 XPG4 系统接口中，用其构造可移植文件名的字符集。要使文件名可跨 XPG4 和 ISO POSIX-1 标准实现可移植，它必须仅由以下字符组成：

ABCDEFGHIJKLMNOPQRSTUVWXYZ

abcdefghijklmnopqrstuvwxyz

0123456789._-

最后三个字符分别是句号、下划线和连字符。连字符不能用作可移植文件名的第一个字符。大小写字母在顺从实现之间保留它们的唯一身份。在可移植路径名中，还可使用斜杠字符。

可移植性 (portability)： 一种特征，它确定源程序是否可被编译和是否可不需要重新编码运行在不同体系结构的计算机上。

可移植字符集 (portable character set): 在 XPG4 系统接口中, XSI 一致系统支持的所有语言环境中存在的字符集合是: Or ABCDEFGHIJKLMNOPQRSTUVWXYZ

abcdefghijklmnopqrstuvwxyz

0123456789!#%&*()_+~;`<>?.\|/@\$

还包含警告、退格、制表符、新行、垂直制表符、换页、回车、空格字符和空字符 NUL。

可用空间 (free space): 在按键排序的数据集或文件的控制间隔内保留的空间, 用来将新记录按键顺序插入到数据集或文件中; 也指为相同目的而在控制区域保留的控制间隔。分布式可用空间 (*distributed free space*) 的同义词。

可用状态 (available state): 设备配置好时所处的状态。ODM 中的“定制设备对象类”中的设备状态字段反映一个设备是否处于可用状态。

可执行程序 (executable program): 能作为独立的过程运行的程序。它包含一个主程序以及可选的一个或多个子程序。

可执行 (文件) (executable): 一个可以加载到内存并作为程序执行的文件。一个可执行文件是由绑定程序 (**ld**) 从一个或多个目标 (**.o**) 文件产生的。编译命令的缺省处理包括调用绑定程序来产生一个名为 **a.out** 的可执行文件。

可执行文件 (executable file): 一个包含对采取的行动执行操作的程序或命令的文件。

可执行语句 (executable statement): 能让程序执行某个操作的语句。例如, 计算、测试条件或改变正常的顺序执行。

客户机程序 (client program): 一个使用 C++ 类的程序。程序称为类客户机。

客户机代理程序 (client agent): 请参阅位置程序代理客户机代理程序 (*Location Broker Client Agent*)。

客户端高速缓存 (client-side caching): 一个高速缓冲存储区, 包含频繁访问与客户机应用程序相关联的信息。客户端高速缓存的主要目的是减少对关键信息的存取时间。

客户机应用程序 (client application): 一类应用程序。另见客户机 (*client*) 和应用程序 (*application*)。

客户机 (client): (1) 分布式文件系统环境中, 一个依赖于服务器为其提供程序或程序访问的系统。(2) “增强的 X-Windows” 中一个通过进程间通信 (IPC) 路径 (如传输控制协议 (TCP) 连接或共享内存缓冲区) 连接到服务器“增强的 X-Windows” 的应用程序。程序可以称为服务器的客户机, 但它实际上就是 IPC 路径本身。协议视具有

多个打开到服务器的路径的程序为多客户机。另见进程间通信 (*inter-process communication*)。(3) “增强的 X-Windows” 中, 一个使用应用程序中小窗口或用来组成另一个小窗口的工具箱例程。(4) AIXwindows 中, 一个在传统客户机/服务器模型 (基于“增强的 X-Windows” 和 AIXwindows) 中担负客户机角色的软件应用程序。另见客户机应用程序 (*client application*)。(5) NCS 中一个使用接口创建远程过程调用 (RPC) 的程序。

克隆设备 (clone device): 一种流 (STREAMS) 设备, 初始打开时返回一个未使用的主和次设备, 而不需要在打开调用中以名称指定次设备。

空白公共区 (blank common): FORTRAN 里的一个未命名公共块。

空白区域 (white space): 间隔字符、制表符以及换行字符。

空间 (space); **空格**; **空号 (脉冲)**: (1) 用于存储数据的站点, 比如存储介质中的位置。(2) 基本区域单元, 通常是单个字符的大小。(3) 一个或多个空格字符。(4) 中性电路中, 引起环路打开或引起信号丢失的脉冲。在极性电路中, 它导致环路电流流向与标记脉冲相反的方向。空号脉冲等于二进制零。

空闲列表 (idle list): 网络上辅站的列表, 由于其不活动, 通常由主站轮询得较少。

空闲时间 (idle time): 可操作时间的一部分, 在该时间中, 某一功能单元不运行。

空信号 (null signal): 0 (零) 信号参数。

空语句 (null statement): 由分号构成的语句。

空字符串 (null character string): 指定由字符组成的字符串的两个单引号。

空字符 (null character, NUL): (1) hex 00 字符用于表示不用打印或显示的字符。(2) 一种控制字符, 它用于完成介质填充或时间填充, 可以插入到字符序列中或从字符序列中除去而不影响序列的意思。(3) 用于字符串定界的控制字符。

控件 (control): CDE 中用于多种元素 (如执行操作或指示选项设置的按钮、复选框和滚动条) 的通用术语。请参阅前面板控件 (*Front Panel control*)。

控制程序 (control program, CP): 操作系统的一部分, 确定基本函数的执行顺序。

控制单元终端方式 (control unit terminal, CUT mode): 一种用于与 3174/3274 控制器或其它相应接口单元通信的协议。在该协议中, 工作站里的程序为用户仿真 3278/79 终端, 接口单元负责实施协议。

控制点概要文件名称 (control point profile name)： 定义与附件相关联的物理单元的节点标识的控制点概要文件名称。

控制点 (control points)： 计算机图形学中，在实数空间里控制样条曲线形状的点。系统为线框有理三次样条 (wire frame rational cubic splines) 提供硬件支持，NURBS 表面的规范需要四个控制点。

控制键 (control key)： (1) 标有 Ctrl 标签的键盘键。(2) 通过按 Ctrl 键然后按键盘上另一个键生成的组合键，执行某些功能或产生特殊字符。

控制块 (control block)： 一个程序用来保留控制信息的存储区域。

控制路径 (control path)： 一组线路、硬件和控制准则，确定某个特定 TTY 的当前特征。

控制命令 (control commands)： 在 shell 步骤中允许条件或循环逻辑流 (flow)。

控制台报警符 (console bell)： 与 BEL 同义。

控制台设备 (console device)： 基本操作系统 (BOS) 安装期间，系统控制台是位于正在安装软件的系统上的显示设备。

控制台显示器 (console display)： 系统控制台上的显示器，操作员可在其上显示、发送及应答消息，还可使用所有控制命令。

控制台 (console)： 主操作系统显示站。与系统控制台 (system console) 同义。

控制语句 (control statement)： 编程语言中用来改变语句继续顺序执行的语句。控制语句可以为条件语句或命令语句。

控制站 (control station)： 多点线上的首要或控制计算机。控制站控制数据的发送和接收。

控制终端 (controlling terminal)： 该术语指一台活动工作站，已授权的用户可在其上输入影响系统操作的命令。任何进程的控制终端通常是活动工作站，该进程的进程组从活动工作站启动。一个工作站只能有一个控制进程组，一个进程组只能有一台的控制终端。控制进程组从控制终端接收某种中断信号。

控制状态 (control state)： 代表当前网络安装管理 (NIM) 操作正在机器上执行的状态。该状态是两个机器状态之一。

控制字符 (control character)： (1) 一种不是图形字符 (如字母、数字或标点符号) 的字符。这些字符称为控制字符，因为它们频繁地用来控制外围设备。回车

(RETURN) 和换页 (FORM-FEED) 是用来控制工作站或打印机的控制字符。与非打印字符 (*nonprinting character*)。(2) 键盘上的 Ctrl 键。(3) 出现在特殊上下文中的字符，启动、修改或停止影响记录、处理、传输或数据解释 (例如回车、字体更改和结束传输) 的任何操作。(4) 在文本文件中执行格式编排功能的非打印字符。

库单元 (library unit)： 在 Ada 语言中，五个句法实体之一：一个子程序声明、一个包声明、一个类属声明、一个类属实例或在没有对应的子程序声明的一个子程序体。正如名称所暗示的，一个库单元驻留在 Ada 程序库中。库单元的意义是它们可以被其它独立编译单元所引用。这个引用可以是显式的 (由一个 with 子句引用或隐式的 (比如一个包体到它的说明的引用))。

库列表文件 (library list file)： 在 Ada 语言中，一个包含一个或更多子库名称的文本文件，这些子库组成一个 Ada 程序库。

库组件 (library component)： 在 Ada 语言中，一个包主体、包的规范说明、子程序体、子程序说明或驻留在库中的链接对象模块。

库 (library)： (1) 一个函数、调用、子例程或其他数据的集合。(2) 一个包含大量单个文件的副本和允许它们被单独访问的控制信息的数据文件。(3) 在 Ada 语言库管理中，一个存储由编译器产生的各种中间代码文件、记录依赖性和 Ada 语言规范要求的编译信息次序的数据库。当编译一个依赖于其他 (以前编译过的) 单元的单元时，所要求的依赖性信息 (比如一个单元的包说明，该包以与一个 with 子句包含) 从该数据库中获得。类似的，当一个 Ada 程序单元被链接时，该库指定创建可执行映像必须包含的单元的集合。

夸克 (quark)： 在“增强的 X-Windows”中，字符串 (*string*) 的同义词。

跨距 (stride)： 内存中数组元素的布局 and 访问那些元素的顺序之间的关系。跨距 1 表示邻近内存的数组元素受到数组处理循环的连续迭代的访问。跨距 N 表示对于每个被访问的数组元素，表示在下个被访问元素之前，略过 N-1 邻近内存的元素。

块设备 (block device)： (1) 文件系统中的一种文件类型，以一个 i-node 描述。(2) 通过设备驱动程序的方法访问的设备。

块输入/输出通信区 (BIOCA) (block input/output communication area, BIOCA)： 一内核地址空间的存储块，用来与块 I/O 子系统通信。

块数据子程序 (block data subprogram)： 在 FORTRAN 中以语句 BLOCK DATA 开头的子程序，用来初始化已命名公共块中的变量。

快速选择 (fast select)： 在 X.25 通信中，一个允许包括调用请求和清除请求信息包数据可选的设施。另见可选的设施 (*optional facilities*)。

快速页码 (expressed folio)： 一种页码编排样式，其中每页（可能不包括第一页）都编号。另见页码 (*folio*)、盲页码 (*blind folio*) 和放下页码 (*dropped folio*)。

块特殊文件 (block special file)： 块设备特殊文件。该文件提供对使用内核缓冲区的输入或输出设备的访问，并能支持文件系统。另见字符特殊文件 (*character special file*)。

块文件 (block file)： 列出磁盘上块使用状况清单的文件。另见特殊文件 (*special file*) 和字符特殊文件 (*character special file*)。

块语句 (block statement)： 出现在符号 { (左大括号) 和 } (右大括号) 之间的任意数量的数据定义、声明和语句。例如，C 语言中把一个块语句作为单个 C 语言语句读入。Ada 语言中的块语句是可能包含语句序列的单个语句。它也可能包含声明部分和异常处理程序；它们的作用域为本地块语句。另见语句 (*statement*)。

块 I/O (block I/O)： 存储在随机位置的数据块的输入 / 输出操作。

块 (block)： (1) 一组相邻的记录，作为一个单元记录或处理。块以块间隔分隔，每个块可以包含一个或多个记录。(2) 在数据通信中，一组记录作为一个单元进行记录、处理或发送。(3) 在编程语言中的一种复合语句，符合至少声明之一包含其中的作用域。一个块也可以为其它目的而指定存储器分配或分段程序。

宽带通道 (broadband channel)： — 6 MHz 宽的数据传输通道。与宽带通道 (*wide band channel*)。

宽带通道 (wide band channel)： 比音频通道具有更宽带宽的通信通道，所以，它能够高速传输数据。宽带通道 (*broadband channel*) 的同义词。

宽带 (broadband)： 一种传输介质和技术，使用宽频率范围，分成较窄频率的子波段，以便不同类的传输可以同时进行。

框 (box)： 围绕文本或表的线。

框 (sash)： 在 CDE 中，指能使您使用鼠标增加或减小窗口窗格大小的分隔符上的框或分割条。您可以使用键盘浏览至此框。

捆绑 (bundle)： 可用于安装的软件产品集合。

括弧读取锁 (bracket read lock)： 获得的共享锁，仅在当前操作的持续时间里保留，然后立即删除。这种锁定方

式确保读取的仅是提交的数据。这不确保后续读取的数据还是相同的值，因为读取操作完成且删除锁之后，其它事务可以立即修改数据值。

括弧 (bracket)： (1) ASCII 字符 [(左括弧) 和] (右括弧)，也称为“方括弧”。(该用法与“括弧”的英国用法不一样，英国用法作圆括号用，也就是扩起这个句子的字符。)(2) 在 SNA 中，一个或多个请求单元链和它们的响应，在两个 LU-LU 会话间交换并表示它们之间的一个事务。在另一个括弧可以开始之前，(前面的)括弧必须是完整的。括弧示例：数据库查询和应答、更新事务和到工作站的远程作业项 (remote-job-entry) 输出序列。

扩展地址 (extended address)： 与扩展地址 (*extended address*) 同义。

扩展二进制编码的十进制交换代码 (extended binary-coded decimal interchange code, EBCDIC)： 为文本数据的表示开发的代码。EBCDIC 由一组 256 八位字符组成。

扩展服务 (Extended Services)： 一组可选择地安装的操作系统函数和程序。

扩展公共目标文件格式 (extended common object file format, XCOFF)： 操作系统第 3 版目标文件格式。XCOFF 把标准公共目标文件格式 (COFF) 和 TOC 模块格式概念结合起来，提供在一个目标文件内动态链接和置换单元。

扩展结果 (extended result)： 一个异常通知，其数据定义在 `dlc_getx_arg` 结构 `result_ext` 字段。

扩展接口 (extended interface)： 提供全函数系统调用的集合 (`readx` 和 `writex` 系统调用) 以同 SNA Server 进行通信。这些调用在调用时包含额外参数 (指向结构的指针包含额外函数请求)。另见接口 (*interface*) 和受限接口 (*limited interface*)。

扩展精度常量 (extended precision constant)： 处理器的实数近似值，它占有 16 个连续存储字节，并可以假定为正值、负值或零。精度大于双精度型类型的精度。

扩展名 (expanded name)： Ada 语言中，代表一个在紧接着的构造中声明的实体。一个扩展名是由选定的组成部分组成：前缀代表该构造 (一个程序单元或一个块、循环或 `accept` 语句)；选择子是实体的简单名称。

扩展名 (extension)： 在“增强的 X-Windows”中，要扩展系统，已命名的扩展可以为核心协议定义，它包含到输出请求、资源和事件类型扩展。

扩展系列 (extended family)： 全部 Ada 语言编译单元支持程序和它们所有系列的集合，包含关联的主体和子单元。单元的支持程序只包含它编译需要的库单元，不包含关联主体和子单元。

扩展许可 (extended permission)： 一种访问方式，它为指定的个人或组修改文件的基本许可权。一种扩展许可权可以拒绝或者准许一种访问方式。另见任意访问控制 (*discretionary access control*)。

扩展选择 (extended selection)： 在 CDE 中，通过增加选择技术添加多个项到选定的集合。例如，可以通过移动指针到一新的项并同时按下“控制”和“选择”鼠标按钮，添加一选项。

扩展字符 (extended character)： 除了 7 位 ASCII 字符以外的字符。扩展字符可以有八位集 (128 到 255 序数) 的 1 字节代码点。另见代码页 (*code page*) 和代码点 (*code point*)。

扩展字体 (extended font)： (1) 字符宽度宽于它相应正常字体的一种字体。(2) 字体部分的集合，用来支持一次使用多于 256 个图形字符的语言。

扩展 curses (extended curses)： 系统库 (*libcurses.c* 库)，它包含向终端写数据和从终端屏幕获取数据的控制功能。它支持颜色、多窗口和增强的字符集。

[L]

垃圾箱 (Trash Can)： CDE 中容纳已删除文件或文件夹的容器。

类层 (class tier)： 一个指定哪个类是最重要的工作负荷管理 (Workload Management) 类层值。如果不使用层值，所有类都同等重要。所有具有较高层类的资源限制 (包含资源目标) 比那些处于任何较低层类更为有利。

类成员运算符 (class member operators)： 用于通过类对象或指向类对象的指针来访问 C++ 类成员。它们是 `..`、`->`、`.*` 和 `->*`。

类赋值规则 (class assignment rule)： 表示什么样的进程属性值集决定进程赋值给哪个类。

类记录 (class record)： 一个包含属于任何给定小窗口类的数据对象的特殊小窗口记录。另见记录 (*record*)、小窗口记录 (*widget record*) 和类 (*class*)。

类键 (class key)： C++ 关键字之一：`class`、`struct` 和 `union`。

类库 (class library)： C++ 类集合。

类名 (class name)： (1) 对于小窗口和小配件，代码中的名称对应于资源数据库，包含类中所有对象的一般属性。(2) 一个 C++ 类类型的唯一标识符，成为其作用域内的保留字。

类模板 (class template)： 一个描述如何构造一组相关 C++ 类集的蓝图。

类属单元 (generic unit)： 在 Ada 语言中，用于一组子程序或一组软件包的模板。使用模板创建的一个子程序或软件包称为类属单元的一个实例。类属实例化是创建实例的那种声明。一个类属单元作一个子程序或软件包编写，但其前面有类属形式部分的说明，该形式部分可以声明类属形式参数。一个类属形式参数是一种类型、一个子程序或一个对象。一个类属单元是程序单元中的一种。另见实例 (*instance*)。

类属接口 (generic interface)： 以独立于任何特殊编程语言的级别而定义的接口。

类属数据链路控制 (generic data link control, GDLC)： 一种类属接口定义，向应用程序和内核用户提供公共命令集合，以控制操作系统中的 DLC 设备管理器。对所有 DLC 管理器指定了入口点定义、提供的功能以及数据结构的要求。以下 DLC 符合此接口：IEEE 802.3 以太网、标准以太网、SDLC 和令牌环。

类属网络 (generic network)： 一种“网络安装管理” (NIM) 网络类型，用于定义 NIM 当前未知的网络，以支持网络引导操作。

类型标识符 (type identifier)： 定给说明类型的名称。另见类型说明符 (*type specifier*)。

类型重组 (type regrouping)： 涉及重新排列混合类型的表达式以使给定类型的所有变量可以组合在一起的优化。

类型定义 (type definition)： 数据类型名称的定义。

类型兼容性 (type compatibility)： 请参阅可兼容类型 (*compatible types*)。

类型声明 (type declaration)： 类型以及规范语句中变量或函数的长度 (可选) 的规范。

类型说明符 (type specifier)： 数据类型的名称。另见类型标识符 (*type identifier*)。

类型转换 (type conversion)： 使应用程序将指定数据字符串从一种说明类型更改为另一种的例程或例程集合。在 AIXwindows 编程中，通过使用 MRM 数据库中包含信息的转换，在字符串上执行类型转换。

类型 UUID： (1) 永久性标识特定类型的 UUID。RPC 运行时库和“位置代理”都使用 UUID 来指定类型。(2)

DCE RPC 中, 标识对象和相关管理器的特定类型的 UUID。另见对象 (*object*) 和通用唯一标识符 UUID (*Universal Unique Identifier UUID*)。

类型 (type)；输入：(1) “增强的 X-Windows” 中用于标识数据的任意基本单位。一种类型仅处于客户机的利益, 服务器不解释。“增强的 X-Windows” 为许多经常使用的类型预先定义了类型基本单元。客户机也能定义新类型。(2) 在 Pascal 中, 请参阅数据类型 (*data type*)。(3) NCS 中, 指一类对象。可以通过同个接口或多个接口访问某特定类型的所有对象。(4) Ada 语言中, 一种类型表现了一组值和一组可用适用于该值的操作的特征。类型定义是定义此类型的语言构造。特定的类型有存取类型、数组类型、专用类型、记录类型、标量类型或任务类型。(5) XOM 中的一个范畴, 其中属性值基于它们的目的而定制。

类作用域 (class scope)：C++ 类成员作用域。

类 (class)：(1) 属于设备的 I/O 特征。操作系统中设备分为块或字符设备。(2) 在“增强的 X-Windows” 中, 一个特定对象所属的常规组。另见小窗口类 (*widget class*) 和类记录 (*class record*)。(3) 在 AIXwindows 中, 一面向对象的数据结构, 包含有关一组相似图形对象 (小窗口或小配件) 的常规信息。每个图形对象类继承对象层次结构中上层类的一些或全部外观特征和行为特征。(4) C++ 类是一种用户定义的数据类型。类数据类型可以同时包含数据表示 (数据成员) 和函数 (成员函数)。另见对象类 (*object class*)。(5) 在工作负荷管理 (Workload Management) 中, 一个具有单一资源限制值集和应用于它们的目标共享的进程集合 (及与它们关联的线程)。

离散访问控制 (discretionary access control)：一种安全机制, 它防止信息通过所有者控制的文件访问导致的未授权的泄漏或修改。另见访问控制列表 (*access control list*)、基本许可 (*base permission*) 以及扩展许可 (*extended permission*)。

离散类型 (discrete type)：在 Ada 语言中具有不同值的有序集合的类型。离散类型为枚举类型和整型。离散类型用于索引和迭代, 以及 case 语句和记录变量中的选项。

例程 (routine)：程序中的一组语句, 它引起系统执行一个操作或一系列相关的操作。另见宏 (*macro*) 和子例程 (*subroutine*)。

粒度 (granularity)：较大实体细分的程度。例如, 一码分成英寸比一码分成英尺具有较细的粒度。

立即超对象 (immediate superobject)：其属性值中包含另一个对象的对象。

立即超类 (immediate superclass)：一个 C 类的超类, 不具有任何其自身是 C 超类的子类。

立即方式 (immediate mode)：在 GL 中, 采用这种方式, 图形命令被立即执行而不是编译成图形列表。

立即数据 (immediate data)：(1) 在汇编程序语言中, 指令中出现的实际数据, 与某数据的符号名称相对。数据可立即从指令中获得, 因此不必从内存读取。(2) 指令运行时传送的数据。

立即子对象 (immediate subobject)：作为另一个对象的属性值的对象。

立即子类 (immediate subclass)：一个 C 类的子类, 不具有任何其自身是 C 子类的超类。

历史记录 (history)：最近输入的 Ada 调试器命令列表。历史记录命令的缺省列表为 20 个最近输入的命令。可以使用历史记录重新输入先前的命令, 或者通过编辑历史记录引用文本形成一个新命令。

立体 (frustum)：指 GL 中截平的四面棱锥; 即头部截断的棱锥。在透视投影中, 切割的卷的形状为截锥体。截锥体底部称为远切割平面, 截锥体顶部是近切割平面, 侧面分别是顶切割平面、左切割平面、底切割平面和右切割平面。在正交投影中, 切割的体是一个平行六面体。另见切割窗格 (*clipping pane*)。

联合标记 (union tag)：命名联合数据类型的标识。

联合 (union)：可以拥有多种类型中任何一个的变量, 但是每次只能有一种数据类型。

联机 (online)：(1) 直接受控或直接用计算机通信。对照脱机 (*offline*)。(2) 当在计算机的直接控制下时的功能单元的操作。

连接标识 (connection identifier, CID)：(1) 一个用于识别资源的值。该值在连接处理建立会话之后返回给连接程序, 并且必须用于对资源的后继请求。(2) X.25 API 中用于标识已经建立或已经接收到的呼叫的名称。

连接单元 (connected unit)：FORTRAN 中通过 OPEN、READ 或 WRITE 语句连接到文件的单元。

连接方式 (connection mode)：一种面向电路的传输方式, 其中数据在已建立的连接中以顺序方式从一个用户传递到另一个用户。

连接概要文件 (attachment profile)：包含与其它具有到网络的 LU 连接的已定义概要文件相关联的参数。这些参数还定义使用的网络类型。

连接概要文件 (connection profile)：一个数据管理文件, 包含将其它已定义概要文件与两个逻辑单元的连接相关联的参数。

连接管理流 (connection management stream) : X.25 中的特殊流, 接收所有指向 DLSAP 地址的输入连接指示, DLSAP 地址不绑定任何其它与特定 PPA 关联的流。

连接建立 (connection establishment) : 连接方式中的阶段, 使两个数据链路服务 (DLS) 用户之间能够创建数据链路连接。

连接类别 (attachment class) : 连接类别指定工作站连接到 FDDI 网络的方式。工作站可以为双连接站 (DAS) 或单连接站 (SAS)。

连接类型 (connection type) : 这是“预定义连接对象类”中的一个字段。对于中间设备, 连接类型标识可以与它连接的设备子类。

连接密钥 (connection key) : 标识可连接到指定位置的中间设备的设备子类。

连接器 (connector) ; 连接符: (1) 一个用来连接两个其它电气部件的电气部件。(2) 一个流程图符号, 代表流程线中的一个中断并指出流程线从何处继续。(3) 一种建立电气流程的方法。

连接时间记帐 (connect-time accounting) : 每个用户登录到系统所花时间的记录。

连接数据集到线路 (connect data set to line, CDSTL) : SNA 中, 一个确定数据终端就绪 (DTR) 信号到调制解调器是如何操作的选项。如果 DTR 表明一个从 DTE (数据终端设备) 到连接的 DCE (数据电路终端设备) 的无条件命令以将自身连接至网络或从网络删除时使用。

连接位置 (connection location) : 标识中间设备上的特定位置, 在此处可以连接子设备。

连接性 (connectivity) : 一种确定位于不同网络中的两台机器能否通信的算法。如果机器能够通信, 连接性还确定应该使用哪个主机名及必须添加哪种 TCP/IP 路由信息。

连接 (attachment) : (1) 到使之工作的网络的物理连接。(2) 一种控制 CP、逻辑链路控制和物理链路控制的资源类型。(3) 在 CDE 的邮件程序中, 电子邮件消息中的数据对象, 以图标显示在附件列表中。附件可以是文本、声音或图形。多个消息可以添加 (附加) 到单个电子邮件消息。

连接 (concatenate) : (1) 连接到一起。(2) 连接两个字符串。

连接 (concatenation) : (1) 连接到一起。(2) 连接两个字符串。(3) 在 GL 中, 将一系列几何变换 (如旋转、平移和缩放) 组合在一起。变换连接对应于矩阵相乘。

连接 (connect) : X.25 通信中连接到 X.25 网络的端口。

连接 (connection) : (1) SNA 中将处于不同节点上的两个 LU 链接在一起的网络路径, 以便为运行在各自 LU 上的应用程序在 LU 间提供通信信道。(2) X.25 通信中两个数据终端设备 (DTE) 间的现有虚电路。交换式虚电路 (SVC) 连接用于呼叫持续时间; 永久虚电路 (PVC) 连接是两个 DTE 间的永久连接。(3) “增强的 X-Windows”中服务器和客户机程序之间的 IPC 路径。客户机程序通常 (但不是必需) 有一个到服务器的连接, 通过该连接发送请求和事件。(4) 系统通信中的通信链路, 通过它在两个系统或一个系统和一个设备之间传递数据。

联系端口 (contact port) : 与众所周知的端口 (well-known port) 同义

联系 (set) : 在 NCS 中, 将分配的远程过程调用 (RPC) 句柄与特定套接字地址相关联。另见绑定 (bind)。

连续行 (continuation line) : 源代码语句的一行, 当源代码语句无法包含在前一行时向该行输入。

连字 (ligature) : 打印到一起, 而连接起来的两个 (或有时候更多个) 字符。

链接编辑器 (linkage editor) : 一个解析在各分散的编译的对象模块之间的交叉引用, 然后指定最终地址来创建一个单一的可再次定位的装入模块的程序。如果一个单一的对象模块被链接了, 链接编辑器就会使它可以再次定位。

链接编辑 (link-editing) : 通过一个链接编辑器创建一个可加载的计算机程序。

链接程序 (linker) : 请参阅链接编辑器 (linkage editor)。

链接地址 (link address) : 初始化所指定的一个地址, 它标识了一个通道或者一个控制单元, 允许发送和接收帧, 并且执行 I/O 操作。一组共享网络的计算机, 不包含广域网链接的网桥。

链接锚 (link anchor) : 给出了一个特定链接的位置的参考点。

链接描述符 (link descriptor) : 在 ODM 中, 类型 link 一个命名的变量, 它用来定义对象类中的一个对象和另一个对象类中的一个对象之间的关系。另见描述符 (descriptor)。

链接目标 (link target) : 请参阅目标 (target)。

链接 (link) : (1) 在文件系统中, 在 i-node 和与它关联的一个或更多个文件名之间的连接。(2) 在数据通信中, 一个与邻近节点之间的传输数据一起的传输介质和数据链路

控制组件。(3) 在编程中, 一个程序中在计算机程序分离的部分之间传递控制和参数的部分。(4) 为了互连数据项或一个或更多个计算机程序的各部分, 比如通过链接编辑器链接对象程序或通过指针链接数据项。(5) 请参阅 X.25 链路 (X.25 link)。(6) 请参阅超文本链接 (hypertext link)。

链路访问过程 (LAP 或 LAPB) (link-access procedures, LAP or LAPB): 在 X.25 通信中, 用于在 DCE 和 DTE 之间数据交换的链路级别元素, 它运行在用户服务的 8 类到 11 类, 就象在 CCITT Recommendation X.1. LAPB 中指定的是一个双面, 异步协议, 用来进行点对点通信。另见 LAPB。

链路跟踪 (link trace): 在链路上发生的一系列有顺序的日志事件。该日志可以帮助确定再次出错源。

链路级别 (link level): 请参阅帧级别 (frame level)。

链路控制 (link control, LC): 请参阅逻辑链路控制 (logical link control)。

链路站 (Link station): 数据链路控制部件, 负责在一个单一的逻辑链路上传输数据。

良好公民权 (good citizenship): 由《客户机间通信约定手册》(ICCCM) 建立的一组标准, 受“AIXwindows 窗口管理器”支持并由它来实现。这些标准规定多客户机环境中的客户机之间的行为, 以避免产生代价高昂的兼容性问题。

量子化 (quantization): 将一个变量的取值范围细分成有限个子区间, 这些子区间不重叠但不一定相等。每个间隔用一个给定值表示。

量子 (quantum): 量子化中的一个子区间。

聊天脚本 (chat script): 远程通信中, 一个调制解调器用来与另一个调制解调器建立通信链路的期待 - 发送 (expect-send) 序列列表。另见握手 (handshaking) 和期待 - 发送序列 (expect-send sequence)。

列标题 (column headings): 为了标识或为各列中的数据给出标题的用途而出现在数据列顶部附近的文本。

列表字段 (list fields): 请参阅索引字段 (indexed fields)。

列表 (list): (1) 由一系列相关记录集合组成的一个数据对象。(2) 有序数据集。

列英寸 (column inch): 打印的文本的计量单位。一个列英寸是在一英寸的输入深度 (一个列宽) 内所包含的文本数量。

列 (column): 字符或其它表达式的水平安排。

临界段 (critical sections): 共享数据的多个部分, 必须防止多个线程或应用程序对其进行同时访问。

邻居网关 (neighbor gateway): 外部网关获取的一个同级设备。所有外部网关不与所有其它外部网关通信。它们获取邻居, 再通过邻居通信。

临时标量 (temporary scalar): 在源级别优化中, 用于临时容纳数组元素内容或计算结构以除去该元素或结果的逻辑冗余存储和负载。

临时文件 (scratch file): 指一种文件, 此文件通常用作工作文件; 它暂时性存在, 直到使用它的程序结束。

零调制解调器 (null modem): 请参阅调制解调消除器 (modem eliminator)。

零缓冲区或零缓冲 (z-buffer or z-buffering): 在 3D 计算机图形操作中, 它适用于通常用作除去隐藏行和隐藏表面的辅助的设备或技术。如果启用零缓冲区, 每个像素存储一个深度值和一个颜色值。在简单的术语中, 深度可看作从查看器眼睛到像素的距离。每当绘制例程尝试更新像素时, 它首先检查当前像素的“深度”或“z-value”, 且如果新的像素比当前像素更近, 它仅会更新具有新值的像素。存储 z-value 的内存区域也称作零缓冲区。另见隐藏表面 (hidden surface)。

令牌环 (token ring): 在“IEEE 802.5 子委员会”支持下开发的局域网的一种。令牌访问使用顺序 (环路) 拓扑结构的过程。

零位插入和删除 (zero bit insertion and deletion): 每四个 1 位后插入一个零, 然后除去零以使数据返回正常状态。

流编辑器 (stream editor): sed 命令, 此命令根据一个编辑脚本修改来自特定文件的行, 并将更改写至标准输出。

流程图 (flow diagram): 对于流图表不建议使用的术语。

流缓冲区 (stream buffer): C++ 流缓冲区是最终用户和格式化数据的“I/O 流库”函数之间的缓冲区。通过 streambuf 类和自 streambuf 派生的类, 它实现于“I/O 流库”中。

浏览 (browse): (1) 当选择导航窗口底部的“书籍列表”按钮时的可用功能。可以与硬拷贝书刊翻页相同的方式向前向后翻动在线书刊。(2) 在命令 xtalk 和 xroute 中, 查看路由表或地址列表项的详细信息。

流量控制 (flow control): 在 X.25 通信中, 控制数据传输率的过程。

流末端 (stream end)： 距离用户进程最远的流的末端。流末端包含驱动程序。

流式磁带机 (streaming tape device)： 请参阅流式磁带机 (streaming tape drive)。

流式磁带机 (streaming tape drive)： 存储大量数据的磁带设备，此设备设计为进行无停止转储或在不使用块间隔的情况下恢复磁盘。

流收集 (stream collection)： 收集将审计记录写至内核中环形缓冲区的审计数据的方法。可以显示、打印或提供纸审计跟踪或可转换为二进制记录的数据。

流水线选项 (pipeline options)： 在 GL 中，在图形流水线中控制处理流程的变量。例如，光线是一个流水线选项。如果光线打开，图元的颜色就可通过估计光线方程式来获得。如果光线关闭，就使用最后指定的颜色。其它流水线选项是：面向后标志、阴影样式标志、深度提示标志、检取标志、颜色方式（颜色指数或 RGB）标志、z 缓冲区标志（启用或禁用向 z 缓冲区绘制）等等。另见属性 (attribute)。

流水线 (pipeline)： (1) 两个或多个进程之间的直接单向连接。(2) 处理器的串行排列或处理器内存寄存器的串行排列。每个处理器或寄存器执行一部分任务并将结果传送给下一个处理器。不同任务的多个部分可同时执行。(3) 串行执行一些进程。(4) 在前一个指令序列执行完之前就开始运行一个指令序列，以便提高处理速度。另见管道 (pipe)。

流头 (stream head)： 距离用户进程最近的流的末端。流头提供了流和用户进程间的接口。流头的主要功能就是处理与 STREAMS 相关的系统呼叫以及在用户进程和 STREAMS 内核空间内消息之间的数据和信息的双向传送。

流 (stream)： (1) 来自打开文件的描述符的有序的输入或输出。(2) 被传输的或将要用于传输的数据元素的连续流，它使用已定义的格式。(3) 所有通过数据通道传输的数据以单一读操作或写操作传输。数据流 (data stream) 的同义词。(4) 通过连接 STREAMS 组件创建的内核聚集，这些组成部分是由 STREAM 机制的应用程序产生的。主要组成部分是一个流头、一个驱动程序以及流头与驱动程序间零个或更多可推动模块。流在用户进程与驱动程序之间的内核中形成一个全双工处理和数据传送路径。除了数据流和数据处理是双向的这一特点，流类似于 shell 流水线。

漏斗作用 (funnelling)： 强制未知的设备驱动程序和内核扩展是 MP 安全的，以仅在主处理器上运行。

路径列表 (path list)： 包含文件的全路径名的结构，或对应的参数。

路径名 (path name)： 指定通向该文件的所有目录的文件名。另见全路径名 (full path name) 和相对路径名 (relative path name)。

路由表 (routing table)： 保留一系列有效路径的一种表，通过它主机可以与其它主机进行通信。路由表可保留静态路由和动态路由。

路由列表 (routing list)： 一种列表，它将用户名同网络地址和其它信息联系起来，以便引导进入 X.25 呼叫。

路由选择信息协议 (Routing Information Protocol, RIP)： XeroxNS 路由信息协议的一种变体，用于维护当前内核路由表条目。

路由 (route)： 为在网络上发送数据而定义的一种路径。

路由 (routing)： (1) 指定一条路径，消息通过此路径到达目的地。(2) 在 SNA 中，按照消息单元中传送的参数的规定，将此消息单元沿着一条特定的路径通过网络转发，例如，按传输头中的目的地网络地址传送。(3) 在 X.25 通信中，包通过它到达目标用户。

轮询测试 (wrap test)： 一种测试，它通过将设备的输出作为输入返回来检查附件或控制器环路而不是检查设备本身。

轮询 I/O 设备 (polled I/O devices)： 由用户进程读取其当前值的设备（键盘、鼠标、按钮、拨号）。

轮询 (poll)： (1) 确定在数据通信中是否有站已做好发送信息的准备而进行的询问。(2) 运行轮询序列。

轮询 (polling)： (1) 在多点连接或点到点连接上，一次请求一个数据站传送数据的过程。(2) 为避免争用，确定运行状态或确定数据发送或接收的准备状况而对设备进行的询问。

罗马数字 (Roman numerals)： 罗马字母的传统组合形成的数，大写 (I、II、III、IV 等) 或小写 (i、ii、iii、iv 等)。另见阿拉伯数字 (Arabic numerals)。

逻辑表达式 (logical expression)： 一种表达式，是由逻辑运算符、关系运算符或二者兼有组成，均可被求值为正或负。

逻辑常数 (logical constant)： 值为正或负的一个常数。

逻辑初等量 (logical primary)： 值为正或负的一个初等量。

逻辑存储 (logical storage)： 一种概念存储布局，首先应用程序映射到逻辑地址，逻辑地址又通过控制块映射到真实地址。

逻辑单元类型 1 (Logical Unit Type 1, LU1) : 一种 SNA 会话, 它支持应用程序和多输入/输出设备间的通信。这个通信可产生在交互式或批处理环境中。

逻辑单元类型 2 (Logical Unit Type 2, LU2) : 一种 SNA 会话, 采用 3270 设备数据流支持应用程序和显示器间的通信。

逻辑单元类型 3 (Logical Unit Type 3, LU3) : 一种 SNA 会话, 采用 3270 设备数据流支持应用程序和打印机间的通信。

逻辑单元类型 6.2 (Logical Unit Type 6.2, LU6.2) :

(1) 一种 SNA 会话, 在分布式数据处理环境中两个应用程序之间。(2) LU 类型用于 SNA 高级的程序到程序和通信 (APPC)。另见对等通信 (*peer-to-peer communications*)。

逻辑单元 (logical unit, LU) : (1) 一种类型的网络可寻址单元, 可使最终用户与其它用户进新通信并获得网络资源的访问权限。(2) 在 SNA 中, 通过一个端口最终用户访问 SNA 网络与其他用户进行通信, 并通过端口最终用户访问系统服务控制点提供这项功能。一个 LU 至少支持两个会话, 一个和 SSCP, 一个和另一个 LU, 或许也可以支持与其它多个 LU 会话。

逻辑分区 (logical partition, LP) : (1) 一个到三个物理分区 (拷贝)。逻辑卷中逻辑分区的数目是变量。(2) 逻辑卷的固定大小部分。在卷分组中, 逻辑分区和物理分区具有相同的大小。除非逻辑卷的一部分是镜像的, 每个逻辑分区对应并且其内容存储在一个单独的物理分区上。另见逻辑卷 (*logical volume*)。

逻辑卷管理器 (Logical Volume Manager, LVM) : 在逻辑级别管理磁盘空间。它控制固定磁盘资源是通过在逻辑和物理存储器间映射数据, 和通过允许数据跨越多个磁盘, 并且是无接触的、复制的和动态的扩展。

逻辑卷 (logical volume, LV) : (1) 物理分区组织成逻辑分区的集合, 所有包含在一个单独的卷组中。逻辑卷是可扩充的, 并可以在一个卷组中跨越多个物理卷。(2) 逻辑分区的集合, 其中每一个存储在一个或多个物理分区中, 这些物理分区在一个给定的卷组的一个或多个物理卷中。一个逻辑卷有一个设备名 (形式为 /dev/hdn) 并包含一个单独的文件系统。另见日志卷 (*log volume*)、迁移安装 (*migration installation*)、逻辑分区 (*logical partition*) 和卷组 (*volume group*)。

逻辑类型 (logical type) : 一种包含值为正或负的数据类型。

逻辑链路控制 (logical link control, LLC) : 在局域网中, 支配传输帧装配和在数据站之间交换的协议, 独立于介质访问控制协议。另见介质访问控制 (*medium access control*)。

逻辑链路 (logical link) : S/370 应用程序和 workstation 应用程序间的逻辑连接。

逻辑名 (logical name) : 在系统中, 分配给一个设备以使其与系统中其它所有设备实例相区分的名称。用于谈及一个特定设备而采用的名称。示例, “tok0”是指令牌环适配器。这和“设备名”相同, 它是一个定制设备对象类的字段。另见设备名 (*device name*)。

逻辑设备 (logical device) : (1) 采用物理设备执行输入和输出的文件。(2) 在虚拟和真实设备间映射用户 I/O 的文件。

逻辑通道号 (logical channel number, LCN) : 一种唯一识别逻辑通道的号码。

逻辑通道 (logical channel) : 在 X.25 通信中, 通过一个数据链路的双工同步传输方法由相关发送和接收通道组成。逻辑通道表示从发送源到网络或从网络到目的地的数据通道路径。另见通道 (*channel*)。

逻辑网络 (logical network) : 一个具有整体网络和独立网络功能的计算机的子网。通常用作一个大的物理网络子网的逻辑网络。

逻辑运算符 (logical operator) : 表示一种运算的符号, 例如, 逻辑表达式中的 AND、OR、NOT。

逻辑运算 (logical operation) : 遵循布尔逻辑规则的运算。

逻辑资源 (logical resource) : 一个软件构造, 例如, 一个锁定或一个缓冲区, 它们是程序执行所要求的, 并且它们的供给是受限制的。

[M]

满意 (satisfy) : 对于 Ada 编程, 请参阅约束 (*constraint*) 和子类型 (*subtype*)。

盲页码 (blind folio) : 一文档已计算页码, 但没有编码。另见页码 (*folio*)、断页码 (*dropped folio*) 和快速页码 (*expressed folio*)。

冒号格式 (colon format) : 其中数据文件可组织的一种格式。每个数据记录占据冒号文件中的一行, 每个数据记录中的数据字段用冒号隔开。

枚举标记 (enumeration tag) : 命名枚举数据类型的标识符。

枚举标量类型 (enumerated scalar type) : 一种通过枚举该类型的元素来定义的标量。每个元素由一个标识符来表示。

枚举常量 (enumeration constant)： C 语言中，在枚举元中定义的标志符（有一个关联的整数值）。可以在允许整数常量使用的任何地方使用枚举常量。

枚举类型 (enumeration type)： Ada 语言中一种离散类型，其值由在类型声明中显式给出的枚举文字表示。这些枚举文字或者是标识符，或者是字符文字。

枚举数据类型 (enumeration data type)： 一种代表整数和枚举常量集的数据类型。每个枚举常量都有一个关联的整数值。

枚举元 (enumerator)： 枚举常量和与其它相关联的值。

美国国家标准协会 (American National Standards Institute, ANSI)： 一个由计算机和商业设备制造协会 (CBEMA) 赞助的组织，通过其公认的组织创建和维护非官方工业标准。

美国国家信息交换标准码 (American National Standard Code for Information Interchange, ASCII)： 该代码由美国国家标准协会 (ANSI) 开发，用于数据处理系统、数据通信系统和相关设备间的信息交换。ASCII 字符集包含 7 位控制字符和符号字符。

每进程数据区 (per-process data area)： 在内核方式中的用户进程堆栈段的一部分。该区域按进程分页，它包含像由进程或 I/O 方式下输入打开的文件的当前目录这样的进程信息。这些信息放在堆栈段的顶部。另见用户块 (user block) 和用户结构 (user structure)。

每字符位数 (bits per character)： 数据字符里的位数。

米 (meter, m)： 一个线性度量，它等于 1.0936 码，3.2808 英尺，或者 39.3696 英寸。

幂等 (idempotent)： 运算的一类。如果一个运算的结果不影响任何运算的结果，则该运算是幂等的。例如，返回时间的调用是幂等的。

密度 (density)： 在打印中指水平每英寸可打印的字符数。

密封 (seal)： 对包含多个字段的记录加密，这样如果不知道加密密钥或留下篡改证据，这些字段就不能修改。

密码安全性 (password security)： 需要用户输入密码登录系统的进程。

密码 (password)： (1) 只为用户和系统知晓的字符串。用户必须指定正确的密码，以获取对系统和用它存储的数据的访问。(2) 用关于软件供应商（供应商密码）或关于软件产品（产品密码）的信息编码的字符串。

密文 (ciphertext)： 加密函数的输出。将明文加密转换成密文。

密钥锁功能 (keylock feature)： (1) 一个安全性功能，在该功能中一个锁和密钥可以被用来限制显示站的使用。(2) 一个限制了键盘使用的程序。

面板 (panel)： (1) 用来跟计算机用户相互交流信息的在屏幕上显示的一系列逻辑相关的信息。(2) 一个或多个窗格作为一个单元来处理。面板的窗格一起显示、一起擦除、并通常表示到使用应用程序的人的信息单元。一个面板用显示为用窗格平铺（完全填充）的长方形区来表示。

面向对象编程 (object-oriented programming)： 基于数据抽象和继承概念的编程方法。和程序上的编程技巧不同，面向对象编程集中在不是如何完成而是代以问题包含什么数据对象以及如何处理它们。

面向连接的协议 (connection-oriented protocol)： 一种基于连接的，可靠的虚电路传输协议，如 TCP；一个运行在基于连接的传输协议之上的 RPC 协议。

描述符： (1) 在 ODM 中命名并确定类型后的变量，该变量定义对象的单个特征。另见结束描述符 (terminal descriptor)、链接描述符 (link descriptor) 和方法描述符 (method descriptor)。(2) 在信息检索中，它是用来对信息进行分类和索引信息的参数词。(3) 在 XOM 中，它是客户机和服务交换属性值和指明其表示方法、类型和语法的整数的方法。(4) 在 XDS 中，它是用来代表 OM 属性类型和单个值的已定义的数据结构。

名称编码 (mangling)： 在编译期间，对 C++ 标识比如函数和变量名包括类型和作用域信息的编码。链接程序利用这些名称编码确保类型安全链接。

名称查找方法解析 (name-lookup method resolution)： Similar to the 方法解析 (method resolution) 技术用在面向对象的 C 和 Smalltalk 中。

名称服务器 (name server)： 为网络提供名称解析的主机。名称服务器将赋给网络和主机的名称转化为机器所用的有效的因特网地址。

名称解析 (name resolution)： 将一个符号名称转化（解析）为它的更有效的因特网地址的过程。

名称字符串 (name string)： 标识某种诸如图标、资源或字体等对象的一个字符串。

名称 (name)： (1) 从 1 到 250 字母数字混合编制的序列，第一个必须是字母，用它标识一个数据对象。(2) 在 Ada 语言中，名称是代表一个实体的构造，这称为名称表示实体，实体是名称的意义。另见声明 (declaration) 和前缀 (prefix)。(3) 在 C++ 中，一个名称通常指一个标识符。然而，在句法上，一个名称可以是一个标识符、操作函数名称、转换函数名称、析构函数名称或限定的名称。另见符号名称 (symbolic name)。

明文原因 (clear cause) : 请参阅原因代码 (cause code)。

明文 (plaintext) : 加密函数的输入或解密函数的输出。解密转换密码文本到明文。

命令编程语言: 允许通过命令的组合而不是用传统编程语言写语句进行编程的工具。

命令方式 (command mode) : 用户可以输入命令的系统或设备状态。另见文本输入方式 (text input mode)。

命令行编辑键 (command line editing keys) : 编辑命令行的键。

命令行 (command line) : 输入命令时显示命令的屏幕区域。

命令解释器 (command interpreter) : 一个发送指令给内核的程序。与接口 (interface) 同义。另见 shell。

命令历史记录 (command history) : 一个先前发出命令的自动列表。

命令名称 (command name) : (1) 命令中的第一个或主体术语。命令名称不包含参数、值、标志或其它操作数。(2) 当计算机识别缩写格式时的命令全名 (例如, 打印当前工作目录 (print working directory) 代表 pwd)。

命令模块 (command module) : 执行命令或进程的文件。用户在命令行或由另一个命令模块输入命令时, 文件可能是激活的。

命令替换 (command substitution) : 通过将命令行置于 `` (重音符) 中, 捕获任何命令的输出作为另一个命令的值的的能力。shell 首先运行该命令或包含在重音符内的命令, 然后以它们的输出替换整个表达式 (包括重音符)。该功能经常用于赋值语句中。

命令帧 (command frame) : 一个作为命令使用并 (多数情况下) 期待响应的链路级帧或信息包。

命令字符串 (command string) : 执行操作请求, 除操作数之外还提供运行运算所需的所有指示信息。

命令字 (command word) : 用于存储图元字符串的 16 位单元名称。第一个命令字确定图元类型并设定字符串长度。后继命令字包含以磅 (或 4 位数据) 倍数计的信息。

命令 (command) : (1) 一个执行操作或运行程序的请求。当参数、值、标志或其它操作数与命令相关联时, 结果字符串是单一命令。(2) 数据通信中, 一个由帧控制文件描述, 由主或混合站发送的指令。它引起寻址站运行数据链路控制功能。

命名公共块 (named common) : 在 FORTRAN 中, 一个单独的由变量和数组组成的命名的公共块。

命名管道 (named pipe) : 应用程序通过名称打开从管道读写数据的管道。按照约定, 管道被放在 /dev 目录当中, 被看作特殊的文件。用命名管道设施帮助在一个发送进程和一个接收进程之间进行通信。

命名作用域 (naming scope) : 请参阅作用域 (scope)。

命中 (hit) : 在 GL 中也称为拾取命中 (pick hit) 或选择命中 (select hit)。只要在拾取或选择区域中绘制某个图元时都会发生命中。只有在自上一次命中以来名称堆栈更改的情况下, 才将命中报告返回给用户。换句话说, 即使只报告一次拾取 / 选择事件, 也可能发生多次命中。另见拾取 (picking)、拾取区域 (picking region)、选择区域 (selecting region) 和选择 (selecting)。

模板 (template) : (1) 键盘的一种表示, 包括没有刻在键盘上的功能。(2) 存储在缓冲区中的每个命令行。(3) 在增强的编辑方式下, 与终端相关的特殊字符缓冲区。(4) “邮件程序”中, 您可以创建的 ASCII 文件, 便于包括经常输入的信息, 比如您的邮件消息中您的名称和电子邮件地址。(5) C++ 类或带有变量类型的函数系列。

模糊 (在“增强的 X-Windows”中) (obscure, In Enhanced X-Windows) : (1) 窗口存在的状态。这个窗口是模糊的, 如果另一个在它前面的窗口使模糊的窗口部分可见。窗口 B 被窗口 A 模糊, 如果二者是可见的 InputOutput 窗口且 A 在全局堆栈次序较高, 用 A 的外边定义的矩形和用 B 的外边定义的矩形相交。窗口边界包含在计算中, 窗口有可能模糊然而仍有可见的区域。(2) 一个窗口操作另一个操作当它遮住另一窗口的一部分时。如果二者都是可见的 InputOutput 窗口, 那么窗口 A 模糊窗口 B, A 在全局堆栈的次序较高, 用 A 的外边的定义的矩形和用 B 的外边定义的矩形相交。对照小窗口可见性 (widget visibility)。

模块 (module) : (1) 一个离散的编程单元, 通常执行一个特定的任务或任务集。模块是先被单独汇编然后链接成一个完整程序的子例程和调用程序。(2) 在编程语言中, 由过程或数据声明组成同其他这样的构造交互的语言构造。(3) 一个封装的功能性硬件单元, 设计用来同其它组件一起使用。(4) 程序单元 (program unit) 的同义词。(5) 另见运行文件 (run file)。(6) 执行报文上的函数的软件, 当这些报文在流报头和驱动程序之间流动时。一个模块 (module) 是与 shell 流水线中的命令相对应的 STREAMS 模块, 除模块包含了一对允许独立的双向 (下游和上游) 数据流流动和处理的函数外。

模拟坐标、模拟空间 (modeling coordinates, modeling space) : 在 GL 中, 所有的图元的绘制都在其中的坐标系。通过转换、旋转、比例或通用变换, 用户可以选择关于总体空间的模拟空间的位置和方向。模拟坐标系和世界

坐标之间的关系由模拟矩阵来决定。当绘制复杂的或重复的景物时，模拟坐标是一个有用的概念上的设备。例如，一个曲别针可以在模拟坐标系中定义一次，然后通过在整个空间移动模拟坐标绘制它几百次。另见视觉坐标 (*eye coordinates*)、屏幕坐标 (*screen coordinates*)、世界坐标 (*world coordinates*)、和变换 (*transformation*)。

模式操作 (pattern-action)： 当 **awk** 命令在输入数据文件（该文件匹配程序文件中的行）中找到模式时，它执行该行上关联的操作。

模式地址 (pattern address)： 通过包含在行中的字符串引用行，而不是通过数字或符号地址。模式地址可以是字符串或正则表达式 (*regular expression*)。另见符号地址 (*symbolic address*)。

模式匹配字符 (pattern-matching character)： 文件中用于匹配一个或多个字符的特殊字符，比如 * (星号) 或 ? (问号)。例如，将 ? 放置在文件规范中，意思是在该位置中可以是任何字符。与通配符 (*wildcard*) 同义。

模式匹配 (pattern matching)： 指定系统应该查找的字符的模式。

模式字符串 (pattern strings)： 由特殊的模式匹配字符构成的正则表达式的字符串。模式字符串可用于地址中指定行，也可用在一些子命令或一行的若干部分中。

模式 (pattern)： (1) 定义一个搜索模式的一个正则表达式或一系列正则表达式。(2) 在 GL 中，定义系统显示上多边形纹理的一个 16x16、32x32 或 64x64 数组位。

模态对话框 (modal dialog)： 在 AIXwindows 中的一个对话框小窗口，它可以中断工作会话以请求来自用户的输入。

模态交互 (modal interaction)： 在各个方式或函数之间的通信。

模态弹出 (modal pop-up)： 在“增强的 X-Windows”中的一个窗口，它在正常时对窗口管理器而言是可用的，只有在管理器关闭之后才可用。该弹出除在对话框中出现的事件外禁用用户事件处理。

模态 (modal)： 次级窗口接收键盘或指针输入而不传递到它的关联窗口所处的状态。

模型号 (model number)： 在 Ada 语言中，一个真实类型的确切代表值。一个真实类型的操作根据该类型的模型号上的操作所作的定义。根据真实类型的所有实现，模型号和它们的操作的属性是保留的最小属性。

末行方式 (last line mode)： 在 vi 编辑器中的一种命令方式。使得用户可以在屏幕的底部输入一个命令。另见文本输入方式 (*text input mode*)。

目标程序 (target program)： 对象程序 (*object program*) 的同义词。

目标代码 (object code)： (1) 能被机器运行的指示信息，通常由用高级语言（比如 C 语言）写成的源代码编译产生。必须被链接的程序，目标代码由可重定位的机器代码组成。(2) 从编译器输出的或者汇编程序都是它自己的可执行机器代码或是产生可执行机器代码的相应处理。对照源代码 (*source code*)。

目标文件 (object file)： (1) 对象库中的成员文件。(2) 编译器的或者汇编程序可以用绑定程序 (**ld**) 处理来产生可执行文件的原始输出。目标文件的名称通常结束于 **.o**。

目标文件 (target file)： **make** 程序创建包含完整程序的文件。

目标 (target)： (1) 系统、系统中的程序或设备，它们解释、拒绝、或满足以及回答从源接收到的请求。对照源 (*source*)。(2) 安装密码于其上的节点。如果密码指定单一节点锁定许可证，目标就是许可运行产品的节点。如果密码指定多重节点锁定许可证（即，节点锁定许可证的复合密码），或其它任何类型的许可证，那么目标就是运行许可证服务器的节点。(3) 对于 NIM 则为您正在安装的客户机。

目的磁盘 (destination disk)： 安装要使用的磁盘。

目的光标 (destination cursor)： 由光标标记的一点或一个位置，数据将在这里被粘贴或插入。

目录标识 (directory ID)： 请参阅目录标识 (*directory identifier*)。

目录标识 (directory identifier (directory ID))： 一个用来区分安装中目录服务的几种不同配置的标识。

目录掩码 (directory mask)： 一种字符模式，它控制目录的哪些部分将保留，哪些部分将不保留。

目录 (catalog)： 一组用来创建 Interleaf 文档的预定义组件和属性。

目录 (directory)： (1) 包含名称并控制其它文件或目录的信息的文件类型。(2) 一个标识符和指向相应数据项的引用的表。(3) 由控制程序用来在直接访问存储中定位独立数据单独集区域中存储的数据块的索引。(4) 对照特殊文件 (*special file*)。(5) 一个相关文件的清单，以实用的层次结构安排。(6) CDS 中用来在一个名称（目录名）下以 CDS 名称空间存储条目的逻辑单元。除对象项之外，目录可以包含软链接和子指针。可以对目录复制、删除及控制对目录的访问。目录的每个物理实例称为一个副本。(7) XDS 中的一个开放系统的集合，它们相互合作以保存有关

一组真实世界中对象的信息的逻辑数据库。(8) CDE 中文件和其它子目录的集合。在图形用户界面应用程序中可以称作文件夹 (*folder*)。

[N]

内部节点 (i-node) : 内部结构用来描述在操作系统中的单个文件的内部结构; 每一个文件有一个 i-node。一个 i-node 包含文件的节点, 类型, 所有者和位置。i-node 的一个表邻近存储在一个文件系统的开始。同义与文件索引 (*file index*)。请参阅 *i-list* 和 *i-node* 号 (*i-node number*)。

内部进程间通信 (inter-process communication, IPC) : (1) 用于程序间的数据相互通信和活动的同步。信号量、信号和内部报文队列是内部进程间通信的公共方法。(2) 在“增强的 X-Windows”中的一种通信路径。另见客户机 (*client*)。

内部例程 (internal routine) : 仅在所声明的词法作用域可用的例程。

内部时钟 (internal clocking) : 在数据通信中, 由适配器提供的数据时钟。

内部数据定义 (internal data definition) : 在块开始处出现的变量描述。此块指示系统为变量分配存储, 并使变量对当前块可访问。

内部数据结构 (internal data structures) : 由程序或函数在内部存储的其它数据格式。用于执行变量列表的类型转换。

内部网关 (interior gateway) : 一个只与本身自治系统的网关进行通信的网关。

内存取消分配 (memory deallocation) : 释放由于一个特定目的而先前分配的内存。

内存负载控制 (memory load control) : 一个 VMM 工具, 它可以检测内存过量使用和临时减少正在运行进程的数目, 这样可以避免系统颠簸。

内存过量使用 (memory over-commitment) : 当前运行程序利用的虚拟内存页面数量超出了可用的物理内存页面帧的情形。如果过量使用十分巨大或者还在持续, 将对系统性能造成损害。

内存行 (line of memory) : 与一个高速缓存行相对应的内存部分, 它与一个单个虚拟内存地址标签相对应。

内存泄漏 (memory leak) : 程序分配内存时的一个软件错误, 失去了跟踪, 然后分配更多。如果程序长期运行, 最终将占用大量的物理内存和页面调度空间。系统性能逐

步恶化; 由于资源丢失的程序失败并不是故障所在地。内核扩展的内存泄漏要内存引脚内存可能代价更高。

内存映象 (memory image) : 内存种的部分进程的逻辑布局。

内存转储 (memory dump) : 计算机系统在发生故障时记录它的状态的方法。

内存 (memory) : (1) 可寻址的程序存储器, 通过它指令和其它的数据可被直接装入到寄存器以便后续的程序运行和处理。(2) 电子芯片上的存储器。存储器的示例有随机存取存储器, 只读存储器, 或者寄存器。请参阅存储 (*storage*); 存储器; 存储量。

内核参数 (kernel parameters) : 指定内核如何分配某些系统资源的变量。与系统参数 (*system parameters*) 同义。

内核方式 (kernel mode) : 进程在其中运行内核代码的状态。对照用户方式 (*user mode*)。

内核设备驱动程序 (kernel device driver) : 请参阅设备头 (*device head*)。

内核转储 (kernel dump) : 与系统转储 (*system dump*) 同义。

内核 (kernel) : (1) 操作系统的一部分, 包含诸如: 输入/输出、管理和硬件控制以及用户任务的调度等程序任务。(2) 包含频繁需要的功能的基本操作系统的一部分。

内联函数 (inline function) : 在 C++ 中, 内联是对编译器执行一个函数成员体的内联扩展的一个技巧。在一个类定义中, 同时声明和定义的函数是内联函数。也可以通过使用关键字 **inline** 显式声明一个内联函数。成员和非成员函数都可以被内联。

内联扩展 (inline expansion) : 一个在其中对过程的引用为过程自身代码所取代的优化方法, 以消除调用开销。

内联组件 (inline component) : 在一个有自己的属性, 能包含组件所包含的任何东西的组件中的组件。它由一个组件的相同设置来创建。

内联 (inline) : 请参阅内联组件 (*inline component*)。

内联 (inlining) : 请参阅内联扩展 (*inline expansion*)。

内置函数 (built-in function) : 编译器可知的函数, 其代码直接合并到程序模块, 而不是通过库调用引用。

逆的转置 (inverse transpose) : 矩阵转置后求逆。

逆矩阵 (inverse) : 对矩阵进行数学运算所得到的矩阵结果。如两个矩阵相乘可得到一个单元矩阵。

匿名联合 (anonymous union) : C++ 中没有类名的联合。其后不得跟声明符。

扭曲 (twist) : 沿着视觉线旋转。

挪用 (页面帧) (steal (a page frame)) : (“虚拟内存管理器”进行的) 重新分配实内存页面帧的行为, 此帧包含正由当前的执行程序使用的虚拟内存页面。

[O]

耦合器 (coupler) : 一个将调制解调器连接到电话网络的设备。

[P]

排队 I/O 设备 (queued I/O devices) : 其更改记录到事件队列的设备 (键盘、鼠标、按钮、拨号)。

派生类型 (derived type) : 在 Ada 语言中操作和值都是从已有类型副本的类型。已有的类型被称为派生类型的父类型。另见父类型 (parent type)。

派生类 (derived class) : 继承基类属性的 C++ 类。可以将额外的数据成员和成员函数添加到派生类中。派生类对象可以象基类对象一样操作。派生类可能会覆盖基类虚拟函数。

派生 (derivation) : 从已有类 (称为基类) 派生 C++ 类的过程。

派生 (fork) : 创建并启动子进程。另见执行 (exec)。

判别联合 (discriminated union) : XDR 中拥有几种数据类型的 C 语言联合, 其中联合的一端为枚举值或判别式, 它保存系统中首先要处理的特定对象。

判别式约束 (discriminant constraint) : Ada 语言中在记录类型或私有类型上指定每种类型的判别式的值。

判别式 (discriminant) : Ada 语言中的一种对象的可区分组件或记录类型的值。其它组件的子类型, 甚至它们的存在或不存在, 都与判别式的值有关。

配置操作/过程 (configuration operation/procedure) : 构造 3601 金融通信控制器 (Finance Communication Controller) 配置映象的多步骤过程 (在主机计算机执行)。

配置反馈窗口 (configuration feedback window) : 在 AIXwindows 中, 当 “AIXwindows 管理器” 重新启动或请求行为切换时, 一个显示在屏幕中央的窗口。该窗口包含诸如客户机窗口大小的位置的变量。

配置方法 (configure method) : 将设备从定义状态引至可用状态。如果设备有设备驱动程序, 那么配置方法负责装入和绑定驱动程序到内核。如果设备支持可选的停止状态, 那么配置方法将设备从定义状态引至停止状态。

配置管理器 (configuration manager) : 一个在初始程序装入 (IPL) 期间管理设备配置的程序。

配置规则对象类 (Configuration Rules Object Class) : 一个包含初始程序装入 (IPL) 期间由配置管理器使用的配置规则的对象类。

配置文件 (configuration file) : 一个指定系统或子系统特征的文件; 例如, 操作系统的队列系统。

配置助手 (Configuration Assistant) : 一个用于执行安装后系统配置任务的图形界面应用程序。

配置 (configuration) : (1) 组成数据处理系统或网络的机器、设备和程序组。(2) 对系统已安装设备、可选功能部件和程序产品的描述过程, 以便可以使用这些功能部件。对照定制 (customization) 和系统定制 (system customization)。

配置 (configure) : 向系统描述已安装到系统的设备、可选功能部件和程序产品。

批处理打印 (batch printing) : 把一个或多个要打印文档作为后台进程在单独作业里排队。同时操作员可以输入或修改其它文档。这是一个后台进程。另见后台进程 (background process)。

批处理 (batch processing) : 一种处理方法, 其中一个或多个程序只要较少或无需操作员操作就可处理记录。这是一个后台进程。对照交互式处理 (interactive processing)。

皮特 (peta) : 2 的 50 次方。

片段 (fragment) : 磁盘存储单元, 小于一页 (4KB)。

片假名 (katakana) : 一种符号的字符集, 主要用于按照发音书写外文的文字, 包含在两种常用日文发音字母中。

偏移量 (offset) : (1) 在 Pascal 中, 在 SPACE 数据类型里的选择机制; 通过置括弧里的整数值来选择元素。SPACE 的起源是基于零。(2) 在出版物中, 文本块的所有行的缩排在第一行之后。

偏移 (skew) : 两个时钟或时钟值之间的差别。

频率调制 (frequency modulation, FM) : 按照提供信息的信号来修改固定振幅载波信号的频率, 以便它能传送数据信号。

屏蔽双绞线 (shielded twisted pair) : 一种由两个带有薄片或编织物屏蔽的双绞线构成的传输介质。

平滑 (antialiasing) : 用于平滑“锯齿线”的技术, 不然的话这些“锯齿线”会因为扫描转换出现在线上或多边形的边上。常用技术包括像素位置调整或按照覆盖在每个点上的像素区域的百分比设定像素强度。

平假名 (hiragana) : 由两种常用的日文语音字母表之一中使用的符号组成的图形字符集。每个字符用一个字节表示。

瓶颈 (bottleneck) : 程序中低效使用 CPU 资源的代码部分, 因而导致不必要的延迟。

平面网络 (flat network) : 在这样的网络中, 所有主机都由中心机构管理。

平面文件 (flat file) : (1) 没有层次结构的文件。(2) 一维或二维数组。项的列表或表。

屏幕保护程序 (Screen Saver) : 在 CDE 中, 指过了特定时间阶段后, 关闭工作站显示或改变显示图像, 以此来延长屏幕寿命的选项。

屏幕空间 (screen space) : 屏幕坐标 (*screen coordinates*) 的同义词。

屏幕屏蔽区 (screenmask) : GL 中屏幕的一个矩形区域, 所有的绘制操作都剪切至此区域。它通常设置为和视口与窗口一样大小。屏幕屏蔽区对字符剪切非常有用。另见剪切 (*clipping*)。

屏幕锁定: 在 CDE 中, 指锁定工作站屏幕, 在输入有效的用户密码前一直禁止进一步输入的功能。

屏幕抓取 (screen capture) : 在磁盘上屏幕的存储显示为文本或图形文件。

屏幕坐标 (screen coordinates) : 定义显示屏的坐标系。在 GL 中, 距离由像素单位度量, 且原点在左下角。大多数系统上的屏幕大小为 1024 像素高, 1280 像素宽。视口定义了从规格化设备坐标至屏幕坐标的映射。与屏幕空间 (*screen space*) 同义。另见视觉坐标 (*eye coordinates*)、图元坐标 (*primitive coordinates*)、模拟坐标 (*modeling coordinates*)、世界坐标 (*world coordinates*) 和变换 (*transformation*)。

屏幕 (screen) : (1) 请参阅显示屏 (*display screen*)。(2) 在扩展 curses 库中, 和工作站屏幕一样大小的一种特殊类型的窗口。(3) 在“增强的 X-Windows”中, 指一种服务器, 它可以提供多个通常具有物理上独立的监视器 (显示屏) 的独立屏幕。当在所有屏幕中只有一个单独的键盘和指针共享时, 这是期望配置。屏幕结构包含有关该屏幕的信息且链接至显示结构。

平铺字符 (flattened character) : 把扩展字符转换为非常类似于它的 ASCII 字符时创建的 ASCII 字符。该代码点信息丢失且该字符不能被重新转换为扩展字符。例如, 一个 cedilla 字符将被平铺为一个普通的“c。”

平移 (translation) : (1) 某物一定距离的移动。(2) 在计算机图形中, 显示图像从一个位置到另一个位置在一直线上的移动。另见转换 (*transformation*)。

普通文件传送协议 (Trivial File Transfer Protocol) : 使用最少的协议在主机间传送文件。

[Q]

期待 - 发送序列 (expect-send sequence) : 在远程通信中, 一个程序或调制解调器期望从远程系统收到的字符和信号列表, 紧随着的是程序和调制解调器在收到期望输入后应该发送给远程系统的字符和信号。这个序列也可以包含一个子序列, 指出程序或调制解调器如果没有收到期望输入应该发送什么。另见握手 (*handshaking*) 和聊天脚本 (*chat script*)。

齐次坐标 (homogeneous coordinates) : 表示三维空间的四维方法。齐次坐标中的点 (x, y, z, w) 通过采用 $X=x/w$ 、 $Y=y/w$ 和 $Z=z/w$ 来表示三维空间中的点 (X, Y, Z)。另见参数双三次曲面 (*parametric bicubic surface*)。

启动方法 (start method) : 将设备从停止状态引至可用状态。启动方法仅适用于支持可选的停止状态的设备。

启动集 (startup set) : 一组可看作用于管理目的的单个单元的应用程序服务器。

起始单元 (home cell) : 请参阅本地单元 (*local cell*)。

企业系统连接 (Enterprise Systems Connection) : 请参阅 ESCON。

企业系统连接 (ESCON, Enterprise Systems Connection) : 一种同时使用激光和 LED 光纤技术以及动态连接概念的 System/390 I/O 体系结构, 它允许路径共享。

启用 (enable) : (1) 使工作。(2) 在交互式通信中, 装入并启动一个子系统。与禁用 (*disable*) 相反。

起止式 (传输) (start-stop) : 一种异步传输, 其中代表字符的一组信号由启动元素开头, 且后面跟着一个停止元素。另见异步传输 (*asynchronous transmission*)。

千克 (kilogram, kg) : 一千克; 2.2046 磅。

迁移安装 (migration installation) : 为了升级当前发行版而保留已存在的根卷组的一种安装方法。该方法保留 */usr*、*/tmp*、*/var* 和 */* (根) 文件系统, 以及根卷组、

逻辑卷和系统配置文件。另见根卷组 (*root volume group*) 和逻辑卷 (*logical volume*)。

千字节 (**kilobyte, K-byte**)：当指内存容量时以十进制表示的 1024 字节；在所有其他情况下，定义为 1000。

千字节 (**K-byte**)：请参阅千字节 (*kilobyte*)。

前后缓冲区 (**front and back buffers**)：在 GL 中，双缓冲区方式下的主帧缓冲区位平面分隔为两组 — 前后缓冲区。前缓冲区平面中的位是可见的，而后缓冲区中的位不可见。通常，一个应用程序进入后缓冲区，并查看前缓冲区的动态图形。

前景色 (**foreground color**)：指定给所有图形元素的一种单一的颜色，它在显示的图象内显示在所有后台图形元素前面。对照背景色 (*background color*)。

前面板控件 (**Front Panel control**)：指 CDE 的前面板中的一个对象，用作基本系统服务和频繁执行的任务和操作的界面。缺省“前面板”中的控件为：时钟、日历、邮件程序、文本编辑器、帮助查看器、文件管理器、打印机、样式管理器、应用程序管理器和垃圾箱。另见控件 (*control*)。

前面板移动句柄 (**Front Panel move handle**)：指 CDE 中前面板每一端的区域，用于移动前面板并使其位于工作区的前面。

前面板 (**Front Panel**)：指 CDE 中位于中央的窗口，包含用来访问应用程序和实用程序的控件，包括工作区切换。“前面板”占有全部工作区。

前台进程 (**foreground process**)：在另一命令发布到 shell 前，必须运行完成的进程。前台进程在前台的进程组中，该组接收由终端生成的信号。对照后台进程 (*background process*)。

前台 (**foreground**)：(1) 运行程序的一种方式，在该方式中，shell 在响应用户输入时等候命令行中指定的程序完成。(2) 在多道程序环境中，最高优先级的程序运行。对照后台 (*background*)。

前向差分矩阵 (**forward difference matrix**)：在 GL，通过添加每一行到下一行且把最底部的那行作为下一点的输出来迭代的 4x4 矩阵。这样生成的点通常落在有理三次曲线上。

前缀 (**prefix**)：在 Ada 语言中，用作某种名称的第一部分。前缀可以是一个函数调用或是一个名称。另见名称 (*name*)。

嵌入式空格 (**embedded blanks**)：由任何其它字符包围的空格。

嵌套类 (**nested class**)：在另一个类的作用域中定义的一个 C++ 类。

嵌套事务 (**nested transaction**)：在另一事务的作用域开始的一个事务。这些也叫做子事务。

嵌套 DO (**nested DO**)：在 FORTRAN 中的一种 DO 循环或 DO 语句，在它们当中范围完全地被包含于另一 DO 循环的范围中。

嵌套 (**nest**)：(1) 把一个结构或多个结构包含到相同类型的结构里面。(2) 把子例程或数据放在其它的不同分层级别的子例程或数据中。因此，该子例程能作为递归的子例程运行，或这样的话数据能被递归地存取。(3) 一个完全地包含类似软件元素的自包含的软件元素，称为有相似的“嵌套的”元素在它里面。示例包含软件循环（被嵌套循环）嵌套在较大的循环（嵌套循环）里面和子菜单嵌套在菜单里。

强调字体 (**emphasized**)：双重打印格式，用这种格式打印的字符，通过一个微小的偏移以多行程（通常为两个）打印，来创建一个模拟的粗体。强调打印以点阵字符形式用来填充间隙和粗糙表面。有时，它也可以打印粗体而无需改变安装的字符集。请参阅粗体 (*boldface*) 和双重突出打印 (*double-strike*)。

强度削减 (**strength reduction**)：用具有较少强度的功能相当的算术优化代替算术操作的优化。例如 $4*2$ 可变换为 $4+4$ 。

强制锁 (**enforced lock**)：一类锁，在该进程中保持一个文件区域，防止任何其它的进程用读或写的系统调用访问那个区域。总的来说，**create** 命令是一种防止截断文件的命令。请参阅咨询锁 (*advisory lock*)。

切换按钮 (**toggle button**)：AIXwindows 和“增强的 X-Windows”中模拟真实世界切换的图形对象；它有序地从一个可选状态切换至另一个状态。

切换 (**toggle**)；开关：(1) 一种切换设备，比如键盘上的切换键。(2) 任何具有两个稳定状态的设备。(3) 在计算机或网络的两种方式间切换。

清除冲突 (**clear collision**)：当 STE 和 DCE 在相同的逻辑通道中同时发送清除请求信息包和清除指示信息包时发生的情况。另见冲突 (*collision*)。

清除请求信息包 (**clear-request packet**)：X.25 通信中，DTE 发送的请求清除呼叫的管理呼叫。另见包 (*packet*)；信息包；分组。

清除确认信息包 (**clear-confirmation packet**)：X.25 通信中由 DTE 发送的通知 DCE 呼叫已经清除的信息包。另见分组 (*packet*)。

清除用户数据 (clear user data) : X.25 通信中, 用户应用程序可选地包含在清除请求信息包中的数据。

清除诊断 (clear diagnostic) : 请参阅 *诊断代码 (diagnostic code)*。

清除指示信息包 (clear-indication packet) : X.25 通信中, DCE 发送的通知 DTE 清除呼叫的管理分组。另见包 (*packet*) ; 信息包; 分组。

清除 (clear) : X.25 通信中拒绝呼叫 (如果还没有接受) 或结束呼叫。

清空代码 (clean-up code) : 在已经开放的 (unrolled) 或封闭的循环中, 一确保所有原始代码的迭代都已经在打开的 (unrolled) 或阻塞的代码中执行的附加循环或循环集。

清空 (clean up) : 清除过程指示系统尝试除去部分安装的软件产品。系统也尝试回复到除去产品的前一版本。如果系统成功恢复到前一版本, 那么这将成为当前的现行版本。如果该过程无法完成, 那么标记此软件产品为毁坏的。清除过程完成之后, 看尝试重新安装此软件。

请求者 (requester) : 显示站或请求程序运行的交互式通信会话。

请求 (request) : (1) 借助于基本传输单元, 以某种访问方法产生的伪指令, 它使网络控制程序执行一次数据传送操作或一次辅助操作。(2) 在 SNA 中的一种消息单元, 它发信号通知一个操作或协议的协议。(3) 在“增强的 X-Windows”中的一条命令, 让服务器在一个连接上发送一个数据块。(4) 在文本格式中的一种请求, 它是格式化器可识别的一条内置命令。

求值 (evaluation) : 在 Ada 语言中, 表达式的求值是计算表达式值的过程。这个过程在程序执行期间发生。

驱动程序 (driver) : 一个最接近外部接口的流的末段。驱动程序的主要功能是处理在外部设备和流之间的信息和任何相关联的设备和正在传输的数据之间的关系。它也可以是虚伪驱动程序, 不直接与设备关联, 它执行内部流的功能, 比如多路复用器或日志驱动程序。

驱动器表 (driving table) : 描述 nroff 命令的所有打印机特定的信息。

区分大小写 (case-sensitive) : 能够区分大写和小写字母。

区分符 (diacritic) : 区分符是添加到信中表明特殊的语音值的标记。区分符是由非空字符序列组成的, 该序列由两个键值组成, 第一个键值为 13 个区分符, 其后紧跟一个字母字符或空格。终端设备驱动程序将这个两键值的序列转换为一个代码点。

区域 (Region) : 像素位置的任意集合。Xlib 库为操作区域提供子例程。不透明类型 **Region** 在 `<X11/Xutil.h>` 头文件中被定义。

区域 (region) : 位图、像素图、屏幕或窗口中的某个区域。

取幂 (exponentiation) : 把一个值提升到某次幂的运算。

取消分配 (deallocation) : 一个除去客户机资源使用权限的操作。

取消连接的模式 (disconnected mode, DM) : 在 SDLC 中来自辅站表明其已经断开连接并希望联机的响应。

取消连接 (disconnect) : X.25 通信中从 X.25 网络取消对端口的连接。

取消选择 (deselect) : 取消对按钮的选择。如果用鼠标, 您可以用“选择”(左)键取消选择突出显示的区域。否则, 您可以使用键盘上的“选择”键。要取消缺省按钮的选择, 请在选择列表中选择另一个按钮。

取消 (Cancel) : CDE 中一个除去窗口 (不应用窗口中所作的任何更改) 的按钮。

取消 (cancel) : 在任务完成前结束任务。

全点解析 (APA) 显示器 (All Points Addressable (APA) displa) : 允许对每个像素 (pel) 都可单独寻址并显示的显示器。APA 显示器允许显示未在字符框中预定义的图像。对照字符显示器 (*character display*)。另见位图显示器 (*bitmapped display*)。

全高 (full high) : 指具有原始高度 (82.5 mm 或 3.25 in.), 宽度 (146 mm 或 5.75 in.) 和长度 (203 mm 或 8.0 in.) 的存储设备。请参阅半高 (*half high*)。

全局变量 (global variable) : 在一个程序模块中定义, 但用于其它独立汇编的程序模块的符号。

全局前向替换 (global forward substitution) : 一种优化方法, 通过该方法, 赋值结果可以通过程序前向传播。例如, 赋值 $B=C$ 后跟 $A=B$ 可由 $A=C$ 替代, 前提是 B 不会用于程序中的其它位置。

全局区域 (global area) : (1) 用来在两个或多个主程序之间通信的存储区域。(2) 分区中未经初始化的部分, 可在给定时间由该分区中某个任务组的任何程序访问。同样的区域可由在同一分区中运行的其它任务组使用。

全局数据 (global data) : 当处于内核方式时, 可由任何进程寻址的数据, 例如表 (如打开的文件表和进程表), 以及其它由内核维护的数据, 例如缓冲区指针。

全局位置代理 (Global Location Broker, GLB) :
“NCS 位置代理”的一部分。维护关于网络或因特网上对象的全局信息的服务器。

全局字符 (global character) : 可用于文件说明中的用以匹配一个或多个字符的 * 和 ? 特殊字符。例如, 将 ? 置于文件说明中意味着该位置可以是任意字符。

全局 (global) : (1) 在编程语言中, 指语言对象与块之间的关系, 在这种关系中, 语言对象的作用域扩展到该块之外, 但包含在一个包围的块中。(2) 关于计算机程序的一个子分区中定义的信息, 在该程序的至少一个其它分区中使用。(3) 有关可用于一个以上的程序或子例程的信息。

全路径名 (full path name) : 表示成以根目录开始的目录和文件字符串的任何目录和文件名。另见全限定名称 (fully qualified name)、路径名 (path name) 和相对路径名 (relative path name)。

全屏编辑器 (full-screen editor) : 同时显示整个屏幕的编辑器, 仅允许通过移动光标和输入命令来访问及修改数据。

全双工 (full duplex, FDX) : 双工 (duplex) 的同义词。

全限定名称 (fully qualified name) : 一个全限定名称包含其所指的结构成员上的层次序列中的所有名称, 且包含该成员自身的名称。另见全路径名 (full path name)。

全限定主机名 (fully qualified host name) : 一个全限定名称。包含其所指的结构成员上的层次序列中的所有名称, 且包含成员本身。

全新安装 (new installation) : 当您安装 BOS 的硬盘为空时的一种安装方法。如果一个硬盘没有包含任何数据或包含的数据不在同一卷组, 就认为它为空白。

全字 (fullword) : 与字 (word) 同义。

缺省按钮标签 (default button labels) : 在 AIXwindows 中, 当未指定其它的按钮标签时使用 XmLabel 小窗口或小配件。

缺省标签 (default label) : 请参阅标号 (label); 标签。

缺省参数 (default arguments) : 在 C++ 函数原型或声明中, 带缺省值声明的参数。如果调用该函数时省略这些参数, 则使用缺省值。带缺省值的参数必须是函数原型参数列表中的末尾参数。

缺省初始化 (default initialization) : 如果程序员没有指定初始值, 则由编译器分配给数据对象的初始值。C 语言中的外部和静态变量缺省初始化为 0, 但自动和寄存器变量的缺省初始值未定义。另见缺省值 (default value)。

缺省打印机 (default printer) : 接受来自分配给它的显示站的所有打印输出的打印机。

缺省单元 (全局访问) (default cell, global access) : 允许从网络中的任何节点访问的 NCS 单元。这是 License Use Management 配置中最常使用的单元, 因为它允许缺省单元中的其它 License Use Management 服务器自由通信。

缺省构造函数 (default constructor) : 一个不带参数的 C++ 构造函数, 或者如果带任何参数, 则所有参数都具有缺省值。

缺省加速键 (default accelerators) : 请参阅加速键 (accelerator)。

缺省目录 (default directory) : 由操作系统提供的目录名, 如果未指定。

缺省驱动器 (default drive) : 由操作系统提供的驱动器名, 如果未指定。

缺省设备 (default device) : 当操作员没有指定备用设备时所使用的连接到计算机的设备 (如打印机或磁盘驱动器)。

缺省文件 (default files) : 资源缺省值以 ASCII 格式存储的数据文件, 允许运行时无需重写或重新编译源代码就可对备用资源值赋值。

缺省文件 (defaults file) : 请参阅缺省文件 (default files)。

缺省值 (default value) : 存储在系统中, 在没有指定其它值时使用的值。另见缺省值 (default) 和缺省初始化 (default initialization)。

缺省值 (default) : 当用户没有指定替代值时假定的值、属性和选项。另见缺省值 (default value)。

缺省 shell (default shell) : 在 AIXwindows 中, 当未指定其它 shell 属性时使用的 shell。

缺陷 (bug) : 程序错误或程序意图的逻辑问题。

缺页故障 (page fault) : (1) 当活动页面引用不在内存中的页面时发生的程序中断。(2) 当处理器试图访问不在实际内存中的虚拟内存页是发生的中断。

确认超时 (acknowledge timeout) : 自数据发送之后, 工作站应该等待来自远程工作站的确认的秒数。

确认字符 (ACK) (acknowledgment character, ACK) : 在二进制同步通信中, 传输控制字符以对数据传输的肯定响应方式发送。

确认 (acknowledge) : (1) 应答、对轮询、地址或消息的响应。(2) 在 X.25 API 中, 用来确认具有 D 位集的数据包已经到达。

确认 (CONFIRM) : 一个请求远程事务程序告知最后的传输是否成功接收的请求。

确认 (confirmation) : 一个允许发送方继续的接收方传输。

群集文件 (clustered file) : 相邻键值记录物理上群集在一起的文件。群集文件组织通过文件中记录的主索引优化顺序存取。

群集 (cluster) : (1) 用于资源共享 (例如, 局域网、主机连接工作站等等) 用途的任意工作站互联配置。(2) 一次分配的一组存储位置。(3) 一个由控制单元 (群集控制器) 和与其相连的工作站组成的站。(4) 一个由 mbuf 管理设施提供给通信软件各层的页大小 (4096 字节) 缓冲区。(也称为 “群集 mbuf”、“mbuf 群集” 和 “映射页面”。)

[R]

热点 (hotspot) : 与报告给指针的坐标对应的光标所关联的点。一个光标具有一个关联的热点, 该热点定义与报告给指针的坐标对应的光标处的点。

热输出 (thermal output) : 计算机系统置于建筑物散热系统上的热负载, 以 “英国热量单位” (British Thermal Units, BTU) 度量。

任务标识 (task ID) : Ada 语言中任务的字母标号或标识。此标号由调试器的 TASKS 选项确定。任务标识指定给每个未终止的任务。

任务分配法 (tasking) : 多任务方法 (multitasking) 的同义词。

任务库 (Task Library) : 一种 C++ 类库, 提供工具以编写由任务组成的程序。

任务 (task) : (1) 要执行的基本工作单元。示例包含用户任务、服务器任务和处理器任务。(2) 一个进程和运行此进程的过程。(3) 在多道编程或多重处理环境中, 一个或多个指令序列, 控制程序将它当作计算机要完成的工作的一个元素。(4) Ada 语言中, 和程序的其它部分并行操作。它写成一个任务规范 (指定任务的名称和任务实体的名称与其条目的形式参数) 以及定义任务执行的任务主体。任务单元是各种程序单元之一。任务类型允许与任何此类型相似的任务的后续声明。任务类型的值用于指定任务。(5) C++ 任务是一个可用于激发程序操作的轻量级的、非优先的例程。任务非优先是因为在任一时间只能执行单个任务。任务轻量级是因为这比真实操作系统进程需要更少的时间和空间来创建一个任务。

认证 (authentication) : 当执行命令 **login** 或 **su** 时验证用户身份。例如, 操作系统的认证方法包含将用户输入的密码与事先为该用户定义的加密密码相对照。可添加次要的认证方法做额外检查, 例如验证针对网络的用户身份。

日本工业标准 (Japanese Industry Standard, JIS) : 编码字符集的标准。

日本 Shift-In 启动定界符 (Japanese Shift-In start delimiter) : 在 SNA Server 中, 由 X.21 物理链接控制所支持的可选功能部件。

日历视图 (Calendar view) : CDE 中显示在主日历 (Calendar) 窗口的日、周、月或年。

日历 (Calendar) : CDE 中使您能够调度约会并创建任务列表 (To Do List) 的软件应用程序。

日文 Shift-JIS 编码 (Shift-Japanese Industrial Standard, SJIS) : 由用于字符编码的单字节与双字节构成的编码方案。因为日语和亚洲其它语言有大量的字符, 8 位字节对字符编码而言是不够的。

日志记录 (journaling) : (1) 在日志中, 由物理文件成员中改变的记录进程。(2) 在数据库中顺序记录信息的进程。

日志记录 (log record) : 日志数据被格式化为的预先定义的数据的结构。记录拥有特定的大小和格式, 并包含特定的相关信息集。日志记录可由它的日志序列号 (LSN) 识别。

日志卷 (log volume) : 磁盘空间的抽象表示, 用于 Encina 日志服务器的存储。日志卷的两种类型: 永久卷和归档卷。永久卷可以存储在文件或磁盘设备中。归档卷只可以存储在文件设备中。内部日志服务器数据和日志文件组必须存储在永久卷中。日志归档组必须存储在归档卷中。另见卷 (volume)、逻辑卷 (logical volume) 和物理卷 (physical volume)。

日志强制组 (log force groups) : 一个关于不同进程的日志记录的逻辑关联。对这些与由一个具体事务产生的所有进程相关的日志记录进行分组, 是用一种简单的方式表达式或操作, 为确保所有与进程相关的代表一个特定的事务的日志记录能够被涉及提供了一种便利的方式。

日志强制 (log force) : 一个引起将所有待写的记录写到永久存储器上的操作。日志强制通常与提交一个事务有关, 它确保与事务有关的日志记录在日志中实际存在。在进行日志强制之前, 这些记录可能仅被存储在内存中, 因此, 容易受系统故障的损坏。

日志文件系统 (journaled file system) : 操作系统所使用的数据库文件标准顺序结构。

日志文件 (log file) : (1) 记录来自许可证服务器、有时候来自许可的产品的消息和错误的文本文件，它驻留在 `/usr/lib/netls/conf` 目录中。(2) 在 Ada 语言中，一个包含命令和注释以及调试器响应的记录的文件。该日志可以被用于以后的分析、记录程序行为或程序修改后的比较。

日志 (log) : (1) 记录。例如，记录系统打印机上的所有消息。(2) 一个消息的列表，比如一个错误日志。(3) 为了记帐或数据收集用途放在一个辅助存储设备上的消息或消息片段的集合。

容纳 (containment) : 一个指针是“被包含的”如果该指针位于窗口中，但不是位于窗口的下级，并且光标热点处于可见窗口或其下级之一的可视区域内。窗口边框视为窗口的一部分。

容器 (container) : CDE 中用来容纳对象或数据的控件。文件夹和笔记本都是容器的示例。在邮件程序中指包含所有邮件消息的电子邮箱和文件系统。消息一旦放入容器就可对它进行显示、修改、删除、打印、包含、转发及回复。

入口点向量 (entry point vector, EPV) : 一条记录，其字段是指向过程的指针，该过程实现了由接口定义的操作。

入口点向量 (EPV) : 请参阅入口点向量 (*entry point vector*) 或入口点向量 (*Entry Point Vector*)。

入口点 (entry poin) : (1) 第一条指令的地址或标号、用来输入计算机程序、例程或子例程。一个程序有几个不同的入口点，每一个对应一个不同的功能或用途。(2) 在一个例程中，控制可以通过任何位置。

软件安装 (software installation) : 将软件从外部介质恢复到本地文件系统的过程。软件在其就绪使用之前可以要求进一步的处理或配置。

软件包封装 (package closure) : 设置需要支持可以创建在软件包中的所有类定义的所有可能的实例的类。

软件包 (package) : (1) 一个可安装的软件产品单元。软件产品包是可以从软件产品的其它软件包独立操作的独立的可安装单元。(2) 在 Ada 语言中，指定一组逻辑相关项，比如类型、此类型的对象、以及带此类型参数的子程序。它作为软件包声明和软件包主体来写。软件包声明有一个包含在软件包外可被显式地使用的所有项的声明的可见部分。它可能也含有包含完成可见项的明确说明的结构的详细信息的私有部分，但是这跟软件包的用户无关。软件包主体包含已在软件包声明中指定的子程序（可能是别的软件包的任务）的实现。软件包是一种程序单元。另见专用部件 (*private part*)。

软件重要产品数据 (Software Vital Product Data, SWVPD) : 唯一定义系统、硬件、软件和处理系统微码元素的信息。

软键盘映射 (software keyboard map) : 将击键映射至字符或预先定义的功能（比如制表符）的表。

软键盘 (software keyboard) : 将原始击键映射至显示符号、预先定义的功能或字符串的表。与操作系统一起提供的软键盘与以下语言相关：美国英语、英国英语、丹麦语、日语等等。

软件配置 (software configuration) : 使已安装软件就绪使用所需的处理。

软件 (software) : 程序、过程、规则以及任何属于系统操作相关的文档编制。对照硬件 (*hardware*)。

软盘驱动器 (diskette drive) : 用来读写软盘上信息的设备。

软盘 (diskette) : 一种薄而软的磁性盘片，它被永久地封装在保护性封套中。它可以用来存储来自磁盘或其它软盘的信息副本。

闰秒 (leap seconds) : 一个对 UTC 的不经常的调整，以说明地球转动的不规则性。

[S]

三角数组 (triangular array) : 一个二维数组，其中只能在由第一个和最后一个元素描绘的假定轴的一边找到活动元素。

散列表 (hash table) : 经由缩短的搜索键（散列值）而访问的信息表。使用散列表使平均搜索时间最短。

散列代码 (hash code) : 一种格式，其中数据以压缩格式存储。

散列特征符 (hash signature) : 由混编一个字符串而产生的固定长度的位串。可以通过比较字符串的散列特征符而迅速比较这些字符串。

散列 (hashing) : (1) 将搜索键变换成一个地址以存储和检索数据项的方法。(2) 将字符串编码成固定长度的位串以进行比较。编码不一定是唯一的。

散射 (scatter) : 对于输入和输出操作，指从设备读取数据并将它定位至非邻接地址。对照聚集 (*gather*)。

扫描代码 (scan code) : 自键盘的原始输入。

扫描线顺序 (scanline order) : 按递增的 y 轴坐标顺序排列的扫描线表示的图像。

扫描线 (scanline)：(1) 以水平查看的像素或位值的列表（所有的值具有相同的 *y* 轴坐标）。这些值按递增的 *x* 轴坐标排列。作为图像的一部分，下一条扫描线按递增的 *y* 轴坐标排列。(2) 通过阴极射线管电子光束的水平扫描在显示器上产生的可视线。

扫描转换 (scan conversion)：从一个应用程序将像素信息生成至帧缓冲区的进程。

扫描 (scan)：(1) 有顺序地，一个部分一个部分地检查。(2) 查找特定字符串或语法错误的记录。

扫视 (panning)：(1) 在计算机图形学中，当图象太大而不能显示在单个屏幕中时通过移动图象的一部分到另一部分来查看图象。(2) 逐步地转换完整的显示图象来给出图象的侧面移动的视觉印象。

色彩图方式 (color map mode)：一种通过颜色查找表（色彩图）传递存储在帧缓冲区中的值以其获得用于显示的红色、绿色和蓝色值的硬件配置。色彩图中的项称为颜色索引。在色彩图方式中，存储在帧缓冲区中值当作色彩图索引。另见 *RGB 方式 (RGB mode)*。

色彩图 (color map)：(1) 计算机图形学中一种查找表，其中每个索引与一个红色、绿色和蓝色值相关联。与颜色查找表 (*color lookup table*)、调色板 (*color palette*) 和颜色表 (*color table*) 同义。(2) “增强的 X-Windows” 中一组颜色单元。像素值对色彩图建立索引以生成 RGB 值强度。色彩图由一组定义色彩值的项构成（当与窗口相关联时），用于显示窗口的内容。取决于硬件限制，可以一次安装一个或多个色彩图，以便使与这些色彩图相关联的窗口显示正确的颜色。色彩图的两种类型是直接彩色和伪彩色。(3) GL 中一个把颜色索引转换为 RGB 三色值的查找表。查找表是夹在帧缓冲区和数模转换器 (DVA) 之间的“三明治”，存储在帧缓冲区中的颜色索引值转换为 DAC 所要求的红色、绿色和蓝色值。大多数硬件配置中，色彩图深度为 8 位或 12 位，允许同时显示 256 或 4096 种颜色。大多数硬件配置中，DAC 的颜色精度为 8 位，允许用户在 16,777,216 种颜色中选择。

色道 (color ramp)：色彩图中的颜色级数。大多数色道是平滑的，仅有少数不连续的小数字（如果有）。例如，如果把彩虹的全部颜色集装入到色彩图，这样会建立色道。

杀死字符 (kill character)：在一个提示符后输入的删除一行字符的字符。

杀死 (kill)：一个停止进程的操作系统命令。

删除 (delete)：(1) 进行删除。例如，删除一个文件。另见擦除 (*erase*)。(2) C++ 关键字 **delete** 标识了一个自由存储取消分配操作符。在 C++ 中，**delete** 操作符用来破坏由的，**new** 创建的对象。

扇区 (sector)：(1) 在单次读操作或写操作过程中写至磁盘或软盘或从中读取的最小数量的信息。(2) 在磁盘或软盘存储上，指磁道的可寻址子部分，它用于记录一个程序块或数据块。

上行 (upstream)：从驱动程序到流头的方向。

上下文地址 (context address)：一个用斜杠 (/) 括起来的正则表达式。

上下文行 (context line)：在 Performance Toolbox 中，以斜杠和三个点 (/...) 结尾的菜单项。斜杠和三个点表示该行自身代表处于下一分层次级别的列表。对照统计行 (*statistic line*)。另见级联菜单 (*cascading menu*)。

上下文结构 (context structure)：一个指定阴影小窗口界面属性（醒目位置）的有序变量组。

上下文子句 (context clause)：用于 Ada 编程，请参阅编译单元 (*compilation unit*)。

上限电阻 (tie-up resistor)：用于保存高级别的逻辑函数或门的输入的电阻。此寄存器的一端连接至适当的正极电压。

上载 (upload)：将数据从一台计算机传送到另一台。通常，用户从一台小型计算机上传到一台大型计算机。

设备处理程序 (device handler)：直接与硬件通信的设备驱动程序的组件。虚拟设备驱动程序 (*virtual device driver*) 的同义词。

设备定义 (device definition)：定制数据库 (Customized Database) 中关于设备的信息，包括属性和连接位置。

设备管理器 (device manager)：对于复杂的接口来说，它是作为驱动程序和虚拟机之间中介的例程的集合。例如，来自虚拟机的超级用户调用将由设备管理器检查，并路由到合适的从属设备驱动器。

设备号 (device number)：分配到任何外部设备的引用号。

设备节 (device stanza)：定义附加在打印假脱机系统中的队列的设备。设备节包含属于设备（通常为打印机）的所有信息，它位于 */etc/qconfig* 文件中。

设备开关表 (device switch table)：(1) 用作指向设备驱动程序的接口的表。(2) 包含指向每个设备头的条目点的指针的表。

设备类型 (device type)：共享同一型号的一类设备的常规名称；例如 2311、2400 和 2400-1。对照设备类 (*device class*)。

设备类 (device class)：按功能对设备进行分组。例如，一组设备类型的一般名称，例如属于同一个设备类的所有显示站。对照设备类型 (device type)。

设备描述 (device description)：用来给出设备简短描述的文本。举例来说，令牌环适配器的设备描述可以是“令牌环高性能适配器” (Token-Ring High-Performance Adapter)。

设备名称 (device name)：(1) 系统预留的逻辑或符号名，用来引用特定设备。(2) SNA 使用网络适配器的操作系统设备名来获取定义该接口的信息。另见逻辑名 (logical name)。

设备配置数据库 (Device Configuration Database)：存储所有与支持设备配置过程有关的信息。它由预定义数据库 (Predefined Database) 和定制数据库 (Customized Database) 组成。另见预定义数据库 (Predefined Database) 和定制数据库 (Customized Database)。

设备驱动程序 (device driver, DD)：(1) 操作特定设备（如打印机、磁盘驱动器或显示器）的程序。(2) 一个控制 I/O 设备适配器和处理器之间接口的子例程的集合。

设备实例 (device instance)：在定义了设备之后，创建了一条“定制设备对象类” (Customized Devices Object Class) 条目。该条目可看作一个设备实例。系统中定义的每个设备都有一个设备实例。

设备头 (device head)：实现指向设备的应用程序接口的设备驱动程序的组件。

设备位置 (device location)：表明设备的位置路径。它是“定制设备对象类”中的一个字段。

设备状态 (device state)：表明设备实例的当前配置状态。可能的值为 defined (已定义)、available (可用) 和 stopped (已停止)。它是“定制设备对象类”中的一个字段。

设备子类 (device subclass)：用同一个功能类区分设备。它用于表明不同接口。举例来说，打印机类有三个子类：rs232、rs422 和并行。

设备 (device)：(1) 连接到电脑上、为特定用途设计的机械、电力或电子器械，如打印机、绘图仪、或磁盘驱动器。(2) 标量设备、按钮或键盘。按钮的值为 0 或 1 (上或下)；标量设备 (鼠标、拨号盘) 返回一个值的范围的值，键盘则返回 ASCII 值。

社会科学格式 (social science format)：请参阅自然或社会科学格式 (natural or social science format)。

设计环境 (authoring environment)：允许创建超文本文档的组织或设备。

设施扩展 (facilities extension)：在 X.25 API，一个信息包中可选的设施字段的扩展，允许添加进一步的、非 CCITT 指定的、可选的设施信息。

设施 (facilities)：请参阅可选的设施 (optional facilities)。

设置用户标识方式位 (set-user-ID mode bit)：设置文件存取权限时，给要执行的文件的所有者标识设置有效的和保存的用户标识。

设置组标识方式位 (set-group-ID mode bit)：设置文件访问权限时，给要执行的文件的组标识进程设置有效的和保存的组标识。

深度提示 (depth-cueing)：在 3D 计算机图形中，用浓度区分线条的亮度。一般来说，线条上远离眼睛的点比较暗，所以线条看起来好像在远处消失了。

深度 (depth)：(1) 在“增强的 X-Windows”中窗口或像素图的每像素的位数。(2) 在三维上下文中的第二维。

审计跟踪 (audit trail)：审计记录的集合。

审计事件 (audit events)：发生在可能有安全性违例的系统中的情况。这些事件导致写下审计记录。

审计子系统 (auditing subsystem)：一种让管理员检测系统中潜在或实际的安全性违例的机制。该子系统的组件检测系统审计跟踪中的审计事件、日志并收集审计事件及处理审计跟踪。

生成阶段 (generation phase)：排序程序的一部分，将顺序规范转换成机器语言。

生存时间分析 (lifetime analysis)：对变量引用的检查过程，以决定对变量的最终赋值需要存储还是废弃。

升级产品 (upgrade)：修复先前发布的软件产品的缺陷的软件。

升级锁 (upgrade locks)：代替读锁用于宣布还要修改受保护数据的潜在需求的锁。如果应用程序获得相同数据的读锁且可能还有写锁，那么升级锁更不可能死锁。

声明部件 (declarative part)：Ada 语言中的声明序列。它还包含如子程序体和表示子句的相关信息。

声明符 (declarator)：描述数据类型的标识符和可选符号。

声明 (declaration)：(1) 一个使已定义对象对函数或块可用的描述。(2) 编程语言中建立语言对象的机制。声明通常涉及标识符连接和分配给语言对象关注的属性。(3) 在编程语言中，一个影响那种语言中的其它表达式解释的有意义的表达式。(4) 在 Ada 语言中，声明与带实体的标识符

(或一些其它记号)相关联。在某个文本区域内(称为声明的作用域)该关联是有效的。在声明作用域内一些地方,使用标识符引用相关联的声明实体是可能的。在这些地方,标识符称为该实体的简单名称;名称用来表示关联实体。另见声明(*declare*)、简单名称(*simple name*)、作用域(*scope*)和名称(*name*)。

声明 (declare): 一个来自“解释器的解释”菜单的可用选择,允许用户分配变量名和结构定义。另见声明(*declaration*)。

失真 (distortion): 数据通信中可能在传输系统中的两点间发生的波形的非期望改变。失真的六种主要形式为偏离(*bias*)、特性(*characteristic*)、延迟(*delay*)、终止(*end*)、偶发(*fortuitous*)和调和(*harmonic*)。

实操作 (real operation): Ada 浮点或定点值上的任何操作。

实常量 (real constant): 一种十进制位的字符串,它表示一个实数。实常量必须含有一个十进制小数点、一个十进制指数或二者兼有。

实存储器 (real storage): 可直接访问处理器的存储器,可从中运行指令并且指令可从中取数据。

实际参数 (actual parameter): 传递给例程的实际值。对照形式参数(*formal parameter*)。

时间戳记 (time stamp): (1) 通常指自最近一次服务器重新设置以来的时间值,以毫秒来表示。时间戳记通常在 49.7 天后环绕舍入。一旦服务器给定其当前时间,它由 *timestamp T* 表示,而且始终通过将时间戳记空间的一半当作早于 *T*,且一半时间戳记空间晚于 *T* 来解释来客户机的时间戳记。服务器永远不会生成由常量 *CurrentTime* 表示的 *timestamp* 值。保留此值以用于请求显示当前服务器时间。(2) 添加至一个记录或其它形式的数据的信息,此信息显示计算机处理该记录或数据的日期和时间。(3) 描述创建一组许可时的日期和时间的整数。(4) 当上次修改 Ada 语言编译单元时,记录日期和时间。

时间戳 (epoch): 标识属于同一集合的目录副本的时间戳记。

时间分段 (timeslicing): 一种机制,通过此机制在固定间隔先抢占正运行的线程。这能确保每个线程有执行时间。

时间片 (time slice): CPU 调度程序进行的已调度的检查的间隔,这些检查用来确定是否应分派不同的线程。未调度的程序可能由于中断或系统调用而发生。

时间循环 (timing loop): 置于代码周围的循环,其性能被计时,这样时间循环将它范围内的代码执行数量增加到能够提供有意义的性能评估的时间。

十进制常量 (decimal constant): 一个包含 0 到 9 任意数字构成的数。

十进制 (decimal): (1) 属于基数为十的数字系统。十进制数字范围为从 0 到 9。(2) 其特征为选择、选项或条件有 10 种可能的不同值或状态。

实例化 (instantiate): (1) 创建一个实例;复制。(2) 在面向对象编程中,用一个具体的类实例表示一个类抽象。(3) 在 AIXwindows,来创建一般类的一个特定的具体实例。(4) 创建或者生成一个数据类型的特定实例(或者对象)。例如,一个类 *box* 的实例 *box1* 可以由以下声明来实例化: *box box1*;

实例记录 (instance record): 一个特定的小窗口记录,它包含适合于任何作为实例的小窗口身份的数据对象。另见记录(*record*)、小窗口实例(*widget record*)和实例(*instance*)。

实例 (instance): (1) 一个抽象对象类的具体实现。一个小窗口或者配件的实例是一种特定的数据结构,该结构包含在运行时用来在屏幕上生成一个特定的图形对象的具体外观和行为的详细信息。另见小窗口实例(*widget instance*)和实例记录(*instance record*)。对于 Ada 编程,请参阅类属单元(*generic unit*)。(2) 一个面向对象的编程术语;与“对象”同义。一个实例是一个特定的数据类型的实例化。这仅是一个包含一个或者一组值的存储区域。例如,一个类 *box* 已先前定义,类 *box* 的两个实例可以声明: *box box1*、*box box2*; 来实例化。(3) 在 XOM 中,由类来表示的类别对象。

十六进制常数 (hexadecimal constant): (1) 字符 0x 或 0X (零 X),后跟有 0 到 9 中的任何数字和 A 到 F 中的大写或小写字母。(2) 一个常数,通常以特殊字符开始,仅包含十六进制数字。

十六进制 (hex): 请参阅十六进制(*hexadecimal*)。

十六进制 (hexadecimal, hex): 关于基数为十六的计数制;范围从 0 (零)到 9 (九)以及 A (十)到 F (十五)。

实内存 (real memory): 任何系统上活动的物理内存。对照虚拟内存(*virtual memory*)。

拾取范围 (picking region): 光标周围对拾取事件敏感的光标一个矩形卷。如果在该卷内画一个图元图形,则报告一个拾取事件。该区域的宽度和高度可由用户设置。如果允许使用 *z* 缓冲区,则该区的深度为整个 *z* 缓冲区。另见命中(*hit*)、选择(*selecting*)、拾取(*picking*)和选择范围(*selecting region*)。

拾取 (picking)：在计算机图形学中，查找在显示屏幕上靠近光标处绘制的图元的方法。另见命中 (*hit*)、拾取范围 (*picking region*)、选择 (*selecting*) 和选择范围 (*selecting region*)。

实设备 (real device)：实际设备硬件。

实时控制微码 (real-time control microcode, RCM)：在 X.25 通信中，在 X.25 接口协处理器 / 2 上运行以提供控制功能的微码。

实时系统 (real-time system)：接收和处理数据的一种系统，因此数据或结果对立即使用是可用的。

实时 (real time)：(1) 装入程序及其完成之间经过的时间。(2) 计算机与机外另一过程相连，按照机外过程对时间的要求进行有关的数据处理。(3) 用于描述会话式的操作方式的系统和过程的进行要受到人为的干预的过程。(4) 指一种应用程序，如一种过程控制系统或一种计算机辅助教学系统，能以足够快的速度响应外部对计算机的输入，并足以影响其后的输入。

实数类型 (real type)：(1) 一种算术数据类型，它能约计实数的值。(2) 在 Ada 语言中，实数类型是其值代表实数近似值的一种类型。有两种实数类型：定点类型由绝对误差界限指定；浮点类型由相对误差界限指定，表达为有意义的十进制位的一个数。

实数 (real number)：(1) 一种含有十进制小数点，并以定点或浮点格式存储的数。(2) 在固定基数计数系统中，可以用有限或无限位数字表示的一种数。

实页号 (real page number)：转化控制字 (TWC) 中的此字段包含实页地址，在系统内存中总线地址映射到该实页地址。

实用程序 (utility)：(1) 一种服务。编程中，执行公共服务功能的程序。(2) 系统、程序或设备执行其已设计功能的能力。

实用户标识 (real user ID)：(1) 标识最后一次使用 **login** 命令或 **su** 命令时被认证的用户。(2) 对于每个用户，在 **/etc/passwd** 文件中指定的用户标识。

时钟 (Clock)：CDE 中一个显示本地时间的前面板控件。

时钟 (clock, CLK)：(1) 一个周期性生成用于同步的信号的设备。(2) 数据通信中提供时间基准的设备，用于传输系统中控制某些功能的定时（如采样）及控制信号元素的持续时间。

实组标识 (real group ID)：对于每个用户，在密码文件中定义的组标识。

释放 (deallocate)：释放分配给特定任务的资源。对照分配 (*allocate*)。

适合宽度 (Wrap To Fit)：在 CDE 的“文本编辑器”中，此设置在您输入文本时控制各行是否自动换行以适合窗口宽度。如果这样设置，各行在窗口边缘换行。如果未设置，您必须按“返回”来结束这行。

事件队列 (event queue)：计算机图形学中，记录输入设备 — 按钮、标量设备和键盘的变化的队列。事件队列提供了输入事件的按时间排序列表。

事件控制位 (event control bit, ECB)：分配给每个队列以标志元素的到来和离开的一位。

事件类 (event class)：分配给一组与一个特定主题或系统部件相关联的跟踪点的一个数字。已定义的事件类在跟踪配置文件中列出。

事件循环 (event loop)：一个循环执行以完成某项任务的步骤序列。它必须包含至少一个输入（源）和一个终点。

事件掩码 (event mask)：计算机图形学中，用户相对于窗口所请求的事件类型的集合。

事件 (event)：(1) 元素的人队或出队。(2) 对某项任务有意义的一次发生。(3) 在计算机图形学中，从设备异步产生的信息或者作为客户机请求的副作用产生的信息。事件按照类型分组，除非客户机已对该类型的信息发出一个特定请求，否则事件不会由服务器发送给客户机。事件通常相对于窗口进行报告。另见回调 (*callback*)。

世界坐标 (world coordinates)：GL 中用户定义的、其中描述图像的坐标系。模型化命令用于定位自然空间中的图元。取景和投影变换定义屏幕空间的自然空间的映射。与自然空间 (*world space*) 同义。另见 模拟坐标 (*modeling coordinates*)、视觉坐标 (*eye coordinates*)、图元坐标 (*primitive coordinates*)、屏幕坐标 (*screen coordinates*)、查看矩阵 (*viewing matrix*) 和变换 (*transformation*)

自然空间 (world space)：世界坐标 (*world coordinates*) 的同义词。

视觉空间 (eye space)：视觉坐标 (*eye coordinates*) 的同义词。

视觉坐标 (eye coordinates)：在 GL 中，查看器的眼睛位于坐标系原点，因此所有测量的距离都参照视觉。查看从世界坐标到视觉坐标的变换映射，和从视觉坐标到规格化设备坐标的投影变换映射。与 视觉空间 (*eye space*)、查看坐标 (*viewing coordinates*) 和查看器坐标 (*viewer coordinates*) 同义。另见模拟坐标 (*modeling*

coordinates)、图元坐标 (*primitive coordinates*)、世界坐标 (*World coordinates*)、屏幕坐标 (*screen coordinates*) 和变换 (*transformation*)。

视口 (viewport)：GL 中从规格化设备坐标到设备坐标的映射。视口将单元立方体 $x/w = +/-1$, $y/w = +/-1$, $z/w = +/-1$ 映射至屏幕空间，以像素测量。视口是图形流水线的最后一次变换。视口可以小于或大于窗口，也可以小于或大于屏幕屏蔽区，虽然在大多数应用程序中，它们是相同大小的。

示忙灯 (busy light)：CDE 中，当“前面板”操作已调用但在窗口出现之前或重新装入操作时，闪烁的控件。

适配器代码 (adapter code)：X.25 通信中，“X.25 接口协处理器 / 2 协议代码”控制帧 (*frame-level*) 和包级别 (*packet-level*) 的通信处理。

适配器 (adapter)：(1) 用于连接两个不同部件或机器的机械装置。(2) 一印刷电路卡修改系统元件以允许它以特定方式操作。另见通信适配器 (*communications adapter*) 和卡 (*card*)。

视频查找表 (video lookup table, VLT)：硬件中实现的色彩图。

试探法 (heuristics)：系统管理员使用的准则，用来干预会在何处另行发生提交或放弃这两种状态的故障。

事务标识 (transaction ID)：请参阅事务标识 (*transaction identifier*)。

事务标识 (transaction identifier)：指定给每个事务的唯一标识，用于标识与该事务相关的所有操作。

事务处理程序名 (transaction program name, TPN)：使用数据通信来发送数据至另一个应用程序或从另一个应用程序接收数据的应用程序的名称。

事务处理程序 (transaction program)：在 SNA 网络中处理事务的程序。事务处理程序的两种类型是应用程序事务处理程序和服务事务处理程序。另见转换 (*conversion*)、应用程序事务处理程序 (*application transaction program*) 和服务事务处理程序 (*service transaction program*)。

事务处理 RPC (transactional RPC)：事务处理远程过程调用。

事务系列 (transaction family)：具有共同的属于同一个事务系列的祖先的嵌套事务。一个事务系列的所有成员一起提交并同时断开锁定。

事务 (transaction)；交易：(1) 工作站与程序之间，两个工作站之间或两个程序之间完成特定操作或达到特定结

果的交换。例如客户存款的输入和客户结算的更新。(2) 批处理或远程批处理条目中的一个作业或作业步骤。

视域 (field of view)：在 GL 中，在视图下的区域范围。视域由正在使用的查看矩阵 (*viewing matrix*) 定义。

收费请求服务 (charging requesting service)：X.25 通信中，一个指定收费信息 (段计数数据、货币单位数据或呼叫持续时间数据) 是必须的可选设施。

收件箱 (Inbox)：在 CDE 中，主或缺省“邮件程序”容器。收件箱接收所有进入邮件。消息可从收件箱中移出并放入各种容器中。

首标 (header label, HDR)：软盘上的一组特殊记录，描述该软盘的内容。

手持光标器 (puck)：用来在图形输入板上选择一个特定位置的设备。

手动拨号 (manual dialing)：在进行一个 ATE 连接时，在电话线路上手动拨号。

手动呼叫 (manual call)：在数据通信时，一个要求运算符的线路类型在交换线路上呼出呼叫。对照自动呼叫 (*auto-call*)。

手工调整 (hand tuning)：手工修改源代码以改进最终目标代码的过程。手工调整是优化的一个方面。

守护程序进程 (daemon process)：一个由 root 用户或 root shell 开始，并且只能由 root 用户停止的进程。守护程序进程通常提供任何时候都必须可用的服务，如发送数据到打印机。

守护程序 (daemon)：一个自动运行并执行标准服务的程序。有些守护程序自动地触发执行它们的任务；另外一些定期运行。一个示例就是 **cron** 守护程序，它定期执行列在 **/var/spool/cron/crontabs** 目录中的任务。与守护程序 (*demon*) 同义。

守护程序 (demon)：*daemon* (守护程序) 的同义词。

首选表现方式 (primary representation)：服务提供属性值给客户机的形式。

首页 (header page)：置于打印文件或打印作业前面的分隔页。

受保护的 (protected)：C++ 类的一个受保护成员仅对该类的成员函数和友元，或该类派生的类的成员函数和友元是可访问的。

受管窗口 (managed window)：请参阅受管子类 (*managed children*)。

受管子类 (managed children): 在“增强的 X-Windows”中, 这个子类中可控制字段含有值真可以改变布局 (几何) 以使他们重新配置和调整大小。

受叫地址扩展 (called address extension): 请参阅地址扩展 (address extension)。

受叫方 (called): X.25 通信中, 应用于受叫位置或用户的 adjective。

受叫子地址 (called subaddress): 请参阅子地址 (subaddress)。

受叫 DLS 用户 (called DLS user): 处于连接方式的数据链路服务 (DLS) 用户, 处理来自其它 DLS 用户的连接请求。

受叫 NS 用户 (called NS user): 一主叫 NS 用户要与之建立网络连接的网路服务 (NS)。

授权区 (zone of authority): 一组由单个命名服务器管理的名称。

授权协议 (authorization protocol): 考虑到受保护对象, 一种用于建立主体授权的正式过程。

授权 (authorization): (1) 考虑到受保护对象, 对主体权限的确定。(2) 考虑到受保护对象, 主体寻求的对权限的批准。

授权 (authorize): (1) 授权用户与计算机系统或显示站通信或使用计算机系统或显示站的权限。(2) 给用户对对象、资源或功能以完全或限定的访问。

受限接口 (limited interface): 一个提供有限功能接口的系统调用的集合。另见接口 (interface) 和扩展接口 (extended interface)。

受限类型 (limited type): 在 Ada 语言中的一个类型, 该类型既没有赋值, 也没有进行预定义的相等性比较等隐式声明。所有任务类型都被限制。一个私有类型可以被定义为限制的。一个相等操作符可以被显式地声明为一个受限的类型。

受限 shell (restricted shell): 对指定用户提供受控的、受限访问的 shell。

输出重定向 (output redirection): 输出目标的规范不同于标准的。

输出处理程序 (output handler): 负责由过程或子过程产生的分布式数据的程序模块。

输出单元 (output unit): 输出设备 (output device) 的同义词。

输出方式 (output mode): 打开记录可以写入文件的打开方式。

输出缓冲区 (output buffer): 在“增强的 X-Windows”中, 用 Xlib 库使用区域来存储请求。

输出列表 (output list): 把值写入文件或设备的变量列表。

输出流 (output stream): 由操作系统或处理程序在输出设备上显示的消息和其它输出数据。

输出设备 (output device): 一种物理设备, 计算机使用它显示数据给用户。与输出单元 (output unit) 同义。

输出文件 (output file): (1) 程序打开一文件这样它就可以写入这个文件。(2) 包含处理结果的文件。

输出总线 (bus-out): 一种不定向数据总线, 系统 360/370 并行 I/O 接口的一部分。该总线把数据从主机传到系统部件。

输出 (output): (1) 处理数据的结果。(2) 属于功能单元或在输出过程中涉及的通道, 或数据或涉及的这个过程。(3) 数据从存储器传输到输出设备。(4) 在 Pascal 中, 预定义的标准文件定义。

输出 (render): 根据描述情景的数据在可视屏幕上创建一个图像。

书目 (bibliography): 文档内部引用的文档列表。对于每个文档, 通常会列出以下内容: 作者、文档、文档注释和 (有时) 有关作者注释。

输入板原点 (tablet origin): 输入板上的一点, 输入板上所有其它位置与其对应。原点位于输入板的左下角或中间。

输入板 (tablet): 具有用于指明其上的位置机制的特殊平面。输入板通常用作定位器。

输入重定向 (input redirection): 指定不是标准输入源的输入源。

输入方式 (input mode): 文件中的记录可以读取的打开方式。

输入管理器 (input manager): 在图形环境中, 控制键盘输入的客户机, 通常是窗口管理器的一部分。

输入焦点 (input focus): 在图形环境中, 一个定义处理键盘输入范围的窗口。缺省的, 键盘事件被发送到使用指针所在的窗口的客户机。也可能将键盘输入连接到一个特定的窗口。事件随后被发送到合适的客户机, 而不需要考虑指针的位置。与焦点窗口 (focus window)。请参阅非活动的 (inactive)。

输入列表 (input list) : 从输入数据赋值的变量列表。

输入流 (input stream) : 通过输入设备提供给系统的操作控制语句和数据序列。

输入设备 (input device) : (1) 一个为计算机提供数据的物理设备。(2) 该设备是正在安装的软件的源。该输入设备可以是一个磁带机、CD-ROM 驱动器、DVD 驱动器、软盘驱动器, 或者是一个目录。

输入数据类型 (input data type) : 包含在输入文件中的数据类型。示例: ASCII 和 PostScript。

输入文件 (input file) : 为读取记录打开的文件。

输入语义学 (input semantics) : 用户必须输入指定的顺序和格式的输入。

输入字段 (input field) : (1) 在其中可输入数据的显示文件的一个区域。(2) 在计算机图形中, 在一个可以输入, 修改, 或者擦除的显示界面上的一个非保护字段。

输入总线 (bus-in) : 一种不定向数据总线, 系统 360/370 并行 I/O 接口的一部分。该总线把数据从系统部件传到主机。

输入 / 输出配置程序 (input/output configuration program, IOCP) : 定义处理器群需要的用来控制 I/O 请求的 I/O 配置数据的程序。

输入 / 输出配置数据集 (Input/output configuration data set, IOCDs) : 一个由 I/O 配置程序 (IOCP) 构建的配置定义, 存储在与处理器控制器相关联的磁盘文件中。

输入 / 输出设备号 (input/output device number) : 通过客户操作系统分配给设备驱动程序或者通过虚拟资源管理器分配给虚拟设备的一个值。该号码唯一标识设备, 而不管它是真实的还是虚拟的。

输入 / 输出通道控制器 (input/output channel controller, IOCC) : 用来监控输入 / 输出总线和处理器之间通信的硬件组件。

输入 / 输出通道 (input/output channel, IO channel) : (1) 在数据处理系统中, 用来处理内部和外围设备之间的数据传送的功能单元。(2) 在计算机系统中由处理器控制的功能单元, 它处理处理器存储器和本地外围设备之间的数据传送。

输入 / 输出文件 (input/output file) : 为了输入和输出用途打开的文件。

输入 / 输出子系统 (input/output subsystem) : 由进程提供为数据传输和 I/O 设备管理和控制的机制的设备管理器组成的操作系统部件。

输入 / 输出 (input/output, I/O) : (1) 指输入、输出或者在计算机和设备间的输入和输出。(2) 指一个设备, 该设备的部件在同一个时间可以执行一个输入过程和一个输出过程。(3) 与输入过程、输出过程或两者并行或不并行的功能性单元或通道和与这一过程有关的数据。

输入 (input) : (1) 要进行处理数据。(2) 在 Pascal 中, 一种预定义的标准文件定义。

鼠标按键 1 (mouse button 1) : 在鼠标上, 配置为配置为右手使用时最左面的按键, 配置为左手使用时最右面的按键。鼠标按键 1 主要用于选取, 它是“单击”操作的缺省按键。

鼠标按键 2 (mouse button 2) : 在一个三键鼠标上, 中间的按键。在一个两键鼠标上, 配置为右手使用时最右面的按键, 配置为左手使用时最左面的按键。鼠标按键 2 (Mouse button 2) 有两种模式: 传输和调整。

鼠标按键 3 (mouse button 3) : 在鼠标上, 配置为右手使用时最右面的按键, 配置为左手使用时最左面的按键。鼠标按键 3 激活弹出菜单。

鼠标比例 (mouse scaling) : 光标相对于鼠标移动所移动的距离。比例因子可以是 1:1 或 2:1。

鼠标阈值 (mouse threshold) : 一个操作系统参数, 它决定屏幕光标移动所要的鼠标水平或垂直运动数量。

鼠标 (mouse) : 一个手持定位器, 用户可以通过在平面上移动它进行操作。它允许用户通过按键选择对象和滚动显示屏幕。

数据包 (data packet) : 在 X.25 通信中, 一用于在 DTE/DCE 接口的虚电路上传输用户数据的信息包。另见包 (packet); 信息包; 分组。

数据报协议 (datagram protocol) : 一种无连接的, 基于数据报的传送协议, 如 UDP; 运行在无连接传送协议之上的 RPC 协议。

数据报 (datagram) : (1) 在报文分组交换中, 一种不依赖于其它分组的独立分组, 无需依赖于 DTE 和网络间的早先交换, 自身带有的信息足够进行从源数据终端设备 (DTE) 到目的 DTE 的路由。(2) 在 DCE 中, 独立于所有其它包的不可靠网络数据包, 缺乏任何递送或顺序的保证。

数据传送 (data transfer) : 从一个位置到在另一个位置数据的数据移动或复制。

数据存储中断 (data storage interrupt) : 当访问存储器或 I/O 空间出错时发出的中断。一个典型的数据存储中断是缺页故障或保护违例。

数据电路终端设备 (data-circuit-terminating equipment, DCE)： 数据站中安装在用户前端的设备，提供数据终端设备 (DTE) 和线路之间建立、维护和终止连接及信号转换和编码所需的所有功能。

数据电路 (data circuit)： 相关联的发送和接收线路对，提供一种双向数据通信的方法。

数据定义 (data definition)： 描述数据特征、指定关系或建立上下文的程序语句。数据定义还可以提供初始值。定义出现在函数外部或块语句的开头。

数据对象 (data object)： 通过一个单一名称来引用的数据集合。另见对象 (*object*)。

数据分解 (unmarshal)： NCS 中从“远程过程调用”包中复制数据。stub 执行数据分解。对照数据编组 (*marshal*)。

数据高速缓存单元 (data cache unit, DCU)： 请参阅高速缓存 (*cache*)。

数据高速缓存 (data cache)： 一个高速缓存，其给处理器提供数据的速度比处理器从 RAM 获得数据更快。

数据加密标准 (Data Encryption Standard, DES)： 一种在美国广泛使用的数据加密算法。

数据库 (database)： 至少由一个文件组成的事实和说明的集合，它完全能满足给定目的。

数据块 (data block)： 请参阅块 (*block*)。

数据类型 (data type)： (1) 在编程语言中，一个值集合加上一个允许的操作的集合。(2) 数据和函数的数学属性和内部表示。(3) 一个用于将数据定义为数字或字符的属性。(4) 数据对象的类型、格式或分类。(5) CDE 中的一种机制，用于将特定数据文件与相应的应用程序和操作相关联。基于文件名约定（如特定扩展名），或文件内容，数据类型可以确定文件类型。另见字符类型 (*character type*) 和类型 (*type*)。

数据链路级别 (data-link level)： 在数据站的层次结构中，高级别逻辑和维护数据链路的控制的数据链路之间控制或处理逻辑的概念性级别。数据链路级别执行如下功能：插入传输位及删除接收位；解释地址和控制字段；生成、传输及解释命令和响应；及计算和解释帧校验序列。与帧级别 (*frame level*) 同义。另见信息包级别 (*packet level*)、物理级别 (*physical level*)。

数据链路控制层 (data link control layer)： 在 SNA 和 X.25 中，由两个节点间的链路调度数据传送及为链路执行错误控制的链路站构成的层。

数据链路控制协议 (data link control protocol)： 在 SNA 中，由数据链路上的两个节点使用的规则集，用以实现信息的有序交换。

数据链路转义符 (data link escape character)： 在 BSC 中，一个通常用于透明文本方式的传输控制字符，表示下一个字符是传输控制字符。

数据链路 (data link)： (1) 两台由链路协议和互连数据电路控制的数据工作站的部件组合，这样可使数据从数据源传输到数据接收器。(2) 两台或更多工作站之间的互连数据电路和链路协议，不包含数据源或数据接收器。(3) 在部件之间通过电信线路交换数据的物理连接和连接协议。另见 X.25 链路 (*X.25 link*)。

数据流 (data stream)： (1) 在一个单独的读或写操作中通过数据通道传输的所有信息（数据和控制信息）。(2) 正在传输或打算传输的持续数据元素流，使用已定义格式的字符或二进制数字格式传输。(3) 使用 **write** 子例程发送到终端设备驱动程序的所有信息。与流 (*stream*) 同义。

数据描述 (data description)： 用于不能自我描述的数据对象，数据对象的组件描述数据，以便其可处理。

数据区域 (data area)： 一个包含特定控制变量（通常在结构或向量中预定义）的内存区域。

数据锁 (data lock)： (1) 确保数据对某个单一应用程序可用，作为对数据记录更新冲突的保护。(2) 将数据段锁进内存的系统锁。

数据提供者 (data supplier)： 在 Performance Toolbox 中，一个提供跨网络的统计信息的程序。与服务器 (*server*) 同义。对照数据消费者 (*data consumer*)。

数据通信 (data communications)： 请参阅通信 (*communications*)。

数据相关性 (data dependency)： 一种一个计算的源操作数是前述计算结果的情况。

数据项 (data item)： 数据处理单元，包含常量、变量、数组元素和字符子串。

数据消费者 (data consumer)： 在 Performance Toolbox 中，对一种程序的描述，该程序通过网络从 **xmservd** 守护程序接收统计信息并打印、后处理或另外使用原始统计信息。与客户机 (*client*) 同义。对照数据提供者 (*data supplier*)。

数据终端就绪 (data terminal ready, DTR)： 一个与 EIA-232 协议一起使用，到达调制解调器的信号。

数据终端设备 (data terminal equipment, DTE)： (1) 数据处理单元中作为数据源、数据接收器或两者的部件。(2) 网络用户。

数据 (组织) (marshal): 在 NCS 中, 拷贝数据到一个远程调用 (RPC) 包。Stubs 执行数据分解。对照数据分解 (unmarshal)。另见 stub。

数据 (data): 一种由人工处理或自动方式, 以适于通信、解释或处理的格式的事实或说明的表示。数据包含常量、变量、数组和字符串。

属性对 (attribute pair): 请参阅属性 (attribute)、值 (value) 和资源值 (resource value)。

属性列表 (property list): 在“增强的 X-Windows”中, 为一个特定窗口定义的属性列表。

属性文件 (attribute file): 系统配置中, 以 stanzas 方式组织的文本文件, 其中每一个都有 stanza 名称和以属性=值对格式给出的属性定义集。配置文件具有属性文件的格式。

属性值 (attribute value): 请参阅属性 (attribute)、值 (value) 和资源值 (resource value)。

属性 (attribute): (1) 一个或多个对象或实体的特征或特性。例如, 一显示字段的属性可为“闪烁”。(2) 在 GL 中, 参数可以影响图元外观。例如, 颜色是一个属性。如果颜色设为“红色”, 那么除非更改它将一直保持红色, 要绘制的任何东西都以红色绘制。其它属性包含线型、线宽、模式和字体。要获得属性和流水线选项列表, 另见流水线选项 (pipeline options)。(3) 设备中, 一已定义或已配置设备的特征。

属性 (property): (1) 在“增强的 X-Windows”中, 与窗口相关的名称、类型、数据格式和数据。通过使用属性, 客户机和窗口管理器共享信息, 诸如调整大小提示、程序名和图标格式。这对客户机来说是一种通用命名机制。协议不解释属性。(2) 和一个窗口相关联的公用信息 (即, 对任何客户机都可用的信息)。

数字补码 (complement of a number): 一个值, 当添加该值到数字时等于一给定值。

数字常量 (numeric constant): 代表整数、实数或复数的常量。

数字错误 (numeric error): 实数或整数的表达式的值超过了这种表达式的编译器的界限 (或编程语言强制的界限) 时的情况。

数字到模拟转换器 (digital-to-analog converter, DAC): (1) 高度集成的芯片, 将从帧缓冲区中的数值转换为监视器所需的快速变化的模拟电压级。(2) 显示子系统的这一部分将像素转换为色彩或灰度。

数字的 (numeric): 与任何 0 到 9 数字有关的。

数字文字 (numeric literal): 数字字符或字符串的值隐含在字符本身中。例如, 777 是文字也是 777 数字的值。数字文字可以包含任何从 0 到 9 的数字, 加减符号和分号。对照 字符文字 (character literal)。

数字字符 (numeric character): 请参阅数字 (digit)。

数字 (digit): (1) 表示非负整数的字符。数字字符 (numeric character) 的同义词。(2) 表示比基数 (radix) 小的一个非零整数的符号。(3) 从 0 到 9 的任意数字。

数组名 (array name): 一组有序数据项的名称。

数组声明符 (array declarator): 描述在程序单元中使用的数组的语句部分。描述包括数组名、维数和每一维的尺寸。

数组填充 (array padding): 在多维数组定义中, 将元素添加到每个数组的优化, 或在用户定义的数组之间添加小组, 以提高数组的内存访问特征的一种优化。

数组元素 (array element): 数组中的单一数据项。

数组 (array): (1) 一包含有序数据元素组的结构。数组中的所有元素具有相同的数据类型。(2) 数据的一维或多维的排列, 如列表、表或项的多维排列。(3) 在编程语言中, 由具有同一属性的数据元素构成的集合, 其中每个数据元素可以通过下标唯一引用。

树结构 (tree structure): 由根段和由根段调用的一个或多个段共同组成的分层调用序列。

树视图 (tree view): CDE 中包含搜索路径中所有低级文件夹的文件夹或文件的视图。

树增量 (tree delta): 具有中继线、发行版和级别指定的更改以及 SID 指定的带有增量的分支的 SCCS 文件。

刷新速率 (refresh rate): 监视器刷新的速率。60 Hz 监视器每秒被重绘 60 次。与垂直回扫 (vertical retrace) 同义。

衰减 (attenuation): (1) 在点与点之间的传输中, 电流、电压或信号功率数量上的减小。可以用分贝或奈培 (neper) 表达。(2) 在三维图形中指随着距离的光强衰减。

双边闭合用户组 (BCUG) (bilateral closed user group, BCUG): 在 X.25 通信中, 一个允许仅在两个指定 DTE 间进行呼叫的可选设施。另见闭合用户组 (closed user group)。

双重突出打印 (double-strike): 两次打印一个字符以创造粗体类型的外观的过程, 常用于击打式打印机。一种较为灵活的双重突出打印形式是强调式打印。另见粗体 (boldface) 和强调字体 (emphasized)。

双点 (dot dot) : 符号 (..) 在相对路径名中表示父目录。

双工连接器 (duplex connector) : 在光纤通道链接环境中, 该组件终止一个机柜中的两根光纤, 并提供连接到双工插座的物理键控。

双工输出 (duplexed output) : 同时使用每张纸的前后两面打印输出。

双工 (duplex) : 属于通信数据, 可以同时发送和接收。与全双工 (*full duplex*) 和 *FDX* 同义。对照半双工 (*half duplex*) 。

双缓冲方式 (double buffer mode) : 在 GL 中, 两个缓冲区交替显示和更新的方式。新的图象可以被画到后台缓冲区, 而前台缓冲区 (包含前一张图象) 则被显示。另见单缓冲方式 (*single buffer mode*) 。

双击 (double-click) : 不移动鼠标指针快速连续单击两次。

双精度 (double precision) : (1) 涉及到使用两个计算机字以所要求的精度来表示数字。(2) 一种规范使得在内部以长 (*long*) 格式存储浮点值。另见精度 (*precision*) 。

双宽打印 (double-wide print) : 字符宽度是正常情况下的两倍的打印格式。

双向通道 (two-way channel) : X.25 通信中运行进入和呼出呼叫的逻辑通道。对照单向通道 (*one-way channel*) 。

双向 (bidirectional) : 请参阅共享端口 (*shared port*) 。

双值运算符 (dyadic operator) : 与二进制运算符 (*binary operator*) 同义。

双字节字符集 (Double-Byte Character Set, DBCS) : 这个字符集的每个字符都用两个字节的存储来表示。如日文、中文和韩文等包含的符号超过了一个字节的表达能力就是用这种字符集来表示的。

水印 (water mark) : 用于流控制的有限值。每个队列具有一个高水印和一个低水印。高水印值指明包含在队列中与字节数相关的上层限制。当队列中的字符到达高水印时, STREAMS 使另一个尝试发送消息到此队列的队列阻塞。当此队列中的字符减少到低水印时, STREAMS 消除另一个队列的阻塞。

瞬时持续时间锁 (instant duration locks) : 表现为一旦获得后立即释放的锁。通常用于在大量的数据对象要求锁定时, 但是该事务希望在等待另一个事务来释放其它的要求锁时 (该等待花费未知数量的时间), 避免获得一

些必须的锁。这些锁对于实现诸如 Jim Gray 的键范围锁定模式非常可用 (另见他的 *Transaction Processing: Concepts and Techniques*) 。

瞬时 (transient) : 不驻留在主存储器中的程序和子例程。

顺序存取 (sequential access) : (1) 以一种基于文件中记录的逻辑按序从文件读取、写至文件或从文件除去的存取方法。(2) 一种工具, 它从存储设备中获得数据或向存储设备输入数据, 这样进程就取决于数据位置以及对先前被存取数据的引用。

顺序文件存取 (sequential file access) : 一个范围内记录的位置, 从键值到以某顺序排列的与那些键值相关的值的后续处理。文件的索引无需是唯一的。

顺序 I/O 型号 (sequential I/O model) : 用于访问所有系统网络资源的操作系统的型号。当 SNA 支持此型号时, 它简化了对网络的访问, 允许程序设计为可移植的, 且允许程序通过重定向使用网络资源。

说明符 (specifiers) : 用于 C++ 说明以指明存储类、基本数据类型以及说明对象或功能的其它属性。

说明语句 (specification statement) : FORTRAN 中一组语句中的语句, 它向编译程序提供有关在源程序中使用的的数据信息以及如何分配存储的信息。

说明 (denote) : 请参阅声明 (*declaration*) 。

私有部分 (private part) : 有关 Ada 编程的信息, 请参阅软件包 (*package*) 。

私有对象 (private object) : (1) 在 XDS 中, 使用对象管理函数在工作区中创建的一个 OM 对象。(2) 在 XOM 中的一种对象, 用未具体说明的方式表示。

私有类型 (private type) : 在 Ada 语言中的一种类型, 其结构和值集合被明确定义, 但不直接可用于该类型的用户。私有类型仅为其判别式 (如果有的话) 和为其定义的操作集所知。私有类型及其可应用的操作在软件包的可见部分定义或在一般形式部分定义。还为私有类型定义赋值、相等和不等, 除非专用类型受到限制。

私有 (private) : C++ 类的一个专用成员仅对该类的成员函数和友元是可访问的。

死变量 (dead variable) : 在 Ada 语言中, 一个已经初始化但在程序上下文内没有被使用的变量。与不可达代码一样, 死变量由优化器检测并除去。

死代码消除 (dead code elimination) : 一种除去从来不会被引用或永远产生分支的代码的编译器优化。一种除去数据实体的存储指令 (其最终值不会被使用) 的编译器优化。

死区 (dead zone): 一个无法生成输入报告的图形输入板区域。每个虚拟终端可以设置自身的死区。与非输入区 (*no-input zone*) 同义。

死锁 (deadlock): (1) 一种出错情况, 因为进程的两个元素中的每一个都等待另一个元素的操作或响应, 导致进程无法继续。(2) 未解决的资源使用争用。(3.) 当多个进程等待一个不会变为可用的资源的可用性时发生的僵局, 因为资源正由另一个进程持有, 而该进程处于类似的等待状态。

死信文件 (dead letter file): 一个包含无法发送到正确目标文件的邮件消息。

搜索循环 (search loop): 用来执行表查找或在数组中查找异常值的数组处理循环。

搜索 (search): 扫描一个数据元素的集合以定位某特定项的所有实例 (比如文本字符串或文件名) 的操作。

速度 (speed): 波特率。线路速度 (*line speed*) 的同义词。

算法 (algorithm): 一种为了以有限的步骤解决某个问题而精确定义的有限规则集; 示例, 用于以给定精度求正弦 x 的值的算术运算顺序的完整规范。

算术表达式 (arithmetic expression): 一个或多个算术运算符和算术要素, 对其求值以产生数值。算术表达式可以是无符号算术常量、算术常量的名称、算术变量的引用、数组元素或函数或使用算术和括号构成的上述要素的结合。另见算术常量 (*arithmetic constant*) 和算术运算符 (*arithmetic operator*)。

算术常量 (arithmetic constant): 在编程语言中, 整型、实型、双精度型或复数常量。另见算术表达式 (*arithmetic expression*) 和算术运算符 (*arithmetic operator*)。

算术对象 (arithmetic object): 一个或多个类型为浮点、双精度型或长双精度型的整型对象。C 编译器能识别以长双精度型作为算术对象的对象。

算术运算符 (arithmetic operator): 一指导编译器执行算术运算的符号。另见算术表达式 (*arithmetic expression*) 和算术常量 (*arithmetic constant*)。

随机存取 (random access): 一种存取方式, 记录可以任何顺序从文件中读取、写入文件或从文件中除去。

随机文件存取 (random file access): 匹配特定索引键值的记录的位置。文件中记录的随机存取要求该文件有一个唯一索引, 并且每个记录有一个唯一索引键值。

缩放 (zooming): 在计算机图形中, 对显示图像进行递增按比例缩放, 以提供图像的从远距离视图到特写视图或从特写视图到远距离视图的视觉映象。

缩写 (abbreviation): 代表一更长字符串的字符串。

锁定按钮 (Lock button): 在 CDE 中, 用于锁定屏幕的前面板控制。

锁定服务 (锁定) (Lock Service, LOCK): 该 Encina Base 组件使事务在访问和修改资源之前锁定它们。

锁定文件 (lock file): n 个多进程应用程序, 磁盘上的一个共享进程用于控制对共享数据或设备的访问的系统文件文件。

锁定 (LOCK): 请参阅锁定服务 (*Lock Service*)。

锁定 (lock): 一种由键锁持有者限制资源的使用的机制。另见记录锁定 (*record lock*)。

索引分量 (indexed component): 指 Ada 语言中表示数组分量的索引分量。它是一个名称表, 包含指定数组分量的索引值的表达式。索引分量还可以表示一系列项中的某一项。

索引应用程序小窗口 (indexed application widget): 在用户界面语言 (UIL) 中命名的所有小窗口, 它不是用户界面定义 (UID) 层次结构中的所有其它小窗口的子窗口。

索引优先级 (index priority): 搜索查询中属性类型的优先级。

索引 (index); 变址: (1) 在建立了索引的文件中包含每条记录的键值和位置的表。(2) 一种计算机存储位置或寄存器, 其内容标识一组元素中的某个特定元素。(3) 文件或文档的内容表, 连同键或引用一起来定位内容。另见基址寄存器 (*base register*)。

所有权 (ownership): 实体的创建者或命名者。

所有者 (owner): 通过对象或操作定义的具有访问一个数据对象或操作的最高级别权限的用户, 通常是对象的创建者。

[T]

塔 (tower): 特定服务器的一组物理地址和协议信息。CDS 使用此信息来定位服务器驻留其上的系统并确定哪个协议可用于服务器。塔值包含在与对象条目相关的 **CDS_Towers** 属性中, 此对象条目表示单元名称空间的服务器。

弹出菜单 (pop-up menu): (1) 弹出 (*popup*) 的同义词。(2) 在 AIXwindows 中的一种 **MenuPane** 小窗口,

由于某些用户操作（通常单击鼠标按钮），这种小窗口就出现；当该操作完成时则消失。（3）转换操作的界面定义。（4）在 CDE 中的一种菜单，在被请求时，它显示在与之关联的对象旁。弹出菜单通常在单击鼠标按钮 3 或按 Shift+F10 后显示。

弹出窗口（pop-up window）： 在激活时自动打开的任何窗口。另见弹出（*pop-up*）。

弹出级联菜单（pop-up cascade）： 在“增强的 X-Windows”中，从一个模态弹出中连续放出的几个跳跃装载弹出。

弹出列表（pop-up list）： 存储在小窗口中的弹出子项的列表。

弹出小窗口（pop-up widget）： 在“增强的 X-Windows”中，根窗口的一个子窗口，与常规小窗口不同，它附加在自己的父小窗口；弹出小窗口在几何图形上不受其父小窗口的约束。

弹出子项（pop-up child）： 在“增强的 X-Windows”中，弹出列表上的一个子项。

弹出（pop up）：（1）在“增强的 X-Windows”中，在显示屏上显示信息或要求您做出选择的一种框。（2）在“增强的 X-Windows”中，与一种小窗口有关的操作，当按下某个指针按钮时这种小窗口自动在某些窗口内打开。（3）使用小窗口在小窗口树定义的窗口层次结构外创建一个窗口。

弹出（pop）： 当紧跟在流头下的模块被除去时使用的一种项。

弹出（popup）： 在 AIXwindows 中，由于某用户操作（通常单击鼠标按钮），一种 **MenuPane** 小窗口突然在显示器上出现（弹出）的方法。同弹出菜单（*pop-up menu*）。另见弹出窗口（*pop-up window*）。

弹下（popdown）： 在 AIXwindows 中，当某用户操作（通常单击鼠标按钮）完成时，一种 **MenuPane** 小窗口突然从显示器上消失（弹下）的方法。

弹下（pop-down）： 在“增强的 X-Windows”中，与一种小窗口有关的操作，当释放指针按钮时这种小窗口关闭。

弹性装入弹出式窗口（spring-loaded pop-up）： 一种类型的小窗口，比如菜单，它不可见于窗口管理器。弹性装入弹出式窗口禁用除了菜单中发生的事件外的用户事件。

套接字地址（socket address）： 唯一标识特定通信端点的数据结构。套接字地址由端口号和网络地址构成。它还标识协议系列。另见协议系列（*protocol family*）。

套接字（socket）：（1）由端口标识和 TCP/IP 地址的串联创建的唯一的主机标识。（2）端口标识。（3）16 位端口号。（4）NCS 中特定主机上的端口；通过某协议系列寻址机制可访问的通信端点。由套接字地址标识的套接字。另见套接字地址（*socket address*）、端口（*port*）和监听（*listening*）。

特定（specific）： 可能出现在给定类的实例中的属性类型，但不会出现在超类的实例中。

特权用户（privileged user）： 以 root 用户权限登录帐户的用户。

特权指令（privileged instructions）： 只能在处理器特权或超级用户状态下运行的系统控制指令。特权指令一般操作虚拟机或内存管理器，并通常不由应用程序程序员使用。另见特权状态（*privileged state*）。

特权状态（privileged state）： 一种硬件保护状态，在其中处理器可运行特权指令。对照非特权状态（*unprivileged state*）。另见特权指令（*privileged instructions*）。

特殊文件（special file）： 用于操作系统以向输入/输出设备提供界面。每个连接至计算机的设备至少有一个特殊文件。对照目录（*directory*）和文件（*file*）。另见块文件（*block file*）和字符特殊文件（*character special file*）。

特殊字符（special character）： 除了字母和数字以外的字符。例如，*、+ 和 % 都是特殊字符。

特性（characteristic attribute）： 反映或影响软件实体行为的属性类型。通常可以设置或更改特性。

提高（raise）： 使窗口的堆栈次序更高。

提交操作（commit operation）： 一个保存文件到永久性存储器的操作。

提交（commit）：（1）使自上次提交操作以来所有作用于数据库文件的更改永久，同时对记录解锁以便对其它用户可用。对照拒绝（*reject*）。（2）提交软件时，是对该软件产品的版本进行提交。提交产品时，从系统除去所有保存自软件产品的以前版本的文件，因而返回软件产品的以前版本是不可能的。在 SMIT 中，安装时可通过设置提问“提交软件？”为是（或通过使用带-ac 标志的 **installp** 命令）来提交软件。注意：对已经应用的软件提交不会更改软件产品的当前现行版本。它仅除去该软件产品的以前版本的保存文件。产品安装级别拒绝与产品更新拒绝的意思并不一样。一旦提交产品的新版本，而如果又要使用以前版本，则必须重新安装以前版本。请与应用（*apply*）比较并对照拒绝（*reject*）和除去（*remove*）。

提示符（prompt）： 请求信息或操作员操作的显示符号或消息。

提示性消息 (informational message) : (1) 一个为操作员提供信息, 而不要求一个响应的消息。 (2) 一个不是一个错误情形的结果的消息。

填充键 (key pad) : 键盘上的按键的物理组合, 如数字填充键和光标填充键。

填充字符 (fill characters) : (1) 用于填充存储字段的字符。 (2) 显示器上的可输入字符位置的可视表示 (例如, 每个位置的点或者位置间的垂直条)。

填充 (pad) : (1) 填充在字段中没有被哑元数据使用的位置, 通常为空格。 (2) 用来电路的引领传输损耗的设备。它可被插入引领损耗或匹配阻抗。

填充 (padding) : 插入到数据流的用来维护在协议请求的自然边界的对齐的字节。填充增加某些机器的体系结构的可移植性。

挑选 (culling) : (1) 在 GL 中, 如果图元小于在命令中指定的最小尺寸, 则它是被挑选的: 图元中的其它命令将不再被解释。另见剪切 (*clipping*) 和修剪 (*pruning*)。 (2) 在图形 API 中, 如果一个多边形是背面的, 则它可能被挑选, 或不被渲染。

条件编译语句 (conditional compilation statement) : 一种预处理器语句, 使预处理器处理文件中的指定代码 (取决于指定条件是如何求值的)。

条件变量 (condition variable) : 一种用于连接互斥 (锁、对象) 的同步对象。一种允许线程阻塞直至某事件发生的条件变量。

条件表达式 (conditional expression) : 一种 C 语言表达式, 包含一个条件 (第一表达式)、一个在条件为非零值的情况下求值的表达式 (第二表达式) 和一个在条件为 0 (零) 值的情况下求值的表达式。

条件分支 (conditional branch) : 当满足指定条件时执行的分支。

条件码 (condition code) : 与标志 (*flag*) 同义。

条件语句 (conditional statement) : (1) 指定表达式求值为非零的情况下运行的语句。 (2) 一条允许执行 (带或不带控制转移) 许多可能操作之一的语句。 (3) 一条基于指定准则, 用于表达赋值或分支的语句。

条件 (condition) : 程序或过程中的表达式, 在程序或过程运行时可以求值为真值或假值。

条件 (conditional) : 某些情况下的条件测试以执行特殊命令。

条目 (item) : 索引字段一行中的数据。

跳格 (tab) : 在显示屏上将光标移动到预先设置的位置。

跳线 (jumper) : 一种改变设备的操作特征的适配器上的连接。例如, 可以设置端口的 I/O 地址或中断类型。

跳跃计数度量 (hop count metric) : (1) 在网关中指示下一个字符串向目的地主机或网络提供跳跃计数。 (2) 某条路径中主机到主机的连接数目。

跳跃计数 (hop count) : 在令牌环网络中, 一帧在向目的地传送路径上所通过的网桥数。

跳转 (jump) : 在计算机程序运行时, 从指令正在运行的隐式或声明的顺序中脱离。

停止的状态 (stopped state) : 允许使设备不可用, 但是依旧可以装入它的设备驱动程序并将驱动程序绑定在内核中, 而且设备驱动程序依旧可以识别此设备。

停止方法 (stop method) : 将设备从可用状态引至停止状态。停止方法仅适用于支持可选停止状态的设备。

停止记录 (stop record) : Performance Toolbox 中值记录的一种特殊类型, 此值记录发送信号表明记录由于一组统计信息而停止并且给出此事件发生的时间。这允许程序使用记录文件来区分记录中的间隙与记录间隔中的差异。

停止位 (stop bit) : (1) 起止传输中位于每个字符末端的信号, 此信号为接收后续字符准备接收设备。 (2) 到接收机制以等待下个信号的信号。

通道标识 (channel ID) : 一个从多路复用设备传回来给文件系统的通道识别, 作为调用 **ddmpx** 入口点的结果。

通道号 (channel number) : 一个标识路径的编号, 通过其在特定输入或输出设备和计算机处理器之间传输数据。主设备号、次设备号和通道号唯一标识一个硬件设备。

通道控制字 (channel control word) : 请参阅 *CCW*。

通道路径标识 (channel-path Identifier, CHPID) : System/390 通道子系统中, 分配给每个已安装系统通道路径的值, 唯一地标识系统路径。

通道路径 (channel path) : 一个连接了一个或多个控制单元的单一接口。

通道 (channel) : (1) 信号或数据通过的路径。 (2) 给定读或写站可访问的这一部分存储介质。 (3) 数据通信中一种单向传输方法。 (4) 一个主机控制的功能部件, 处理处理器存储器与本地外围设备之间的数据传输。 (5) 控制单一通道路径的系统元件, 其操作方式取决于它所连接的硬件类型。 (6) 一个将处理器连接到输入输出设备的设备。 (7) 表

中 32 位之一，用来表示哪个事件类是活动的或非活动的。最高有效位称为通道 0，最低有效位称为通道 31。另见逻辑通道 (logical channel)。

通配符 (wildcard)：可用于匹配一个或多个字符的特殊字符，比如 * (星号) 或 ? (问号)。与模式匹配字符 (pattern-matching character) 同义。

通信端点 (communications endpoint)：X.25 中 DLS 用户和 DLS 供应商间的本地通信信道。

通信服务 (communications service)：由套接字应用程序编程接口执行的服务，允许数据包递送到指定的目标。提供三种类型的通信服务：可靠流递送、非连接方式数据报递送和原始套接字 (raw socket) 递送。

通信链路 (communications link)：请参阅数据链路 (data link)。

通信权限密码 (communications authority password)：两个通信安全性密码之一。控制对通信配置菜单的访问，以便只有授权人员才能够更改概要文件、对通信概要文件数据库中的一部分进行加密或控制 SNA 进程的启动。密码必须为 30 到 80 字符长的短语，其中允许空格。另见 BIND 密码 (BIND password)。

通信适配器 (communications adapter)：与软件相关的电路卡，使处理器、控制器或其它设备可以连接到网络。另见适配器 (adapter)。

通信线路适配器 (communications line adapter, CLA)：一个功能单元，把站 (station) 的按位串行 (serial-by-bit) 输出转换为并行位 (parallel bit) 格式并从并行位格式转换为按位串行输入到站。另见线路适配器 (line adapter)。

通信线路 (communications line)：在其间发生数据通信的线路；例如电话线。另见 X.25 线路 (X.25 line)。

通信协处理器 (communications co-processor)：扩充电路板上的一个微处理器，辅助系统部件中处理器的操作，使计算机能够与其它操作并行使用通信服务。

通信信道 (communications channel)：一个便于从一个位置到另一个位置的信息传输的电子路径。

通信 (communications)：计算机或远程设备间 (通常跨越很长的距离) 按照某种协议进行的数据传输。

通用报文传送协议 (Versatile Message Transfer Protocol, VMTP)：在用户级别提供数据报通信服务的协议。不像大多数使用 UDP/IP 的程序，使用 VMTP 的程序不必实现超时执行重新传输或网络延迟的预测，因为 VMTP 协议提供了端对端的数据报传送。

通用寄存器 (general purpose register, GPR)：可明确寻址的寄存器，可用于各种用途 (例如，作为累加器或变址寄存器)。另见寄存器 (register)。

通用唯一标识符 (Universal Unique Identifier, UUID)：用于标识的 128 位值。NCS 使用 UUID 来表示单元、接口、对象和类型。单元的 UUID 是由 uuid_gen 工具生成的，它是完全唯一的，并且是基于工作站和时间戳记的唯一系统标识而创建的。单元 UUID 生成后，就被置于 glb_obj.txt 文件中。另见对象 UUID (object UUID) 和类型 UUID (type UUID)。

通用异步接收 / 传输 (Universal Asynchronous Receive/Transmit)：用于异步数据通信应用程序的电路，它为在串行输入并行输出方式下恢复数据以及在并行输入串行输出的方式下传输数据提供所有必要的逻辑的电路。它通常为全双工，也就是说，它可以通过选择处理各种数据工作长度的选项，同时传输和接收数据。

通用坐标时间 (Universal Coordinated Time, UCT)：世界报时的新的标准术语，与格林威治标准时间相同含义。

通知标志 (notify flag)：表示事件循环时在那个点是否要求有应答的参数。

同步传输 (synchronous transmission)：(1) 数据通信中，由定时信号控制字符发送和接收的传输方法。对照异步传输 (asynchronous transmission)。(2) 一种数据传输，其中每个表示一个位的信号的发出时间与固定的时间基础相关。

同步管道 (conduit)：一种保护电线或电缆的管道。

同步数据链路控制 (synchronous data link control, SDLC)；同步数据链路控制协议：(1) 一种通信线路控制的形式，它使用命令来控制通信线路上的数据传送。对照二进制同步通信 (binary synchronous communication)。(2) 依照 ANSI 的高级数据通过程程 (ADCCP) 和国际标准化组织的 HDLC 的子集的规程。它管理链路连接上同步的，代码透明的、按位串行的信息传送。在交换或非交换链接上的传输交换可以为双工或半双工。链路连接的配置可以为点对点、连接或循环。

同步字符 (synchronization character, SYN)：二进制同步通信中，提供信号给定时接收站的传输控制字符。

同步 (clocking)：(1) 二进制同步通信中，使用时钟脉冲控制数据和控制字符的同步。(2) 数据通信中，一种控制在给定时间内在通信线路上发送的数据位数的方法。

同步 (synchronous)：(1) 两个或更多取决于发生特定事件 (比如公共定时信号) 的进程。(2) 以常规或可预测时间关系或顺序发生。

同级信任 (peer trust)： 两个单元间建立的信任关系的类型，依靠两个单元维护的相互认证代用品共享的密钥。同级信任关系启用一个单元中的主体来与其余单元中的主体安全地通信。

同余类 (congruence class)： 高速缓存中的行组，给定内存位置可对其进行映射。

同轴电缆 (coaxial cable)： 由一根导体（通常细铜管或金属丝）构成的电缆，该导体处在另一根直径更大的导体（通常铜管或铜编织物）之内并与之绝缘。

统计行 (statistic line)： 在 Performance Toolbox 中，指列表中表示特定值的行。对照上下文行 (context line)。

投标 (bid)： 邀请或选择的争用方式，尝试获得线路控制以传输数据。

头记录 (header record)： 文件开始处的一条记录，详述文件的大小、位置和其它信息。

头文件 (header file)： 包含由一组函数或用户使用的声明的文本文件。与包含文件 (include file) 同义。

透明绑定 (transparent binding)： 一种形式的格式绑定，它允许客户机应用程序绑定至任何导出期望接口的可用服务器。

透明方式 (transparent mode)： 二进制同步文本传输的方法，在其中仅数据链路转义 (DLE) 字符前面的传输控制字符作为传输控制字符处理。

透明 (transparent)： (1) 通信中指不能干扰数据链路（无论何种格式或内容）的传输。数据链路控制不识别透明传输。(2) 在数据通信中，指接收程序或设备不能将其作为传输控制字符识别的信息。(3) 指对用户不重要的操作或数据。

透视投影 (perspective projection)： 在绘制图元时用于获得真实感的一种制图技术。在透视投影中，投影线在视点处相遇；这样，图元的大小随着它与源投影的距离成反比变化。一个图元或图元的一部分离观察者越远，该图就画得越小。这种效果称为透视法缩小绘制图，它与照相取得的效应以及人可视系统取得的效应相似。另见正交投影 (orthographic projection)。

突出方式 (standout mode)： terminfo 结构使用的一般用途的突出显示机制。

突出显示 (highlight)： 使用几种方法中的任一方法在显示屏强调显示某一区域，例如加亮该区域或使该区域内的字符反色。

图标编辑器 (Icon Editor)： 在 CDE 中用来创建新图标以及修改现有图标的软件应用程序。

图标布局策略 (icon layout policy)： 在 AIXwindows 中确定是将表示最小化窗口的图标置于根窗口上还是置于图标框中的规范。

图标化 (iconify)： 请参阅图标 (icon)。

图标框 (icon box)： AIXwindows 中的一个窗口，用作表示最小化窗口的图标的可视存储区。

图标 (icon)： (1) 一个对象在显示屏上的图片或图形表示，用户可以使用一个设备（如鼠标）指向它，以便选择一个特殊操作或执行某一操作。(2) 在公共桌面环境中，指文件或目录、或者由桌面使用的对象的直观表示，由桌面上或目录窗口中的图片组成。

图片标记 (art tag)： 超文本中链接文本到图片窗口的标记。

图片框 (artwork frame)： Interleaf 中图形数据的容器。

图像高速缓存 (image cache)： 在 AIXwindows 中，将图像与名称关联的一种方法。一旦该关联到位，适当的 AIXwindows 子例程会通过引用 .Xdefaults 文件（按名称）和通过一个参数列表（按像素图）生成像素图。另见像素图 (pixmap)。

图像 (picture)： 用于显示公共桌面环境图标、背景模式和控件的像素图。

图形流水线 (graphics pipeline)： 指 GL 中一个图形图元在屏幕上变得可见之前所经历的步骤顺序：从模拟坐标变换成 NDC 坐标；3-D 剪切（如果出界）；透视分区；通过光照方程式或深度识别信号确定颜色；NDC 坐标变换成屏幕坐标；2-D 剪切（由屏幕屏蔽区）；光栅化（绘入帧缓冲区）以及显示帧缓冲区。

图形上下文 (graphics context, GC)： 指“增强的 X-Windows”中用于各种图形输出（如前景像素、背景像素、行宽）的存储区域和剪切区域。也称为“GC”和“GContext”，一个图形上下文只能和与其具有相同的根和相同深度的可绘制图元配合使用。

图形用户界面 (graphical user interface)： (1) 两个功能单元之间的边界，通过基本绘图单元（例如线条和多边形）显示给用户。(2) 一种计算机界面，由真实世界场景的可视隐喻组成，通常是桌面。在该场景内是各种图标，代表用户可以通过定位设备访问和操作的实际对象。（也称为图形界面 (graphical interface)。）

图形支持库 (Graphics Support Library, XGSL)： 至各种输出设备的一个二维图形应用程序编程接口。

图形字符 (graphic character)： 可以显示或打印的字符。

图形 API (graPHIGS API): PHIGS 的一个实现, 基于美国国家标准协会 (American National Standards Institute, ANSI) 提议的标准, 程序员层次交互式图形系统 (Programmer's Hierarchical Interactive Graphics System, PHIGS)。

图形 (graphics): 由诸如线条、曲线、多边形之类的基本绘图单元创建的一类数据。

图元空间 (primitive space): 图元坐标 (*primitive coordinates*) 的同义词。

图元小窗口 (primitive widget): 在“增强的 X-Windows”中, 一种小窗口, 它实例化自身已知的子类而不期望外部客户机也这么做。图元小窗口没有通用的几何管理方法。实例化子的图元小窗口对其下的所有向下请求遍历操作负责。另见小窗口 (*widget*)。

图元字体 (primitive font): 将字符定义成图元的字体。图元字体字符和所有其它原语一样, 可以改变比例和旋转。另见光栅字体 (*raster font*) 和字体 (*font*)。

图元坐标 (primitive coordinates): 在其中定义图元的空间。选择一个方便的点作为起始地址, 并且相对于该点定义图元。图元空间 (*primitive space*) 的同义词。另见视觉坐标 (*eye coordinates*)、屏幕坐标 (*screen coordinates*) 和世界坐标 (*world coordinates*)。

图元 (pel): (1) 在计算机图形学中, 能独立被赋予颜色和亮度的显示空间最小元素。(2) 在帧缓冲区或显示器上的点。另见像素 (*pixel*)。

图元 (primitive): 绘制命令, 如 **arc**、**line**、**circle**、**polygon** 或 **charstr**。这样的命令被称为原语, 因为它们不是由更小的部件组成的, 而且因为它们是组成更复杂情景的基本片断。还用于描述绘制命令创建的图。

推安装 (push installation): 在网络安装管理环境中, 从机器而不是目标初始化的一种安装。

推许可权 (push permissions): 启用远程命令执行的许可权。

推 (push): 当模块被插入到紧接着流头下的一个流时使用的一种术语。

退出按钮 (Exit button): CDE 中, 用于从桌面登出的前面板控件。

退出值 (exit value): (1) 在命令完成时发送给标准输出或标准错误的代码。(2) 命令返回的一个数值表示它是否成功完成。某些命令返回的退出值给出了诸如一个文件是否存在等其它信息。Shell 程序能够测试返回值以控制分支和循环。

退出 (quit): 通知系统返回前一状态或停止进程的键、命令或操作。

吞吐量级别协定 (throughput-class negotiation): 在 X.25 通信中, 允许 DTE 协商它的包在包交换网络中移动的速度的可选设施。另见可选的设施 (*optional facilities*)。

吞吐量 (throughput): 每个时间单元内可以完成的工作负荷操作的数量。

拖动 (drag): 用鼠标指针指向一个对象, 按下鼠标按钮, 移动鼠标, 然后释放鼠标按钮。一种对象“拖动”到目标点的方法。

脱机 (offline): (1) 当它不在计算机的直接控制下时的功能单元的操作。(2) 既不直接受控也不用计算机通信。对照联机 (*online*)。

拓扑排序 (topological sort): 将有序对的无序列表进行排序的排序文件。

拓展 (widening); **加宽**: 通过将符号位的拷贝填充到位于值左边的位实现的值大小的扩展 (例如从 **short** 到 **int**)。

[W]

外部变量 (external variable): 一个可以为另一个编译单元访问的变量。另见编译单元 (*compilation unit*)。

外部单元 (foreign cell): 不是本地机器所属的单元。另见本地单元 (*local cell*)。

外部的数据表示 (external Data Representation, XDR): 一种数据描述和编码标准; 它使用一种语言描述数据格式, 但是此语言只用于描述数据, 不是编程语言。例如 RPC 和 NFS 协议使用 XDR 描述数据格式。

外部调制解调器 (external modem): 一个同它操作的单元分离的调制解调器。

外部符号 (external symbol): 一个定义在不同于符号出现文件的文件中的符号。一个普通符号代表一个外部引用。

外部过程 (external procedure): 与外部例程 (*external routine*) 同义。

外部函数 (external function): 在 FORTRAN 中, 与外部例程 (*external routine*) 同义。

外部例程 (external routine): 从定义例程的外部程序调用的过程或函数。与外部过程 (*external procedure*) 同义。

外部名称 (external name) : (1) 可以被任何控制部分或单独的汇编或编译模块引用的名称; 一个控制部分名称或者在另一个模块的项名称。(2) 在一个程序中, 一个作用域名称没有必要限制在一个块和其它包含的块中。

外部数据定义 (external data definition) : 出现在函数外部的变量的描述。它使系统分配存储量给变量, 并使按照定义存在于和该定义相同的文件中的函数可以访问变量。

外部同步 (external clocking) : 在数据通信中, 提供进行数据同步的调制解调器的能力。

外部网关 (exterior gateway) : 在一同其它自治系统通信的自治系统上的网关。

外部引用 (external reference) : 一个符号的引用, 定义为一个在其它程序或模块的外部的名称。

外部主机 (foreign host) : 远程主机 (*remote host*) 的同义词。

外围单元 (peripheral unit) : 见外围设备 (*peripheral device*) 。

外围设备同时联机操作 (simultaneous peripheral operation online) : 请参阅假脱机 (*spooling*) 。

外围设备 (peripheral device) : 关于特殊处理单元, 可与该单元直接通信的任何设备。

完全备份 (full backup) : 系统上所有文件的备份副本。对照增量备份 (*incremental backup*) 。

完全覆盖安装 (complete overwrite installation) : 一种完全覆盖已经安装在系统中的现有 BOS 版本的安装方法。该过程可能削弱数据恢复或破坏所有硬盘驱动器上的现有数据。执行完全覆盖安装前请确保备份系统。

完整包序列 (complete packet sequence) : 一个单独的 X.25 数据包或 M 位设为 1, D 位设为 0 的包序列, 后跟 M 位设为 0, D 位按需要设置的更多数据包。

完整类名 (complete class name) : 嵌套 C++ 类名的完整限定, 包含所有内嵌的类名。

完整性 (integrity) : 在安全 RPC 通信中可能指定的一个保护级别, 用以确保在两个主体之间的数据传输不能在传送中被修改。

网关至网关协议 (Gateway-to-Gateway Protocol, GGP) : 网关使用该协议来确定与网络和邻近网关的连接性, 并实现最短路径路由算法。

网关主机 (gateway host) : 与独立网络连接的主机。它有多个接口, 每一个都具有不同的名称和地址。

网关 (gateway) : (1) 一种在链路层上运行的实体, 在需要时将一个网络使用的接口和协议转换成由另一个截然不同的网络使用的接口和协议。(2) 连接主机的网络。另见活动网关 (*active gateway*) 。(3) 使不同体系结构的网络或系统互相连接的设备及其相关软件。连接通常在参考模型网络层上进行。例如, 网关允许 LAN 访问 System/390 主机。对照网桥 (*bridge*) 。

网际控制报文协议 (Internet Control Message Protocol, ICMP) : 网关和源主机进行通信的协议, 例如, 报告数据报中的错误。它是网际协议 (IP) 的完整部分。

网际路由器 (Internet Router) : 使 IP 主机作为网关, 在使用特定适配器的不同网络上路由数据。

网际协议 (Internet Protocol, IP) : 一种为从高端主机到主机协议到本地网络协议提供接口的协议。主机到主机的寻址经常发生在这个级别。

网络安装管理 (NIM) : 一种提供在网络接口内安装和配置软件的环境。

网络地址 (network address) : (1) 表示特定网络的地址的那部分。一个用于网络上机器的完整地址, 由网络地址与主机地址构成。(2) 在 NCS 中, 网络或因特网中的特定主机的唯一的标识符 (在地址系列内)。网络地址足够用来识别一台主机, 但不能标识主机内的通信端点。

网络对象 (network object) : 网络安装管理 (NIM) 数据库的项, 它代表一个局域网。

网络供应商 (network provider) : 在 X.25 通信中, 该组织通常是 PTT, 它提供公共网络。

网络管理 (network management) : 数据站的概念性控制元素, 它和该数据站的所有层都有接口并设置和重新设置控制参数、获取出错状态的报告, 以及确定该数据站是否应该和介质连接或断开连接。

网络计时协议 (NTP) : 建议 Internet 时间标准。

网络计算内核 (Network Computing Kernel, NCK) : RPC 运行时库与“位置代理”的结合, 它包含了运行分布式应用程序需要的部分。

网络计算体系结构 (Network Computing Architecture, NCA) : 支持分布式计算的协议与体系结构的集合。

网络计算系统 (Network Computing System, NCS) : 由 Apollo Computer Inc. 开发的遵从网络计算体系结构的软件工具的集合。这些工具包括“远程过程调用”运行时库与“位置代理”。NCS 是由 License Use Management 使

用的基础通信协议，在客户机与服务器间传送发放许可证事务。消息从客户机广播到受 NCS 管理的网络。

网络接口 (network interface)： 在网络层把信息包格式化为特定的网络适配器可以理解和发送的分组报文的软件。

网络适配器 (network adapter)： 在网络中允许设备与其它设备通信的电路。

网络通信控制设施 (Network Communications Control Facility, NCCF)： 一个用作为命令处理器的基本许可程序，它能监视、控制与改进网络的操作。

网络文件系统 (Network File System, NFS)： 一种分布式文件系统，它使用户可以访问远程计算机上的文件和目录并把这些文件和目录象在本地一样对待。NFS 通过使用远程过程调用 (RPC) 而独立于机器类型、操作系统和网络结构。

网络协议 (network protocol)： OSI 网络结构的网络层通信协议，例如网际协议 (IP)。

网络信息中心 (NIC)： DARPA TCP/IP 信息分发中心

网络许可证系统 (Network License System)： 请参阅 *License Use Management*。

网络掩码 (netmask)： 32 位的掩码，用于标识局域网 (LAN) 的最本地部分。

网络引导映象 (network boot image)： 支持独立的、无盘的无数据机器的一个引导映象。

网络用户地址 (NUA)： 在 X.25 通信中，X.121 地址包含多至 15 位的二进制代码数字。

网络用户识别 (NUI)： (1) 使传送 DTE 在每次调用基础上为 DCE 提供支付、安全或管理信息的设施。(2) NUI 可以不依赖使用的端口而识别网络用户。另见 *地址 (address)*。

网络用户 (network user)： 访问网络层服务的内核级协议或用户级应用程序。

网络终止单元 (NTU)： 在 X.25 通信中，访问网络的点。

网络状态 (network state)： 一种状态，它表示网络对象可参与 NIM 操作或者网络对象定义中的错误。

网络 (network)： 通过通信线路连接的数据处理产品的集合，用于在不同位置间进行信息交换。

网桥 (bridge)： (1) 本地环路、通道或环的连接中，用来匹配电路和便于正确数据传输的设备和技术。(2) 一个功能

部件，用来连接两个使用相同的逻辑链路控制 (LLC) 过程但不同介质访问控制 (MAC) 过程的局域网 (LAN)。对照 *网关 (gateway)*。

微文档 (microdocument)： 一个帧中的有它自己的属性，而且可以和主文档共享组件的一个文档。

维护包更新 (maintenance level update)： 服务更新 (修正和增强) 是必要的，以更新基础操作系统 (BOS) 或当前发行版级别的可选软件产品。另见 *服务更新 (service update)*。

维护方式 (maintenance mode)： 一个产品或系统可被服务的状态。与 *服务方式 (service mode)* 同义。

维护分析过程 (maintenance analysis procedure, MAP)： 由客户工程师和服务代表使用的文档以便修复设备。一个 MAP 包含是 / 否提问和过程以指引用户到设备的故障部分。

维护系统 (maintenance system)： 一个从软盘装入的操作系统的特别版本，用于执行系统管理任务。

维 (dimension)： 赋予数组和表的大小属性。

伪彩色 (pseudocolor)： 在“增强的 X-Windows”中的一类色图，在该色图中像素值索引色图条目以产生独立的红、绿、蓝值。即，色图被作为三个一组的数组 (RGB 值) 查看。RGB 值可以被动态修改。这是对直接颜色色彩图类互斥的。

伪设备 (pseudo device)： 基于设备的一种软件；例如，pty 设备。

尾数 (endian)： 一种数据表示的属性，反映了某种多八位元数据怎样存储在内存里。请参阅 *大尾数法 (big endian)* 和 *小尾数法 (little endian)*。

伪终端 (pseudo terminal, PTY)： /dev 目录中的一种特殊文件，它能作为使用 Berkeley 线路规程的键盘和显示设备有效地工作。伪终端由一对字符设备组成，这一对字符设备指“主”和“从”。从设备 (/dev/pts) 由另一个进程通过伪终端的主半 (/dev/ptc) 来操作。

伪 PostScript (Pseudo-PostScript)： 类似于 PostScript 的一种图形语言，用于驱动文本和图形的输出。

谓词 (predicate)： 一种布尔逻辑项，指示确定一些变量状态的逻辑表达式。例如，一个谓词可以是一个表达式，说明“变量 A 必须有值 3”。与条件变量联合使用的控制表达式基于谓词。使用一个条件变量等待某个谓词变成“真”，例如，等待一些东西进入队列。

未定义 (undefine)： (1) 表明用于正从系统中将设备实例取出。这可以通过运行设备的未定义方法来实现。在

“定制数据库”中有关数据的所有信息通过此操作清除。
(2) 使当前计算机系统不再识别某命令。

位分散 (bit scattering): 一种内存位映射技术, 确保内存字中最小量的位数是存储在单一的 DRAM 里。有了最小量的位数, ECC 能更好地检测并改正由坏 DRAM 引起的错误。

未格式化文件 (unformatted file): 文件显示时, 其数据未使用特定字符排列。

位块传送 (BLT) (bit block transfer, BLT): 通过指定图像的左下角和右上角及目标地址, 移动二进制图像 (位图或像素图)。

位面掩码 (plane mask): (1) 确定哪些显示适配器存储位置由输出函数修改。(2) 在“增强的 X-Windows”中, 限制影响位平面的一个子集的图形操作的一种位掩码。它存储在图形上下文中。图形操作可限制为只影响目的地的位平面的子集。

位面 (plane): 当像素图或窗口被作为位图的一个堆栈看待时, 每个位图就被称为一个位面或一个位平面。

未配置 (unconfigure): (1) 表明用户正将设备从可用 (配置的) 状态引至定义的状态。这可以通过运行设备的未配置方法来实现。“定制设备对象类”中的设备状态字段会反映此操作。(2) 使当前计算机系统不能再使用。

位平面 (bit planes): 在计算机图形学中, 位平面为屏幕上的每个像素提供一位色彩信息。因而一个八位位平面系统允许每个像素显示 2 的 8 次方的不同色彩。

未使用子程序 (unused subprograms): Ada 语言中, 只有在特定程序或单元集上下文中未使用到的子程序。例如, 一个程序可能仅调用实用程序包中子程序中很小的子集。剩余未被调用 (执行不到的) 子程序代替了该程序上下文中执行不到的代码。

位率 (bit rate): 位发送速率, 通常以位/秒表示。

位同步 (bit clocking): 在 EIA-232C 或 D 接口中, 表明设备的哪一部分 (调制解调器 (DCE) 或计算机 (DTE) 中的任一者) 为同步数据事务提供时钟信号的字段。

位图文件 (bitmap file): 一个文件, 包含用来创建位图的高度和宽度指示信息。

位图显示器 (bitmapped display): 一个显示器, 它的显示适配器具有显示器上每个独立可解析点的硬件表示。该硬件表述可以是处理器内存或适配器内存。另见全点解析显示器 (All Points Addressable Display)。

位图 (bitmap): 深度为一个位平面的像素图。

位引力 (bit gravity): 在“增强的 X-Windows”中, 窗口内容对窗口位置的吸引力。当窗口改变大小时, 其内容可以重定位。可以请求服务器重定位先前内容到窗口里的某个区域。另见引力 (gravity)。

未映射窗口 (unmapped window): 屏幕上不可见的窗口。

位域 (bit field): 包含 1 位或更多命名位的结构或联合成员。

位置参数 (positional parameter): (1) 一种 shell 设施, 它将命令行的值指定给程序中的变量。(2) 必须出现在相对于其它位置参数某个指定位置的一种参数。

位置代理客户机代理程序 (Location Broker Client Agent): NCS 位置调度部件。程序利用本地调度客户机代理来同全局位置调度和远程位置调度通信。另见本地位置代理 (Local Location Broker) 和位置代理 (Location Broker)。

位置代理 (Location Broker): 在 NCS 中, 一组包含本地位置代理, 全局位置代理, 和本地位置协调代理的软件。位置代理保持对象位置信息。另见代理 (broker)、本地位置代理 (Local Location Broker), 和位置代理客户机代理程序 (Location Broker Client Agent)。

位置代码 (location code): 一个来自处理器适配器的路径, 它通过信号电缆和扇出框, 如果有一个, 到设备或者工作站。代码有 4 个信息字段: 抽屉、插槽、连接器和端口。

位置关联 (positional association): 在 Ada 语言中指定一项与列表中一个位置的关联, 方法是使用文本中的同一位置来指定该项。

位置计数器 (location counter): 汇编程序中表示代码分配的下一个可用字节的一个计数器。位置计数器为程序语句指定存储器地址。另见指令地址注册 (instruction address register)。

位置引用 (locality of reference): 一个运行的程序利用指令和 / 或数据的一个地址压缩范围的级别。

位置 (在一个串中) (position (within a string)): 一个字符串元素相对于另一个字符串元素的序数位置。

位置 (在一个属性中) (position (within an attribute)): 一个值相对于另一个值的序数位置。

位置 (position): (1) 字符串中可被一个元素占据并用序列号标识的任何位置。(2) 在一个记录、显示消息或计算机打印输出序列中某个字符的位置。

未转义 (unescaped)：表达式中，前面不是转义序列，所以作为控制字符解释的字符。另见转义序列 (*escape sequence*)。

位 BLT (bit BLT)：请参阅位块传送 (*bit block transfer*)。

位 (bit)：计算机用于存储信息的二进制数字 0 或 1 中的任一个。与二进制数字 (*binary digit*) 同义。另见字节 (*byte*)。

文本编辑器 (Text Editor)：CDE 中用来创建和编辑文档的软件。

文本端 (textport)：GL 中显示屏上的区域，用于表示来自图形或非图形程序的文本输出。

文本格式化程序 (text formatting program)：确定页面上如何放置数据的方式的程序。

文本光标 (text cursor)：指明何处输入字符的光标。此文本光标由键盘控制。

文本环绕 (text wrap)：请参阅字环绕 (*word wrap*)。

文本缓冲区 (text buffer)：文本存储区域。

文本输入方式 (text input mode)：当在文件中输入字符时，编辑器解释输入的字符的方式。另见命令方式 (*command mode*) 和末行方式 (*last line mode*)。

文本锁定 (text lock)：允许调用进程将它的文本段锁定或解锁至内存。

文本小窗口 (text widget)：用于定制用户界面和程序界面的文本编辑器。

文本指示符 (text indicators)：INed 窗口底部的符号，用于显示用户能够滚动查看文件附加部分的方向。

文本字段 (text field)：CDE 中的窗口中信息输入的矩形区域。带有键盘焦点的文本字段，它们具有闪烁文本插入光标。

文本字符串 (text string)：由用户定义的字符（字母数字或特殊）序列。

文本 (text)：(1) 一种类型的数据，由一组语言学字符（字母、数字和符号）和格式化控制符构成。(2) 程序中能够运行的部分。(3) 在 kernel 方式中，包含运行的内核程序代码。它只能由用户进程读取。(4) ASCII 和数据通信中的一个字符序列，它是一个开始文本通信控制字符领头，并以结束文本通信控制字符结束的实体。另见程序文本 (*program text*)。(5) 字处理中，用于人员查看的信

息，它显示在一个二维表单中，比如打印在纸上或显示在屏幕上的数据。(6) 消息的一部分，它不是报头或控制信息。

文件传输协议 (File Transfer Protocol, FTP)：在 TCP/IP，协议使在主机之间或间接使用外部主机传送数据成为可能。

文件传送 (file transfer)：在远程通信中，从一个系统通过通信链接到另一个系统的文件传送。

文件存储器 (file memory)：虚拟内存页是当前在实际内存中，不是计算内存的一部分。通常，这些是非可执行文件页。

(文件的) 属性 (attribute (of file))：有关文件的信息的某些部分，用于确定其访问和组织特征。

文件服务器 (file server)：在 CDE 中，存储由应用程序使用的文件数据的主计算机。

文件格式 (file format)：在 ASCII 文件中创建的项的描述，诸如配置和定制概要文件。

文件管理器 (file manager)：操作文件和目录的程序。

文件规范 (file spec)：请参阅文件规范 (*file specification*)。

文件规范 (file specification, filespec)：文件的名称和位置。文件规范是由驱动器说明符，路径名称和文件名组成。

文件后缀 (file suffix)：在 CDE 中，在文件名后面添加的后缀，经常用在文件类型化或者按用户归类文件。

文件集 (fileset)：(1) 一个单独的安装或更新选项。选项提供特定功能和更新以更正先前安装的选项中的错误或增强先前安装的选项。(2) 一个或多个单独的安装，在一个安装软件包中的逻辑商组成一个单元。

文件夹 (folder)：(1) 消息处理程序中的消息目录。(2) 在 CDE 中，代表目录的图标。

文件结束 (EOF)：文件结束 (End of file)。

文件类型 (file type)：在操作系统中，五种可能的文件类型之一：普通文件，目录，块设备，字符设备和先进先出 (FIFO 或者命名管道)。

文件描述符 (file descriptor)：一个小的正整数，系统使用它而不是文件名标识一打开的文件。另见共享内存标识 (*shared memory ID, shmid*)。

文件名匹配替换 (globbing)：文件名替换 (*file name substitution*) 的同义词。

文件名替换 (file name substitution) : shell 识别词语 (字符串) 的过程, 词语包括任何 *、?、[、或者 { 字符、或者以 ~ 字符开始, 并用与词语模式相匹配的文件名 (按字母顺序排序) 的列表替换。与文件名匹配替换 (globbing) 同义。

文件名 (file name) : (1) 为文件指定或声明的名称。 (2) 程序使用此名称标识文件。另见标号 (label) ; 标签。

文件名 (filename) : 在 DOS 中, 扩展名之前的文件名一部分。另见文件 (file) 。

文件模式创建掩码 (file mode creation mask) : 请参阅掩码 (mask) 。

文件树 (file tree) : 一特定节点的完整目录和文件结构, 从根目录开始。一个文件树包含所有本地和远程在目录和文件上执行的远程安装。

文件索引 (file index) : 描述文件的 64 字节的信息。文件索引包含文件的类型和大小, 以及在物理设备上的文件数据的位置。此索引与操作系统 i-node 相同。与 i-node 同义。

文件所有者 (file owner) : 拥有文件最高访问权限级别的用户, 由文件定义。

文件系统 (file system) : 在物理或逻辑海量存储设备 (诸如软盘和小型磁盘) 上的文件和文件的管理结构的集合。

文件指针 (file pointer) : 一个表示包含文件名的结构标识符。

文件中的空穴 (hole in a file) : 某稀疏文件中的空白空间, 通常保持打开状态以供将来添加数据使用。另见稀疏文件 (sparse file) 。

文件作用域 (file scope) : 一个在所有块和类之外声明, 具有文件作用域的 C++ 名称, 能够为源文件中随后的声明点使用。

文件 (file) : (1) 相关数据的集合, 由一指定的名称存储和检索。对照特殊文件 (special file) 。 (2) 记录的序列。如果文件位于内部存储, 它就是一个内部文件; 如果是一个输入 / 输出设备, 它是一外部文件。 (3) 相关信息的集合, 存储在为组织目的单个位置, 并作为一个单元进行处理。另见文件名 (filename) 。

纹理 (texture) : 用于填充矩形、凸多边形、弧形和环形的模式。

文字字符串 (literal string) : 一个不包含模式匹配字符, 而且必须要按原样来解释的字符。对照正则表达式 (regular expression) 。

文字 (literal) : (1) 在一个源程序中的一个本身就是数据而不是一个对数据引用的符号或者量。 (2) 在编程语言中, 一个直接表示一个值的单元。例如, 14 代表整数 14。 (3) 在 Ada 语言中, 通过字母和其他字符的方式, 一个文字字面意义上代表一个值。一个文字或者是一个数值型文字, 枚举文字, 字符文字, 或者是一个字符串文字。

问题管理记录 (Problem Management Record, PMR) : 支持中心指定给已报告问题的号。

问题确定过程 (problem determination procedure) : 步骤的一种指示序列, 该步骤是从问题条件中恢复或欺骗问题条件的步骤。

问题确定 (problem determination) : 标识问题源的过程。通常该过程将程序、设备、数据通信设施或用户错误标识为问题源。

问题状态 (problem state) : (1) 两个虚拟机保护状态之一, 以处理器的非特权状态运行的状态。用户写的应用程序通常以问题状态运行。 (2) 一种状态, 处理单元在其中不能运行输入 / 输出和其它特权指令。

握手 (handshaking) : 调制解调器和设备在传送数据之前所使用的过程, 用来建立电路径和同步。另见聊天脚本 (chat script) 和期望 - 发送序列 (expect-send sequence) 。

无编号确认 (unnumbered acknowledge, UA) : 一个链路控制帧。

无编号信息帧 (unnumbered information (UI) frame) : 无编号格式中的帧, 用于传送无编号控制功能。

无间隔字符序列 (nonspacing character sequence) : 用于强调字符, 由有效附加符号后跟按字母字符或空格组成的两部分序列。系统将该序列转换为单一代码点, 它产生有特定附加符号的字母字符。

无间隔字符 (nonspacing character) : 请参阅区分 (diacritic) 。

无连接方式 (connectionless mode) : 一种传送方式, 从一个用户传递到在自包含单元 (在单元间无需逻辑关系) 中的另一个用户。

无连接信息包递送 (connectionless packet delivery) : 一种数据包递送方法, 单独对待每一个信息包且不保证递送 (成功) 。

无盘工作站管理器 (Diskless Workstation Manager, DWM) : 初始化和维护无盘客户机和无盘服务器的资源的操作系统软件。它是一组命令、awk 命令脚本和源码。

无盘工作站 (diskless)：一台没有本地文件系统或本地引导映象，远程访问自身的部分资源的工作站。无盘客户机从无盘服务器远程启动，并使用服务器进行远程页面调度。

无数据 (dataless)：一台没有本地文件系统或本地引导映象以远程访问自身的部分资源的工作站。无数据客户机将本地磁盘用于页面调度和转储设备。

无特权状态 (unprivileged state)：一种硬件保护状态，其中处理器只能运行无特权指令。处理器的无特权状态支持虚拟机操作系统状态和问题状态。对照特权状态 (*privileged state*)。

无条件分支 (unconditional branch)：每次遇到都被接收的分支。

无线电频率干扰 (radio frequency interference)：一种信号 (噪声)，它来自一个电路或可能导致不正确操作干扰另一个电路或电子设备的电子设备的辐射或引导。一种电路，或可能是电子干扰 (EMI) 的发射器的设备，或易受电子干扰的设备。

无限 (infinity)：(1) 一个数据集的上界名称。(2) 在二进制浮点概念中，一个相应标志的值在大小方面比任何二进制浮点数都大。

物理层 (physical layer)：根据 ISO 开放系统互连 (OSI) 参考模型指定的网络设计的最低层。该层担负以下责任：与介质交互，检测和生成介质上的信号，转换和处理从介质上和数据链路层接收的信号。另见物理级别 (*physical level*)。

物理层 (physical level)：在 X.25 通信中，用于激活、维护和取消激活数据终端设备 (DTE) 和数据电路端接设备 (DCE) 之间的物理链路的机械的、电气的、功能的和过程的介质。与级别 1 (*level 1*) 同义。另见级别 (*level*)、帧级别 (*frame level*)、数据链路级别 (*data-link level*)、信息包级别 (*packet level*) 和物理层 (*physical layer*)。

物理单元 (physical unit, PU)：在 SNA 中，控制与节点关联的实际物理硬件的一组程序。

物理分区 (physical partition, PP)：对逻辑卷进行磁盘空间分配的最小单位。物理分区是一个物理卷上连续的空间。物理卷的固定大小部分。一个或多个物理分区组成逻辑分区的基本物理存储介质。

物理卷 (physical volume, PV)：(1) 对单个读/写机制可访问的存储器的单个单元的部分；例如，一个磁鼓、一个磁盘组或磁盘存储模块的一部分。(2) 物理连接到计算机上的读写固定磁盘。单个固定磁盘驱动器提供的实际存储空间。另见日志卷 (*log volume*)。

物理块 (physical block)：请参阅块 (*block*)。

物理设备 (physical device)：请参阅设备 (*device*)。

物理数据块 (physical data block)：请参阅块 (*block*)。

物理网络 (physical network)：由物理网络电缆、调制解调器或其它硬件链接的机器的网络。一个物理网络可包含一个或几个逻辑网络。

物理文件 (physical file)：(1) 含有为其创建一个或多个替代索引的数据的索引文件。(2) 它描述如何将数据提供给程序或从程序接收数据，并描述数据实际上如何存储在数据库中的。物理文件包含一个记录格式和一个或多个成员。

[X]

析构函数 (destructor)：类的特殊的成员函数，名称为其类名称前加一个 ~ (代字号)。不能为此函数指定参数或返回类型。析构函数将为对象进行“清理工作”，也就是诸如释放任何在创建对象后动态分配的存储空间之类的工作。

西里尔 (Cyrillic)：西里尔字母。用于撰写旧教堂斯拉夫语及用于俄语和其它多种斯拉夫语的字母。

稀疏数组 (sparse array)：一个数组，其中极少数的定义单元得到使用。

稀疏文件 (sparse file)：一种文件，它创建时的长度比它包含的数据更长，留下个空白空间用于未来数据的增加。另见文件中的空穴 (*hole in a file*)。

系列 (family)：有主体和子单元的 Ada 语言库。不是所有的潜在的系列成员都需要出现；一个库单元可能有丢失的可选的主体，一个主程序过程可能只有一个隐式的规范。次级单元的系列是同任意子单元在一起的单元。

系统板 (system board)：系统单元中的主要电路板，支持各种基本系统设备，比如键盘、鼠标等等。系统板还提供其它基本系统功能。

系统菜单 (system menu)：AIXwindows 中，窗口左上角的下拉菜单，它允许用户恢复、移动、最大化、最小化窗口以及调整窗口大小。它还允许用户退出应用程序或关闭窗口。还能使对话框外观包含活动应用程序列表。通过可选的分割窗口的技术，用户能同时查看相同对象的许多部件。

系统参数 (system parameters)：内核参数 (*kernel parameters*) 的同义词。

系统程序控制器 (System Program Controller, SPC)： 控制其它在后台 (守护程序) 运行的应用程序的操作的系统程序。

系统重新启动 (system restart)： 初始程序装入 (initial program load)。

系统存储器 (system memory)： 与主存储器 (main storage) 同义，但用于硬件，指半导体存储器 (模块)。

系统单元 (system unit)： 系统中包含处理单元的部分。

系统地址列表 (system address list)： 由系统管理器控制的地址列表，系统上所有用于可以通过 **xtalk** 使用它来进行呼出 X.25 呼叫。另见地址列表 (address list) 和用户地址列表 (user address list)。

系统颠簸 (thrashing)： 由于内存严重过量使用而引起的情况，此时系统正花费它所有时间写出虚拟内存页并将它们读回去。应用程序没有进展，因为它们的页面在内存中停留的时间不足以使用它们。内存负载控制是为了避免或停止系统颠簸。

系统调用 (system call)： 程序对操作系统子例程的调用。

系统定制 (system customization)： 为特定数据处理系统指定设备、程序和用户。对照配置 (configuration)。另见定制 (customization)。

系统服务控制点 (system services control point, SSCP)： 在 SNA 中，指 SNA 网络内的焦点，用于管理配置、协调网络操作员和问题确定请求、以及向网络最终用户提供目录支持和其它会话服务。多个互相协调的 SSCP 能够将网络划分为多个控制域，每个 SCP 对其域中物理单元和逻辑单元具有分级控制关系。

系统概要文件 (system profile)： 包含用于系统操作的缺省值的文件。

系统管理界面程序 (System Management Interface Tool, SMIT)： 一组菜单驱动的服务，它们便于各种系统任务的性能，比如软件安装和配置、设备配置和管理、问题确定以及存储管理。在基于字符的 curses 界面以及基于 AIXwindows 的图形用户界面中都提供 SMIT。

系统管理 (system management)： 涉及维护系统的正常工作秩序以及修改系统以使其满足更改要求的任务。

系统节点 (system node)： 在设备位置层次结构中，这是最高的节点。如果您跟随连接路径，每个硬件设备会带回系统节点。例如，SCSI 磁盘连接至 SCSI 适配器，SCSI 适配器连接至总线，此总线连接至系统节点。

系统控制台 (system console)： 通常装备键盘和显示器屏幕的控制台，操作员用它来控制系统并与系统通信。与控制台 (console) 同义。

系统启动 (system startup)： 初始程序装入 (initial program load)。

系统日期 (system date)： 系统用户在安装过程中指定的日期，并由系统维护。请参阅创建日期 (creation date) 和会话日期 (session date)。

系统时间 (system time)： 操作系统提供服务给应用程序所花的时间量。系统时间包含操作系统分配存储器或设备到您的程序所花的时间，以及处理您的程序作出的操作系统调用所花的时间。

系统提示符 (system prompt)： 命令行 (command line) 的同义词。系统提示符是在操作系统命令行出现的符号。系统提示符指明操作系统已就绪，用户可输入命令。

系统网络体系结构 (System Network Architecture, SNA)： (1) 在数据通信网络控制信息传送的体系结构。(2) 传输信息单元的逻辑结构、格式、协议、操作顺序、以及控制配置和网络操作的描述。

系统性能评估界面 (System Performance Measurement Interface, Spmi)： Performance Toolbox 中，允许应用程序注册有关它的性能或某些其它系统组件性能的定制性能统计信息的 Agent API。一旦注册，定制统计信息就对本 地或远程任何统计信息的消费者可用。它还允许应用程序在不使用接口的情况下访问在本地系统上的统计信息。这种应用程序称作本地数据消费者程序。

系统应用程序体系结构 FORTRAN (System Application Architecture FORTRAN, SAA FORTRAN)： ANSI X3.p - 1978 FORTRAN 77 标准的超集。

系统映象 (system image)： 程序 (及其相关数据) 的表示，自程序驻留在系统内存中时它就存在。

系统用户 (system user)： 使用计算机系统设施的人、设备或系统。

系统转储 (system dump)： 自所选数据区域的存储器的副本。与内核转储 (kernel dump) 同义。

系统资源控制器 (System Resource Controller, SRC)： 用于创建和控制子系统的一组命令或子例程。SRC 控制使用公用命令行和 C 界面的子系统进程。当您需要一个公用方法来启动、停止以及收集进程上状态信息时，SRC 非常有用。

系统 ROS (system ROS)： 负责装入引导映象的系统微码。

系统 (system)：计算机及其相关设备和程序。

细微值 (nice value)：用来使过程的优先级移位的数字。高优先级的数字决定低优先级。

下标量 (subscript quantity)：FORTRAN 中一个下标的组成部分。一个下标量是一个整数、实常数、变量或表达式。

下标声明符 (subscript declarator)：在数组定义或说明中，跟在数组名称后括在方括号里的表达式。指定了数组维数中的元素数目。

下标约束 (index constraint)：指 Ada 语言中某一数组类型的下标约束，它为该数组类型的每个下标范围定义下界和上界。

下标 (subscript)；**注脚**：(1) 一个整数或变量，其值在表或数组中选择特定元素。(2) 印于常规打印线下二分之一行的字符。

下行流 (downstream)：从流头到驱动程序的方向。

下行跳转 (downward jump)：对于 bfs 命令，从文件中当前位置移动到文件的底部或末尾的操作。

下级的 (inferiors)：在“增强的 X-Windows”中，在一个窗口下的所有子窗口。

下拉安装 (pull installation)：在网络安装管理环境中，从目标初始化的一种安装。

下拉菜单 (pulldown menu)：一种 MenuPane 小窗口，由于某用户操作（通常单击鼠标按钮），MenuPane 小窗口显示的样子好像是从 MenuBar 小窗口“下拉”来的。

下拉 (pulldown)：由于某用户操作（通常单击鼠标按钮），MenuPane 小窗口具有的一种显示方式，它的样子好像是从 MenuBar 小窗口“下拉”来的。

下限电阻 (tie-down resistor)：用于保存低级别的逻辑函数或门的输入的电阻。此寄存器的一端连接至地面或适当的负极电压。

下一步 (next)：相关工作站的对话管理操作，它显示对象数据的下一部分。

下载 (download)：从一台计算机上把数据传送到另一台上使用。通常，用户从一台较大的计算机上下载到软盘或另一台较小的计算机的固定磁盘上，或者从一个系统单元下载到另一个适配器。

先进先出 (first-in-first-out, FIFO)：在操作系统中，一种命名的永久性管道。FIFO 允许两个无关的进程通过管道连接交换信息。同先来先服务 (first-come-first-served) 同义。

先决条件 (prerequisite)：一种软件产品或服务更新，它必须在另一个软件产品或服务更新安装前被安装。如果您尝试在没有必需的必备软件情况下就安装软件产品或服务更新，则系统消息显示必需的必备软件名称。对照从属 (dependent)。

先来先服务 (first-come-first-served, FCFS)：通常情况下的一种队列技术，队列中的下一个要检索的项是已在此队列时间最长的项。与先进先出 (first-in-first-out) 同义。对照最短作业优先 (shortest-job-next)。另见规程 (discipline)。

弦 (chord)：图形中一条端点位于圆周上的一条短线段。弦是一种从直线制作圆周图形的方法。每个圆周上的弦数越多，圆周图形就越平滑。

显式绑定 (explicit binding)：绑定的一种形式，让客户机应用程序控制涉及到的服务器。

显示符号集 (display symbol set)：放在一张表中的显示符号集。共有多达 1024 种显示符号。显示符号 0 到 31 代表控制功能，没有图形表示。

显示符号 (display symbol)：一种预定义的可打印的符号（例如，字符、数字、数学符号、希腊字母等等），可以在图形显示器上显示。

显示会话 (display session)：一种“3270 主机连接程序 2.1 和 1.3.2” (HCON) 操作方式，在与仿真 3278/79 终端显示的主机进行交互期间的一种“3270 主机连接程序 2.1 和 1.3.2” (HCON)。

显示目录/对象 (display list/object)：在 GL 中，也叫做对象。这是绘图命令的序列，并且已经编译成为一个单元。从概念上来说，显示目录类似宏：通过简单引用它的名称，它可以多次被调用。通过适当的使用转换矩阵，可以在不同的位置、以不同的大小和方向对该对象进行实例化。例如，按螺母形状排列的多边形序列可以编译成为一个对象。通过调用其显示目录，这个螺母可以绘制多次。

显示屏 (display screen)：显示设备上可视地显示信息的部分。终端屏幕 (terminal screen) 的同义词。请参阅屏幕 (screen)。

显示器 (display)；**显示**：(1) 数据的可视化表示。(2) 以可视化的方式表示数据。(3) 在其上显示可视信息的计算机的输出屏幕。(4) 在“增强的 X-Windows”中，指一组由单个 X 服务器驱动的一个或多个屏幕以及输入设备。监视器 (monitor) 的同义词。

显示设备 (display device)：请参阅显示器 (*display*)；显示。

显示站 (display station)：一种输入 / 输出设备，包括操作员把信息发送到系统使用的键盘和操作员还可以看到发送到或接收来自计算机的信息的显示屏。

显示 PostScript (Display PostScript, DPS)：由 Adobe 创建的对 X 服务器的扩展。

显式 shell (explicit shell)：指由开发人员通过“创建”菜单的 Shell 子菜单手工创建的任何类型的 shell。

线程串行服务 (thread-serial service)：如果可重入系统服务阻拦当前线程和所有在第一个调用返回前，尝试调用同一个服务或其它相关服务的其它线程，它就是线程串行服务。另见可重入服务 (*reentrant service*)。

线程同步服务 (thread-synchronous service)：如果可重入服务仅阻拦当前线程，而允许其它线程在阻拦过程中执行相同操作，它就是线程同步服务。另见可重入服务 (*reentrant service*)。

线程 (thread)：AIX V4 中的可分派的实体。每个线程表示单个程序实例的当前执行状态。每个用户线程在特定进程提供的环境中运行，但是多重线程可以共享该进程所有的资源。

限定表达式 (qualified expression)：在 Ada 语言中，其类型或子类型的指示在前面的一种表达式。在以下情况下使用这样的限定：在缺少它时，在表达式可能不明确时（例如，结果重载时）。

限定符位 (qualifier bit)：请参阅 *Q 位 (Q-bit)*。

限定符 (qualifier)：(1) 用于标识另一个名称的唯一名称。(2) 使名称唯一的修饰符。(3) 限定名中除去最右边的名称（称为简单名）之外的所有名称。

限定类名 (qualified class name)：任何类名或具有一个或多个 :: (作用域) 运算符的类名。

限定类型名 (qualified type name)：用于减少复杂类名语法，方法是使用 typedef 定义来表示限定类名。

限定逻辑链路控制 (qualified logical link control, QLLC)：一种数据链路控制协议，它启用在 X.25 网络上的 SNA 到 SNA 通信。

限定名 (qualified name)：(1) 经过附加一个或多个限定符使之成为唯一的名称。(2) 一种数据名，它显式地伴有在指定的分类系统中该数据名所属类别的规范。(3) 在 C++ 中用于用其类名来限定诸如成员的非类类型名称。

陷阱处理程序 (trap handler)：当发生陷阱时使用的用户定义的陷阱例程。另见异常 (*exception*)。

陷阱 (trap)：一个未编程的、硬件引发的、到特定地址的条件转移。它因为错误或一些其它情况而发生。记录由跳转发生的位置组成。

线框 (wire frame)：一种图形表面绘制技术，其中图元的边缘和轮廓由简单的线条表示。

线路交换 (line switching)：电路交换 (*circuit switching*) 的同义词。

线路适配器 (line adapter)：一个将按位串行转换输入到一个站的功能单元。另见通信线路适配器 (*communications line adapter*)。

线路速度 (line speed)：(1) 数据通过一个远程通信线路从一个点传输到另一个点的速率。(2) 在远程通信线路上几秒钟可以发送的二进制位的个数，表示为位 / 秒 (bps)。速度 (*speed*) 的同义词。

线型 (linestyle)：画一条线的模式。一个线型可以是实线或者被分割成虚线模式。

线性插值 (linear interpolation)：通过假定数据值位于一条直线上来估计数据值的方法。典型的，两端的数据点是已知的。例如，如果 A 是 a 处的一个值，B 是 b 处的一个值，且 $a < t < b$ ，则在 t 处的值 C 是（通过两点公式）： $B - A$ 除以 $b - a$ ，乘以 $t - a$ ，加上 A。

现有文件 (existing file)：已定义并驻留在存储介质上的文件。

限制响应 (restricted response)：在 X.25 通信中，当限制响应为快速选择调用指定时，该调用必须已清除；它可能不被接受。

相对槽号 (relative slot number, RSN)：槽的号由相对文件中的一个记录占用。RSN 是用户数据记录的部分。相对文件中的每个记录包含保存其 RSN 的一个文件。

相对地址 (relative address)：(1) 相对于基地址指定的一种地址。(2) 相对于符号计数的一种地址。当程序重定位时，地址本身更改，但相对地址保留不变。

相对绘图命令 (relative drawing commands)：在计算机图形学中的一种命令，它相对于当前图形位置进行绘制，与绝对位置的绘制相反。

相对记录号 (relative record number)：表达记录的位置与包含它的文件中的基本位置关系的一个号。

相对路径名 (relative path name)：用从当前目录后根一个文件名的一个目录序列表示的目录或文件名。相对路径名不以一个 / (斜杠) 开始，但相对于当前目录。另见路径名 (*path name*) 和全路径名 (*full path name*)。

相对目录 (relative directory): 其名称以一个 / (点和一个斜杠) 开始的一种目录。

相对时间 (relative time): 通常用于添加到绝对时间或从绝对时间减去的一种离散时间间隔。

相对文件 (relative file): 一种文件, 作为固定长度槽的数组进行组织。记录可插入从文件的开始或结尾找到的第一个空闲槽, 或可明确地放置到文件中的某个槽中, 用其相对槽号 (RSN) 标识。

相对专有名称 (Relative Distinguished Name, RDN): 一组属性值断言 (AVAs), 每个都是“真”, 它关于特定条目的特异值。

相关行 (dependency line): 描述文件中每个条目的第一行。它包含一个目标文件列表, 后跟一个冒号, 还有一个可选的先决文件或相关性的列表。

相位调制 (phase modulation): 改变载波信号的相位, 以传达数据信号。

详尽类型说明符 (elaborated type specifier): 通常用于 C++ 的不完全类的声明中, 或者去限定另外要隐藏的类型。

详细描述 (elaboration): 在 Ada 语言中, 声明的详细描述是一个过程, 通过该声明来达到其目的 (比如, 创建一个对象); 该过程在程序执行中发生。

响应时间 (response time): (1) 等待数据通信系统响应请求的时间。例如, 如果您在 workstation 键盘上输入一个客户号, 则当按最后一个键时响应时间开始, 当您应答的第一个字符在 workstation 显示时响应时间结束。(2) 从启动一个操作到其启动程序有足够的信息继续之间的时间。

响应 (response): (1) 对一次查询的答复。(2) 在 SNA 中, 一种确认收据请求的消息单元; 响应通常由一个请求标题 (RH) 和 / 或应答单元 (RU) 组成。

向量积 (vector product): 向量交叉积的另一个名称。如果 $a = (a_1, a_2, a_3)$ 和 $b = (b_1, b_2, b_3)$ 是两个三维向量, 则向量积 a 乘 $b = (a_2b_3 - b_2a_3, a_3b_1 - b_3a_1, a_1b_2 - b_1a_2)$ 。

向量图形 (vector graphics): (1) 显示从坐标生成图像的图形, 这与像素数组相反。(2) 最普通的图形类, 其所有的向量输出由输出单元根据计算机指令点对点绘制的线和曲线构成。

向量 (vector): (1) 一个一维数组。(2) 通常由一组有序数字表现特征的数量。(3) 计算机图形中一个直线段。

橡皮筋轮廓 (rubber-band outline): 一种具有可移动轮廓的窗口。

像素图 (pixel map): 一个位的三维数组。一个像素图可以被认为是一个二维像素数组, 每个像素是一个零到 2 的 $N-1$ 次方之间的数值, 这时 N 为像素图的深度。像素图 (pixmap) 的同义词。

像素图 (pixmap): (1) 像素图 (pixel map) 的同义词。(2) 在 AIXwindows 和“增强的 X-Windows”中, 图标 (最初作为位图创建) 转换成的一种数据类型。在此转换后, 相应的 AIXwindows 子例程可通过引用 .Xdefaults 文件 (按名称) 和通过自变量列表 (按像素图) 生成像素图。另见映象高速缓存 (image cache)。

像素值 (pixel value): 在“增强的 X-Windows”中, 特定窗口或像素图中使用的位平面的数目。对于一个窗口, 像素值为彩色映射做索引并导出将要显示的实际色彩。一个像素是一个 N 位数值, 其中 N 为特定窗口或像素图中使用的位平面 (深度) 的数目。

像素 (pel): 请参阅图元 (picture element)。

像素 (pixel): 矩形图元。用于组成图像的最小元素, 单个点。显示屏幕由像素数组组成。在黑白系统中, 像素打开和关闭以形成图像。在彩色系统中, 每个像素有三个组成部分: 红、绿和蓝。每个组成部分的亮度可控制。另见图元 (picture element)。

项序列号 (entry sequence number, ESN): 对应次序的号码, 该次序是记录输入到项序列文件。项序列文件的主索引是基于其记录的 ESN。另见项序列文件 (entry-sequenced file)。

项序列文件 (entry-sequenced file): 一个面向记录的文件, 其记录按照输入到文件的前后顺序存储。项序列文件的主索引是基于项序列号 (ESN), 对应于记录插入顺序。也可以参考顺序文件 (sequential file)。另见项序列号 (entry sequence number)。

项 (entry): (1) 在 FORTRAN 中, 过程内的一种语言构造, 指定过程执行顺序的开始。(2) 在 Ada 语言中, 用于任务间通信的项。在外部, 仅当子程序被调用时, 项才被调用; 它的内部行为由一个或多个的 accept 语句来指定, 这些 accept 语句指定了当项被调用时执行的操作。(3) 在表、列表、队列或其它数据的组织结构或控制信息里的元素信息。(4) 在工作站上的单个的输入操作。

项 (term): 作为另一个表达式子区段的表达式, 通常通过添加或删减, 由其它子区段构成。

消零 (zero suppression): 删除或用空格替代数字中开始几位的零。例如, 当使用消零时 00057 就变成 57。

消息队列标识符 (message queue ID, msqid): 指定到一个消息队列, 并用于一个特定进程的一个标识符。在应用上类似于一个文件的文件描述符。

消息队列 (message queue)： 连接到一个“队列”的一个消息链接列表。

消息控制块 (message control block)： 在 X.25 API 中，用来表明什么类型的包到达了，以及所到达的包含包信息的结构。

消息类型 (message type)： 标识了一条消息的内容的已定义的一系列值。

消息 (message)： (1) 来自于系统说明了可能影响当前系统下一步处理的一种情形的信息。(2) 一个错误指示，或者任何简短信息，它被一个程序写到一个标准错误和一个队列中。(3) 一个多用户操作系统的一个用户向另一个系统发送的信息。(4) 两个进程间通信的一种常规方法。(5) 一组作为一个实体来传输的字符和控制位序列。(6) 数据或者信息的一个或者多个数据链路块，关联的 STREAMS 控制结构包含一种消息类型。消息仅仅意味着一个流之间的通信。

小窗口程序员 (widget programmer)： 将新的小窗口添加至“增强的 X-Windows”（或其它）工具箱的程序员。

小窗口标识 (widget ID)： 与界面中例示的每个小窗口相关的唯一标识号码。

小窗口菜单 (widget menu)： 允许用户对所选小窗口执行任何数量的操作（比如剪切、复制和粘贴）的菜单。

小窗口层次结构 (widget hierarchy)： 与小窗口树 (widget tree) 同义。

小窗口记录 (widget record)： 与任何给定小窗口相关的相关数据对象的集合，比如变量和参数。另见实例记录 (instance record)、记录 (record)、类记录 (class record) 和超类 (superclass)。

小窗口可见性 (widget visibility)： 对照模糊 (obscure) 和堵塞 (occlude)。

小窗口类型 (widget type)： 小窗口类 (widget class) 的同义词。

小窗口类 (widget class)： 特定小窗口隶属的常规组，另外也称作小窗口类型。物理上，它是指向一个结构的指针。与小窗口类型 (widget type) 同义。另见类 (class)。

小窗口实例 (widget instance)： 不同于常规小窗口类的特定小窗口对象。它由一个包含特定于实例的值的的数据结构和另一个包含可用于该类的所有小窗口的信息的数据结构构成。另见实例 (instance)。

小窗口树 (widget tree)： (1) “增强的 X-Windows 工具箱”代码的符号结构。基本元素就是一个小窗口类。另见叶 (leaves)、中间节点 (intermediate nodes) 和根 (root)。(2) 小窗口树是特定客户机应用程序内的小窗口

层次结构。**Shell** 小窗口是小窗口树的根。没有任何种类子小窗口的小窗口是树的叶。与小窗口层次结构 (widget hierarchy) 同义。

小窗口引力 (widget gravity)： 窗口引力 (window gravity) 的同义词。

小窗口 (widget)： (1) “增强的 X-Windows 工具箱”的基本数据类型。(2) 提供用户界面抽象的对象；例如滚动条 (Scrollbar) 小窗口。它是“增强的 X-Windows”（或子窗口）和其相关的语义的组合。逻辑上，它是具有具有相关输入和输出语义的矩形，虽然有些可能是仅输入或仅输出。每个小窗口属于一个小窗口类。小窗口通过它的小窗口类结构执行过程。另见复合小窗口 (composite widget)、核心小窗口 (core widget)、图元小窗口 (primitive widget) 和 shell 小窗口 (shell widget)。(3) 小窗口一种能够从键盘和鼠标接收输入并通过回调和应用程序或另一个小窗口来通信的图形设备。每个小窗口仅是一个类的成员，并始终有一个与之相关的窗口。

小配件标识： 指 AIXwindows 中指定给每个小配件实例的在给定图形界面中使用的唯一标识号。

小配件 (gadget)： 指 AIXwindows 中的一种无窗口图形对象，看起来像其等价物——具有类似名称的小窗口 (widget)，但不支持该小窗口所提供的转换、操作或弹出式子小窗口。

小体大写字母 (small caps)： 请参阅大写字母 (caps)。

小尾数法次序 (little endian order)： 一种存储方法，在该方法中整数值首先存储在具有较小意义的字节。另见大尾数法规则 (big endian order)。

小尾数法 (little endian)： 数据表示的一个属性，反映了多八位元 (multi-octet) 数据在内存中是如何存储的。在小尾数法表示法中，一个多八位字节数据项的最低八位地址意义是很小的。另见尾数法 (endian) 和大尾数法 (big endian)。

小型计算机系统接口适配器 (Small Computer Systems Interface Adapter, SCSIAdapter)： 支持将各种直接访问存储设备磁带机连接至系统设备的适配器。

小字 (small word)： vi 编辑器中一组连续的字母数字字符，它们至少在一端与不是空白、跳格、或新行指示符的字符绑定在一起。例如，在字 isn't 中，字符的两个集 isn 和 t 是小字。对照大字 (big word)。

协处理器 (coprocessor)： (1) 一个与另一个处理器协同执行运算的辅助处理器。(2) 在个人计算机中，一个扩展主板上的微处理器，扩展系统部件中处理器的地址范围，或添加处理特殊运算类别的专门指令。

协调 (tuning)：请参阅人工协调 (*hand tuning*)。

协商 (negotiation)：在 X.25 通信的进程，通过它两个 DTE 建立要用于呼叫过程的数据包的大小、数据包窗口大小与吞吐量类。对照确认 (*validation*)。

协议边界 (protocol boundaries)：SNA Server LU6.2 支持的 SNA 动词集。

协议端口 (protocol port)：在主机内由传送协议使用的用于指定目的地的一个唯一主机标识。

协议系列 (protocol family)：一组相关通信协议；例如，国防部的因特网协议。一个协议系列的所有成员都使用一个公共寻址机制来标识端点。与地址系列 (*address family*) 同义。另见套接字地址 (*socket address*)。

协议 (protocol)：(1) 在 SNA 和 SNA Server 中，管理网络、传送数据以及使网络各组件的状态同步所使用的请求及响应的含义及顺序规则。(2) 一组确定在实现通信时各功能部件的行为的语义和语法规则。(3) 客户机间互相一致同意的实现某些操作的通信机制。

写队列 (write queue)：在模块或驱动程序中包含向下游移动的消息的消息队列。这与来自用户进程的输出相关。

写屏蔽 (writemask)：8 位或 12 位的集合（取决于帧缓冲区的配置），每个帧缓冲区的位平面一位。在任何绘制操作的过程中，只有被位屏蔽中的 1（一）启用的那些平面可以更改。设置为 0（零）的平面标记为只读。

写入时拷贝 (copy-on-write)：一个选项，创建将更改保存在系统页面调度空间里的映射文件，而不是将文件副本的更改保存在磁盘。

写验证 (write verification)：系统在执行写操作之后自动执行读操作的方式。然后它比较数据以确定它们是相同的。

新过程映象 (new-process image)：通过 **exec** 子程序覆盖当前程序的新建程序。

信号标识 (semaphore ID, semid)：指向包含信号信息的信号集或数据结构的整数。

信号处理程序 (signal handler)：当信号出现时调用的子例程。

信号堆栈 (signal stack)：可以在其上处理信号的备用堆栈。

信号屏蔽 (signal mask)：定义当前不能传送至进程的信号集。

信号 (semaphore)：(1) 用于控制对系统资源访问的实体。如果进程遵循某种编程约定，则进程可能被锁定至具有信号的资源。(2) 它提供了两个进程间通信的一般方法，这些进程是信号功能部件的扩展。

信号 (signal)：(1) 两个进程间通信的简单方法。当事件发生时，一个进程能够通知另一个进程。(2) 操作系统操作中模拟软件中断的进程间通信的方法。对照异常 (*exception*) 和中断 (*interrupt*)。

信息包窗口 (packet window)：在 X.25 通信技术中，允许的尚未确认的包的数目。另见帧窗口 (*frame window*) 和窗口 (*window*)。

信息包大小 (packet size)：在 X.25 通信技术中，在数据包的上下文中指用户数据的长度。

信息包方式操作 (packet mode operation)：与包交换技术 (*packet switching*) 同义。

信息包级别接口 (packet-level interface)：在 X.25 通信技术中，以信息包方式操作的 DTE/DCE 接口的级别，与数据和信号交换有关，这里，这些信息包含在信息包中。另见帧级别接口 (*frame-level interface*)。

信息包级别 (packet level)：在 X.25 通信技术中，包交换的包格式和控制过程包含在数据终端设备 (DTE) 和数据电路终端设备 (DCE) 间的控制信息用户数据。与级别 3 (*level 3*) 同义。另见级别 (*level*)、帧级别 (*frame level*)、数据链路级别 (*datalink level*) 和物理级别 (*physical level*)。

信息字段溢出 (information field overflow, I-field overflow)：发生在当接收数据的信息字段大小超出主站的缓冲区容量时的情形。一些信息字段丢失。

形式参数 (formal parameter)：在例程的报头中声明的参数。它指定什么能作为实参数传递给例程。另见构造字符串 (*conformant string*) 与哑参数 (*dummy argument*)。对于 Ada 编程，另见参数 (*parameter*)。对照实参数 (*actual parameter*)。

兄弟 (sibling)：同一个父窗口的子窗口。

修订版号 (modification number)：一个程序的修改级别，它是不影响程序外部接口的变化的一个指示符。版本、发行版、修订版、和修正级别，包括程序级别 (*program level*)、修正号 (*fix number*)、发行版号 (*release number*) 和版本号 (*release number*)。

修订版本 (revision text)：见版本标识 (*version identifier*)。

修订控制系统 (Revision Control System, RCS)：管理文本文件的多重修订。它设计为控制频繁修订的文本，

例如程序、表单信件和论文。它具有自动标识、存储、记录、检索及合并文件修订的功能。另见源代码控制系统 (Source Code Control System)。

修剪 (pruning)： 在 GL 中，因边界框测试显示它们不可见而将显示列表中的部分消除不画。另见挑选 (culling)。

休眠进程 (sleeping process)： 等待输入或输出完成、时间片、事件发生或来自其它进程的信号的进程。当一个进程处于休眠时，它可以调页至内存外。

修饰符键 (modifier key)： 在 CDE 中，当按下并保持另一个键或鼠标按钮按下时能改变第二个按键或鼠标点击的意思的键。例如，“控制”键 (Control)、“选择”键 (Alt) 和“切换”键 (Shift)。

修饰符键 (modifier keys)： 在“增强的 X-Windows”中，诸如 Shift、Shift Lock、Control、Alt、Caps Lock 和 Meta 等键。

修饰符 (modifier)： 用于改变一条指令以引起指令的执行不同于最初的指令的字或者限定词。因此，同样的指令，通过一个修饰符的连续改变，可以用于重复执行一个每次使用都不相同的操作。

修正号 (fix number)： 程序的修正级别，是小的更新的指示符，这些更新将构建到常规修订或以后的发行版中。版本、发行版、修订和修正级别一起组成程序的程序级别或版本。另见程序级别 (program level)、修订版本号 (modification number)、发行版本号 (release number)、版本 (version) 和版本号 (version number)。

修整环 (trimming loops)： GL 中用于设置 NURBS 平面边界的一组定向闭合曲线。另见 NURBS。

虚拟安装点 (virtual mount point)： 指在文件树中的目录或文件，在其位置安装另一个系统。例如，/dev/hd9 安装在 /fred 上，那么 /fred 就是虚拟安装点。

虚拟存储扩展 (Virtual Storage Extended, VSE)： DOS 扩展的一个操作系统。由许可 VSE / Advanced Functions 支持和要求满足用户的数据处理的需求的所有程序构成的 VSE 系统。VSE 与它控制的硬件一起形成了一个完整的数据处理系统。它的当前版本称作 VSE/ESA。

虚拟存储 (virtual storage)： 是虚拟内存 (virtual memory) 的同义词。

虚拟打印机定义 (virtual printer definition)： 描述特定打印机的特定数据流的属性值集合。

虚拟打印机 (virtual printer)： 打印机的一种视图，它仅指打印机理解的高级数据流 (比如 ASCII 或 PostScript)。

它不包括关于如何将打印机硬件连接至主机计算机或连接至用于传送到打印机或从打印机传送数据字节的协议的任何信息。

虚拟地址 (virtual address)： 虚拟存储中位置的地址。虚拟地址必须转化为真实地址，以便使 CPU 能够访问由它寻址的数据和指令。

虚拟电路 (VC)： 在 X.25 通信中，指网络提供的那些设施，它们给用户一个真实连接的外观。另见交换式虚拟电路 (switched virtual circuit) 和永久虚拟电路 (permanent virtual circuit)。

虚拟函数 (virtual function)： 用关键字 virtual 声明的 C++ 成员函数。当您调用虚拟函数执行的实现取决于其调用对象的类型。这在运行时确定。

虚拟呼叫设施 (virtual call facility)： 数据通信中的用户设施，其中的呼叫建立过程和呼叫清除过程确定了两个数据终端设备 (DTE) 之间的通信周期 (其中用户数据以操作的包方式在网络中传送)。所有用户数据从网络传送，以通过网络接收它。它是等同于已拨号的线路的包网络。

虚拟键绑定 (virtual key binding)： AIXwindows 中用户设计的键定义。

虚拟内存 (virtual memory)： 看起来像真实存储的可寻址空间。指令和数据从虚拟内存映射至真实存储位置。虚拟存储的大小受到计算机系统寻址方案以及可用辅助存储器数量的限制，但不受系统内存位置的实际数量的限制。对照实内存 (real memory)。与虚拟存储 (virtual storage) 同义。

虚拟设备驱动程序 (virtual device driver, VDD)： 是设备处理程序 (device handler) 的同义词。

虚拟设备 (virtual device)： 在用户看来是一个独立实体的设备，但事实上它是真实设备的一个共享部分。

虚拟文件系统 (virtual file system, VFS)： 为了让本地用户访问而安装的远程文件系统。

虚拟远程通信访问方法 (Virtual Telecommunication Access Method, VTAM)： 控制 SNA 网络中通信和数据流的许可程序。它提供了单域、多域和互连网络能力，且还支持应用程序和子系统。

许可程序 (LP) (licensed program, LP)： (1) 一个保持制造者属性的软件程序，客户向该制造者付许可证费。(2) 一个单独的定价的程序和它的相关材料，这些材料有一个版权，在许可证协议的期限和条款下提供给客户。

许可代码 (permission code)：用于表示访问许可的一个三位数字的八进制代码或一个九字母的字母代码。访问许可即读、写和运行。另见访问许可 (*access permission*)。

许可证服务器守护程序 (license server daemon)：一个管理软件产品许可证的软件程序，它由 **netlsd** 调用。该 **netlsd** 命令可以在 **/usr/lib/netls/bin** 目录下找到。

许可证密码 (license password)：一个用软件产品的许可证信息编码的字符串。

许可证数据库 (license database)：由许可证服务器维护的许可证数据库。该许可证数据库文件 **lic_db** 驻留在 **/usr/lib/netls/conf** 目录中。

许可证信息 (license information)：描述许可证的有关信息。该信息由产品名称、产品版本、许可证号许可证类型、许可证的开始和结束日期、目标类型、目标标识和一个时间戳记组成。

许可证注释 (license annotation)：以一种由软件产品供应商定义的方式修改一个许可证的使用的一个特殊的数字字符串。

许可证 (license)：一个使用许可的软件产品或服务的许可实例。有时，一个用户要使用一个产品需要超过一个的许可证。

许可字段 (permission field)：位于目录列表的许可列中的一个三字符字段。许可字段为文件或目录的所有者、为组和为所有其它人指明读、写和运行许可。

许可 (permission)：访问受保护对象的方式。

许可 (permissions)：确定在该系统上工作的任何用户如何使用一个文件的代码。在公共桌面环境中，确定用户对文件和目录的访问的标志集，您可使用“文件”菜单上的**属性...**命令查看它。

序数类型 (ordinal type)：一种类型，它的成员能够计数以表示位置。

序数 (ordinal number)：计数的一种，一般用来表示位置。

悬空 else (dangling else)：由 IF-ELSE 语句的 IF 部分嵌套 IF 语句时而导致的情况。ELSE 语句与最近的 IF 语句相关联，在这里也就是内部的 IF 语句。把一句空 ELSE 语句放置在嵌套语句中，可通过强制外部 ELSE 语句与外部 IF 语句相关联而防止错误解释。

旋转 (rotation)：在计算机图形学中，通过绕轴旋转它转换图元。另见变换 (*transformation*)。

选定部件 (selected component)：在 Ada 语言中，指由前缀和称为“选择子”的标识符构成的名称。选定部件用于表示记录部件、条目和由存取值指定的对象；它们也可用作扩展名称。

选项按钮 (option button)：在公共桌面环境中，菱形按钮允许从选项号中选择。

选项 (choice)：弹出式窗口或菜单里的选项，用来影响系统操作。

选项 (option)；**选件**：(1) 除了基本系统外，购买硬件或者软件的物品。选项可以包含电缆、适配器、保修和其它项。(2) 语句中的规范可以影响正在运行的语句。(3) 软件包的可安装单元。软件产品选项是单独的可安装单元，它可以从软件包的其它选项单独的操作。

选择范围 (selection range)：请参阅**键范围 (key range)**。

选择区域 (selecting region)：在 GL 中，指世界坐标中的菱形体，它对选择事件很敏感。如果在此区域内绘制绘图图元，将报告选择事件。另见**命中 (hit)**、**选择 (selecting)**、**拾取 (picking)**、**变换 (transformation)** 和**拾取区域 (picking region)**。

选择区域 (selection area)：在 AIXwindows 中，指 RowColumn 小窗口的一部分，鼠标指针可以置于其上以选择其它小窗口。

选择条件 (selection criteria)：在 **select** 子例程中，指 **readlist**、**writelst** 和 **exceptlist** 参数值，它指定了用什么来检查读、写以及异常。

选择子 (selector)：(1) 在 Pascal 中，指在 CASE 语句中的项，一旦被求值，它确定了 CASE 语句中可能要处理的分支。(2) 对于 Ada 编程，请参阅**选定部件 (selected component)**。

选择 (select)：(1) 在显示器屏幕上选择一个按钮。(2) 将光标置于一个对象（名称或命令）然后按鼠标“选择 (Select)”键（左）或按键盘上的“选择 (Select)”键。(3) 指出您选择的下个命令会影响的一个项或多个项。该项可能会突出显示来确认您的选择。选择并非实际执行命令。

选择 (selecting)：在 GL 中，指在三维空间的给定卷中查找正在绘制的图元。另见**命中 (hit)**、**选择区域 (selecting region)**、**拾取 (picking)** 和**拾取区域 (picking region)**。

选择 (selection)：(1) 在可选的调用回路上为一个工作站或组件指定地址。(2) 计算机请求一个站发送给它消息的过程。(3) 另见**寻址 (addressing)**。(4) 在“增强的 X-Windows”中，指由客户机（所有者）所有但是属于用

户的动态类型的间接属性。它并非某个特定窗口子层或特定客户机集所专有。当客户机请求选择内容时，它指定一个目标类型。此目标类型可以用来控制已传输的内容表示。

循环的非因式分解 (loop defactorizing)：一种优化从循环中除去不变量因子，此循环求和计算为一个标量。在退出系统时求和标量和因子相乘。

循环的取消 (loop elimination)：循环完全被展开的一种循环展开方式，并引用循环指数，其中的展开循环由常量值替代。

循环的再实现 (loop rerolling)：一种优化，是将用户展开的循环变换为原始的未展开的循环，使得其它的优化得到尝试。

循环的展开 (loop unrolling)：一种优化，是为了增加循环的步数，并复制循环内的表达式以反映步数的增加。这可以提高指令调度和内存访问时间。

循环合成 (loop fusion)：一种优化，将循环体进行恒等迭代计算并将它们合成为一个循环。

循环毁坏 (loop collapse)：在嵌套数组处理循环中的一种优化，这个优化将嵌套循环毁坏为单个循环，采用的迭代计数是原始循环的迭代计数，并适当的调整了数组索引。

循环开销 (loop overhead)：不能归因于循环计算的循环所消耗的 CPU 时间。

循环嵌套记录 (loop nest reordering)：一种优化，改变循环嵌套中循环的顺序，得到极小值或消除数据相关性。

循环去皮 (loop peeling)：一种优化，用于提高循环映射数组的性能，将数组映射到柱坐标系。

循环冗余校验字符 (cyclic redundancy check character)：经修改的循环码中使用的字符代码，用于错误检查和改正。

循环语句 (looping statement)：一种语句，可运行任何次数，依赖于指定表达式的值。

循环 (loop)：(1) 一序列指令重复的执行直到到达结束条件。(2) 一个封闭的无向信号路径将输入和输出设备连接到系统上。

寻址 (addressing)：(1) 在数据通信中，发送或控制工作站选择要向其发送数据的工作站的方法。(2) 一种识别存储位置的方法。(3) 在文件内指定一个地址或位置。(4) 程序指令的地址赋值。另见选择 (selection)。

[Y]

哑参数 (dummy argument)：在 FORTRAN 中，在一个子程序或语句函数定义中的变量，来自程序或函数引用的调用的参数与位置相关联。请参阅形式参数 (formal parameter)。

压缩打印 (condensed print)：一种字符变小，空格在水平方向上更紧凑（通常密度为每英寸 17 个字符数）的打印格式。

压缩输出 (compressed output)：与压缩 (compression) 同义。

压缩 (compress)：(1) 把磁盘上的文件和库移动到一起以创建未使用空间的一个连续区域。(2) 在数据通信中删除字符串中的一系列重复字符。

压缩 (compression)：(1) 一种在发送数据之前除去重复字符、空白、空字段和尾部空格的字符串的技术。与压缩输出 (compressed output) 同义。(2) 在 SNA 中，用一个经过编码的控制字节替换长达 64 字符的字符串，以减少发送给 LU-LU 会话伙伴的数据流长度。

压纸卷筒 (platen)：打印机上纸张的支持机制，通常是圆柱形的，打印机制对它进行撞击以产生印痕。

延迟端口 (delayed port)：该端口象共享端口一样被启用，除非登录提示直到您输入一个或多个字符（通常是回车符）后才被显示。直接连接到远程系统或智能调制解调器的端口通常被作为延迟端口启用。

严重性代码 (severity code)：指明错误情况有多严重的代码。

严格类型检查 (strict type checking)：比起 C 编译程序，依照 C 语言更严格地检查数据类型，比如用 lint 程序。

颜色表 (color table)：与色彩图 (color map) 同义。

颜色查看器 (color viewer)：AIXwindows 中，一个允许用户为小窗口属性选择颜色值的界面。

颜色查找表 (color lookup table)：与色彩图 (color map) 同义。

颜色扩展运算 (color expansion operation)：一种图形编程运算，当源像素数据区域每像素仅包含 1 字节而目标像素图数据区域为彩色显示适配器缓冲区帧（定义为每像素多于 1 位）时自动发生。

掩码 (mask)：(1) 控制保留、删除或测试另一字符或位模式的字符模式，通常通过 AND 或 OR 操作来完成。(2) 应用掩码。

验证 (validation) : X.25 通信中, 接收 DTE 的接受发送 DTE 发送的包大小、包窗口大小以及吞吐量类的进程, 前提是它们都是合法的。对照协商 (*negotiation*)。

验证 (verify) : (1) 即确认正确性。(2) 确定数据的抄录或其它操作是否已经正确完成。(3) 在软件安装中, 验证过程指导验证您正在安装的软件。系统确认您的软件文件为正确长度, 且包含正确数量的数位与字符。如果报告任何错误, 可能需要再次安装软件产品。验证过程可能会为安装过程添加大量时间。

样式管理器 (Style Manager) : CDE 中用于定制一些虚拟元素和工作区环境的系统设备行为 (包括颜色和字体、键盘、鼠标、窗口和会话启动行为) 的一种软件应用程序。

页脚 (footer) : 显示在文档每页底部的文本, 例如, 页码。

页框表 (page frame table) : 包含在实际内存的表, 包含所有当前定义的页面的实际内存位置。

页码 (folio) : 一种页码格式, 它把页码放在该页顶端的运行头部的外面。另见盲页码 (*blind folio*)、断开页码 (*dropped folio*) 与快速页码 (*expressed folio*)。

页眉 (header) ; 头; 报头: (1) 格式化在某个文档中的打印页面顶端空白处的固定文本。(2) 放在用户数据之前的系统定义的控制信息。(3) 报文的一部分, 包含该报文的控制信息, 例如目标字段、始发站和优先级。(4) 在 CDE 中的邮件程序中, 当某条电子邮件消息出现在邮件程序容器中时, 有关该消息的基本信息。报头显示发送方的名称、主题、接收日期和时间以及报文大小。

页面调度 (paging) : (1) 在实存储容量和外部的页存储量间传送指示信息、数据、或两者的操作。(2) 当需要时在内存和大容量存储器设备间移动数据。(3) 在 System/370 虚拟存储系统中, 在实存储容量和外部页存储量间的页面传送过程。

页面空白 (margin) : 在屏幕或硬拷贝页面上的文本左右边界。

页群集 (page cluster) : 内存缓冲区的一种类型, 它从完整内存页面 (通常为 4096 字节) 构造。

叶项 (leaf entry) : 一个没有下层的目录项。它可以是一个别名项或一个对象项。

页帧 (page frame) : (1) 在实存储容量中, 有页面大小的存储位置。(2) 主存储器的一个区域包含一个页面。(3) 用来保留虚拟内存页的在实际内存中的连续的 4096 字节部分。

叶 (leaf) : 一页文本。另见页面 (*page*)。

叶 (leaves) : 在一个小窗口树上, 没有子类的小窗口。另见小窗口树 (*widget tree*)。

页 (page) : (1) 指令、数据、或两者的块。(2) 可被放进窗口的行数。(3) 在虚拟存储系统中, 为有虚拟地址的并被作为一个单元在实存储容量和辅助存储间传输的固定长度的块。(4) 虚拟内存段的连续的 4096 字节部分。从段开始的每个页面的偏移量为完整的 4096 的倍数。另见叶 (*leaf*)。

一次性使用许可证 (use-once license) : License Use Management 中的一种协议类型, 它由许可证服务器管理, 可用于调用产品或使用服务的单个实例。产品每使用一次, 许可证服务器就减少一次性使用许可证的数量。

一级中断处理程序 (first-level interrupt handler, FLIH) : 接收系统控制作为硬件中断的结果的例程。六个中断级别的每一个都分配一个 FLIH。

依赖 (dependent) : 一个软件产品, 需要在安装前或安装的同时安装另一个产品或更新。对照先决条件 (*prerequisite*)。

依赖 (dependents) : Ada 语言编译单元, 如果其它单元将被重新编译, 那么它们也将被重新编译, 因为 Ada 语言已经规定了编译顺序。

一元表达式 (unary expression) : 包含一个操作数的表达式。

一元运算符 (unary operator) : 表示一个操作数上的一个操作的运算符。对照二进制运算符 (*binary operator*)。

一致 (consistent) : 指某个文件系统, 没有内部差异。

移出 (yank) : 将文本的字或行复制到内存中。

已定义状态 (defined state) : 某设备在其已定义的方法运行时或可用设备的取消配置方法运行时所处的状态。设备在此时不是可用设备。

已格式化的软盘 (formatted diskette) : 磁道与特定计算机系统的控制信息已写入的软盘, 但它可能包含或不包含任何数据。

已格式化的数据 (formatted data) : 在主存储器与输入/输出设备之间按指定的格式传送的数据。另见直接显示数据 (*list-directed data*) 与格式 (*format*)。

已禁用端口 (disabled port) : 在“异步终端仿真”(ATE) 中表明端口可以呼出的端口配置。

以太网 (Ethernet) : 一种使用 CSMA/CD (带冲突检测的载波侦听多路访问) 的 10M 基带局域网。这种网络允许

多个站点随意访问网络介质而无需事先协商，它通过使用载波侦听和延迟来避免争用，并使用冲突检测和传输来解决争用。

已映射的 (mapped)： 在“增强的 X-Windows”中，如果窗口上一个映射被调用，称该窗口已映射。

以 null 结束的 (null-terminated)： 末尾是零字节。在 C 语言中，用这种方法在内部存储字符串。

异步操作 (asynchronous operation)： 一个操作自己不会导致进程请求阻拦该操作进一步使用 CPU。这意味着进程和操作是并发地运行的。

异步传输 (asynchronous transmission)： 一种数据传输，可在任何时刻开始传输字符或字符块，但其中代表字符或块的位具有相同的时间间隔。对照同步传输 (*synchronous transmission*)。另见起止式 (传输) (*start-stop*)。

异步设备 (asynchronous device)： 数据传输中使用的设备，可在任何时刻开始传输字符或字符块，但其中代表字符或块的位具有相同的时间间隔。

异步终端仿真 (Asynchronous Terminal Emulation)： 提供远程异步终端仿真的程序。

异步终端 (asynchronous terminal)： 一种计算机终端，使用异步信号与主机通信。

异步 (async)： 请参阅异步传输 (*asynchronous transmission*)。

异步 (asynchronous)： 在时间上不同步。例如，输入事件由用户控制；程序可以在稍后读取它们。

异常处理程序 (exception handler)： (1) 用于检测死锁状态或处理非正常状态的例程集。异常处理程序允许进程的正常运行中断和恢复。(2) 异常处理程序是 C++ 中的 **catch** 块。**catch** 块在异常从封闭在 **try** 块中抛出时捕捉到它们。**try** 块、**catch** 块和 **throw** 表达式是 C++ 中用于实现正式异常处理的构造。

异常处理 (exception handling)： 错误处理的一种，它允许在异常发生时把控制和信息传递给一个异常处理程序。**try** 块、**catch** 块和 **throw** 表达式是 C++ 中用于实现正式异常处理的构造。

异常结束 (abend)： (1) 任务的异常结束。(2) 因为出错情况而导致任务在完成前结束，恢复设施无法在任务运行时解决错误。

异常终止 (abort)： 该术语指事务没有完成。任何由异常终止的事务所做的更改，不管出于何种原因，必须撤销。

一旦事务撤销 (回滚)，在事务处理系统日志里不会留下该事务曾被尝试但仍无记录的证明。另见回滚 (*rolled back*)。

异常 (exception)： (1) 程序设计语言中，可能在程序运行过程中产生的非正常情况，它可能造成对正常运行次序的偏离，并且存在针对异常的处理程序。(2) 非正常状态，例如在处理数据集或文件时遇到的 I/O 错误。(3) 在浮点异常中可能发生的五种类型的错误之一。它们是：操作不合法、上溢、下溢、被零除和不精确结果。(4) C++ 语言中，被函数检测到的任何用户、逻辑或系统错误，函数本身并不处理错误，而是把错误传递给一个处理例程。传递错误称为抛出异常。(5) Ada 语言中，一种可能在程序执行过程中产生的错误状态。引发一个异常是指放弃正常的程序执行以便发出错误发生的信号。异常处理程序是程序正文的一部分，它指定了对异常的响应。这种程序正文的执行称为处理异常。对照中断 (*interrupt*) 和信号 (*signal*)。另见处理程序 (*handler*) 和陷阱处理程序 (*trap handler*)。

溢出 (overflow line)： 指定页面上被打印的行为最末行。

溢出情况 (overflow condition)： (1) 当运算结果的一部分超出存储器的目标单元的容量时出现的情况。(2) 当页面上的溢出已经打印或传递时出现的情况。

溢出区域 (spill area)： 用于保存寄存器内容的存储区域。

溢出 (overflow)： (1) 运算结果的一部分超出执行存储单元的容量。(2) 在寄存器中，丢失最左边整数数字的一个或多个，是因为运算的结果超出寄存器的大小。

意向锁 (intention locks)： 用于分层资源的锁定，它允许应用程序避免锁定大型数据对象，例如记录文件，当仅需要修改数据对象的一部分时，如文件中的一个记录。获得文件的意向锁定表明文件中的一些记录可以在当前进行修改，但是允许访问没有被锁定访问权限的记录。

阴极射线管 (cathode ray tube, CRT)： 一真空显象管，其中电子束可以在其发光屏幕上移动绘线或字符或符号。

因特网地址 (Internet address)： 一种编号系统，用于 TCP/IP 互连网络通信中，在进行通信的网络上指定一个特定的网络或特定的主机。因特网地址采用通常用总分十进制形式表示。

因特网 (Internet)： (1) 一种连接成千上万个在工业、教育、政府和研究机构中的离散网络的广域网。因特网采用 TCP/IP 作为传输信息的标准。(2) 任何连接多于一个网络的广域网。

阴影小窗口 (shadow widget) : 指向某结构的不透明的指针, 每当创建小窗口时创建此结构; 它在界面中标识小窗口。又称为 *swidget*。

阴影颜色 (shadow color) : 对话框周围或对话框后边的阴影区。

阴影 (shadow) : 窗口下边和右边或上边和左边的变暗区域, 它代表窗口可能创建的阴影。

因子化 (factorization) : 一个算术变换, 公因数除以一个表达式中的子表达式, 然后乘以结果表达式。

隐藏表面 (hidden surface) : 几何图元的一个表面, 因被其它表面遮掩而不可见。另见 *z 缓冲器 (z-buffer)*。

隐藏文件 (hidden file) : 不通过目录清单显示的操作系统文件。隐藏文件的名称通常以句点开始。

引导程序块 (bootstrap block) : 与引导块 (*boot block*) 同义。

引导程序 (bootstrap) : 系统初始化过程中, 一个装入大程序的小程序。

引导处理 (boot processing) : 当引导映象装入后发生的处理类型。引导设备类型 (磁盘、磁带或网络) 决定了发生的引导处理类型。取决于配置文件, 网络引导处理使机器进入运行状态、BOS 安装状态或诊断状态。

引导符 (leaders) : 一个用目录中以将视线引导从标题页码均匀分隔的点行。

引导块 (boot block) : 文件系统中引导程序驻留的第一个块。另见引导块 (*bootstrap block*)。

引导设备 (boot device) : 根卷组 (rootvg) 中指定将包含启动 (引导) 映象的硬盘的设备。

引导映象 (boot image) : 一包含内核、文件系统、库和程序的映象。引导映象在机器开机或复位之后装入, 然后使机器进入运行状态。

引导 (boot) : 请参阅初始程序装入 (*initial program load*)。

引发异常 (raising an exception) : 对于 Ada 编程, 请参阅异常 (*exception*)。

隐含 DO: 指 FORTRAN 中的一种索引规范, 带有数据元素列表而不是一组语句作为其范围。类似于 FORTRAN DO 语句, 但不使用单词 DO。

隐秘性 (privacy) : 一种保护级别, 它可在安全 RPC 通信中指定并加密 RPC 自变量值。

隐式类型转换 (implicit type conversion) : 由于不同类型变量之间的关联而使编译器生成的类型转换。例如, 将浮点数组初始化成循环计数器的值就是从整数到浮点的隐式类型转换。

引用计数 (reference count) : 在 i-node 中, 引用 i-node 的目录项的总数的一个记录。

引用监控器 (reference monitor) : 控制对某个对象的访问的代码。

引证 (quote) : 掩码某些字符的特殊含义, 导致这些字符按文字取。

英寸 (inch, in) : 2.54 厘米或 25.4 毫米。

应答 (reply) : (1) 对查询的响应。(2) 在 SNA 中, 仅仅是因为对接收的请求单元作出反应而发送的接收请求单元。例如, “停顿完成”是在接收“在链结束停顿”后的应答。(3) 在“增强的 X-Windows”中的一种方式, 其中由某客户机程序请求的信息被回送给该客户机。事件和应答都在同一连接上多路复用。大多数请求不生成应答; 某些请求生成多个应答。

应用程序标识 (ID) (application identifier, ID) : 在分布式环境中发送的 RPC 中用来识别应用程序的唯一标识符。

应用程序程序员 (application programmer) : 使用 API 编写应用程序的程序员。

应用程序程序 (application program) : 一个用来执行应用程序或部分应用程序的程序。

应用程序窗口 (application window) : 一显示与指定应用程序相关联的图形的矩形区域。应用程序窗口可以打开、关闭、与其它类型窗口组合、移动、堆叠或者通过用户和窗口管理器交互作用来操纵。

应用程序服务器 (application serve) : 在 CDE 中, 提供对软件应用程序访问的主机计算机。

应用程序管理器 (Application Manager) : 在 CDE 中, 一个包含代表可用的系统操作的对象的窗口。

应用程序接口 (API) (application program interface, API) : (1) 一组运行时例程或系统调用, 允许应用程序使用由操作系统或另一个许可程序提供的某个服务。(2) 正式定义的编程语言接口, 介于系统控制程序/许可程序和程序用户之间。

应用程序开发环境 (application development environment) : 用于构造事务性应用程序的监视器函数。

应用程序开发者工具 (Application Developer's Kit) : 请参阅 *ADK*。

应用程序事务处理程序 (application transaction program) : (1) 一个执行应用程序或部分应用程序的程序。(2) 一个与网络中工作站连接和通信的程序, 是用户能够执行面向应用程序的活动。另见 *事务处理程序 (transaction program)* 和 *服务事务处理程序 (service transaction program)*。

应用程序图标 (application icon) : 请参阅 *操作图标 (action icon)*。

应用程序组 (application group) : CDE 中, 一个存放特定软件应用程序或一组软件应用程序的应用程序管理器文件夹。

应用程序 shell (application shell) : 一顶级 shell 的子类, 该 shell 主要用于应用程序的顶层窗口。

应用程序 (application) : (1) 应用于某一特定商业领域的一个或一组程序, 如库存控制或应收帐应用程序。(2) 由 / 为最终用户编写的软件, 执行服务或完成与工作相关的任务。(3) 在 AIXwindows 和 “增强的 X-Windows” 中, 环境以传统的客户机 / 服务器关系建模, 其中兼容的应用程序看作是 “增强的 X-Windows” 服务器的客户机。

应用列表文件 (apply list file) : 一个包含安装或更新过程中要恢复的每个文件的项的文件。

应用 (Apply) : 在 CDE 中, 一个使对话框中的选择 (或选择组) 生效的选项。

应用 (apply) : (1) 在日志记录中, 把记录的后映像 (after-images) 放入物理文件成员中。后映像日志中以项记录。(2) 安装或应用了服务更新之后, 它进入已应用状态并成为软件的当前活动版本。当更新处于已应用状态时, 更新的前一版本被存储在一个特殊存储目录中。这允许必要时恢复到前一版本而无需重新安装。已经应用于系统的软件可以是已提交或已拒绝。命令 **installp -s** 可用来获得状态为已提交或已拒绝的已应用产品和更新的列表。另见 *提交 (commit)* 和 *拒绝 (reject)*。

硬返回 (hard return) : 在 Interleaf 文本文件中, 通过按 Shift-Enter 键创建的, 使文本换到下一行的控制字符。

硬件 (hardware) : 执行计算活动和计算机指挥的活动的物理设备。计算机系统的物理组件。请与 *软件 (software)* 对照。

硬拷贝 (hard copy) : 真实可读形式的机器输出的打印副本, 例如打印的报告、清单、文档和摘要。

硬空格 (hard space) : Interleaf 文件中的非中断空格字符。

硬链接 (hard link) : 允许 **ln** 命令向一个文件指定多个名称的机制。新名称和所链接的文件都必须位于同一文件系统中。

硬盘驱动器 (fixed-disk drive) : 用于在硬盘上读与写的设备。

硬盘驱动器 (hard-disk drive) : 请参阅 *固定磁盘驱动器 (fixed-disk drive)*。

硬盘 (fixed disk) : (1) 平面的、圆形的、不可移动的盘, 带有磁性表面层, 数据可由磁性记录存储在该层上。在硬盘驱动器中使用的硬磁盘。(2) 该硬磁盘术语也宽松地用于工业上的板与盒, 它们包含了模拟硬盘驱动器操作的微芯片或磁泡存储器。

硬盘 (hard disk) : 请参阅 *固定磁盘 (fixed disk)*。

映射对话 (mapped conversation) : 在应用程序和高级程序 - 程序通信 (APPC) 会话之间的临时连接, 在该会话, 系统提供了所有的协议信息。它允许两个程序以由传输程序指定的任意格式交换任意长度的数据记录。只有 LU6.2 会话允许映射对话, 它主要用于应用事务程序。对照 *基本对话 (basic conversation)*。

映射文件 (mapped file) : (1) 能够通过直接内存操作访问的文件, 而不是每次都需要从磁盘上读取来访问。(2) 一个访问起来好像在内存中的固定磁盘上的文件。

映射 (map) : 请参阅 *映射 (mapping)*。

映射 (mapping) : (1) 在 “增强的 X-Windows” 中, 一个执行了映射调用的窗口。如果没有模糊或关闭窗口, 映射使得一窗口可见。(2) 一个列表, 通常在一个概要文件中, 它在两组的各项之间建立对应关系。例如, 当某个字符被按下时, 键盘映射能够建立什么字符被显示出来。另见 *键盘映射 (keyboard mapping)* 和 *概要文件 (profile)*。

映象结构 (image structure) : **XmlInstallImage** 函数中的现有映象。

永久链路 (permanent link) : 多路复用器下没有与其关联的打开控制流而能存在的连接。

永久性存储器 (permanent storage) : 一种其内容不能修改的存储设备。

永久性错误 (permanent error) : 不能通过重试操作消除的错误。

永久性 (permanence) : 事务处理系统的基本属性。此术语意思是, 一旦事务提交, 该事务对数据所做的修改必须是永久的。请求由前一事务修改的数据的后继事务必须总是看到新数据。这些更改必须是可保存的, 即使系统发生故障的情况下。

永久虚拟电路 (permanent virtual circuit, PVC)： 在 X.25 通信中，在每个 DTE 上具有永久指定给它的逻辑通道的一种虚拟线路。不需要呼叫建立协议。对照交换式虚拟电路 (switched virtual circuit)。另见虚拟电路 (virtual circuit)。

用户编号 (user number, UID)： 唯一标识系统的用户的编号。它是与用户标识相关的内部编号。另见用户标识 (user identification, user ID)。

用户标识 (user ID)： 请参阅用户标识 (user identification)。

用户标识 (user identification, user ID)： (1) 一到八个字母数字字符，以字母、#、\$ 或 > 字符开头，它标识了用户。此字符串限制了操作员可以使用的功能和信息。通常，在将用户登录名作为值的命令中，用户标识可以被取代。另见用户编号 (user number)。 (2) 指定用户标识的参数，应用程序或事务处理程序在此用户标识下运行。对照用户名 (user name)。

用户地址列表 (user address list)： 单个用户可以将其与 **xtalk** 命令一起使用来进行呼出 X.25 呼叫的地址列表。另见地址列表 (address list) 和系统地址列表 (system address list)。

用户定义的变量 (user-defined variable)： 一种 shell 变量，用户将字符串作为一个值指定给它。

用户方式 (user mode)： 进程在用户程序中而非内核中执行的方式。对照内核方式 (kernel mode)。

用户概要文件 (user profile)： 处于用户主目录中的名为 **.profile** 的文件，它包含设置初始用户定义的登录会话的特征和缺省值的 shell 命令。

用户结构 (user structure)： 内核方式中，包含当运行进程时必须访问的信息的数据区域。每个活动进程分配一个用户结构。另见 每进程数据区域 (per-process data area) 和用户块 (user block)。

用户界面 (user interface)： 用户用于与系统、程序或设备通信的硬件和 / 或软件。例如键盘、鼠标、命令语言或窗口子系统。

用户空间 (user space)： 用户方式中进程看到的地址空间。另见用户结构 (user structure)。

用户块 (user block)： 由内核维护的数据结构，它包含有关用户进程的系统信息，比如它的真实和有效用户标识、打开文件描述符的列表以及信号处理设置。**user** 结构 (在 **/usr/include/sys/user.h** 的头文件中定义) 指定了保存在用户块中精确信息。另见 每进程数据区域 (per-process data area)。

用户名 (user name)： 唯一标识系统的用户的一串字符。对照用户标识 (user identification)。

用户区域 (user area)： 主存储器和磁盘的可用于用户的部分。

用户时间 (user time)： 程序在 CPU 中运行时所花的时间量。它不包含与提供给此程序的操作系统服务相关的时间，程序的 I/O 时间或其它进程先抢占程序的 CPU 使用的时间。

用户数据报协议 (User Datagram Protocol, UDP)： 直接构建在“网际协议”层上的一种包级别协议。UPD 使用主机间的应用程序至应用程序的程序。

用户数据段 (user data segment)： 在内核方式中，包含用户数据的虚拟内存段，它由初始化的数据变量构成。

用户文件 (user file)： 指定能 (或不能) 使用许可软件产品的用户的文本文件。

用户帐户 (user account)： 请参阅帐户 (account)。

用于 (user)： (1) 与帐户相关的名称。 (2) 要求计算机系统服务的任何人员。

优化单元 (optimized unit)： Ada 语言的编译单元已经被一个或多个编译器优化阶段处理过。当有些源代码级别的信息检查这个单元时对调试器是不可用的。在相同的上下文中可以使用优化单元作为相应的非优化单元。优化和非优化单元可以混合在一个 Ada 程序里内部。

优化 (optimization)： 获取改善的性能或者减小应用程序代码大小的过程。优化可以用编译器，预处理器或者通过手调源代码执行。

优化 (optimize)： 在处理过程中提高程序的速度或减少存储器的使用。

优先级号 (priority number)： 建立打印机请求的相对优先级的号。

优先级值 (priority value)： 调度程序为每个进程维护的一个数，用于表明进程优先级。进程的优先级值越小，它的优先级就越高。

优先级 (priority)： (1) 赋给一个任务的级别，用它确定该任务在接收系统资源方面 (特别是 CPU) 的优先权。 (2) 在资源分配竞争中，一个作业相对其它作业的重要性。进程的重要性或紧急性。

优先权 (precedence)： (1) 用于分组的优先级系统，随其操作数不同运算符类型也不同。 (2) 在编程语言中定义一个表达式中运算符的使用顺序的一个顺序关系。

优先 CUG (preferential CUG): 在 X.25 通信中, 缺省封闭用户组。

邮递程序 (mailer): 实际在传送邮件的程序。

邮件程序控制 (Mailer control): 在 CDE 中前面板用于启动邮件软件应用程序的控件。在控制装入文件中删除文件到“邮件撰写”窗口。

邮件程序容器 (mailer container): 在 CDE 中包含所有邮件消息的电子邮箱和存贮系统。消息一旦放入容器就可对它进行显示、修改、删除、打印、包含、转发及回复。

邮件程序 (Mailer): 在 CDE 中, 可以发送、接收和组织电子邮件报文的应用程序。

邮件 (mail): 工作站间通过网络进行消息格式的通信。与电子邮件 (*electronic mail*) 同义。

邮筒 (mail drop): 消息最先被收到的文件存储位置。

邮箱 (mail box): 网络上发往用户的报文的存储位置。

有限子集 (limited subset): 一个大的数据集的一小部分或程序的简化的版本。

有效根目录 (effective root directory): 在查找文件时的系统开始点。路径名以 / (斜杠) 开始。**chroot** 子例程使一个由路径参数命名的目录成为有效根目录。

有效率 (effective rate): 设备在实际情况下, 在处理典型工作负荷时进行操作的平均支持速度。

有效用户标识 (effective user ID): (1) 与最后认证的用户有关的用户标识或者最后 **setuid** 程序。与实际的或者已保存的用户标识相同。(2) 当前用户标识, 但不是必要的用户的登录标识。例如, 用户在登录标识下登陆可能变成另外一个登录标识。用户更改的用户标识变成有效的, 直到用户转变成最初的登录标识。所有任意的访问决定是基于有效的用户标识的。

友元函数 (friend function): 一种 C++ 函数, 授权为可访问一个类的私有和受保护部分。它在该类的声明中进行指定, 并使用关键字 **friend** 作为前缀。

友元类: 一种 C++ 类, 其中所有成员函数都授权为可访问另一个类的私有和受保护成员。它在另一个类的声明中进行命名, 并使用关键字 **friend** 作为该类的前缀。例如, `class me { 在一行上, 接着在下一行上是 friend class you; 然后是 // ..., 最后结束于 };`, 这使类 `you` 中的所有函数成为类 `me` 的友元。

右边距 (right margin): 最右文本字符和显示器或纸张的右边缘之间的区域。

右手规则 (right-hand rule): 如果右手回绕旋转轴, 则手指以同一方向卷曲作为正旋转, 拇指点也以同一方向作为旋转轴。右手旋转是反时针方向。

语法错误 (syntax error): 编译时由不正确语法引起的错误。另见语义错误 (*semantic error*)。

语法分析 (parse): (1) 在分时系统中, 用来分析用命令输入的操作数并通过这些信息为命令处理器创建参数表。(2) 在命令行解释器可以转换操作系统名称到可执行的机器代码格式前, 命令必须用解释器分析成简单的代码元素、或“分析”。

语法规则 (grammar rule): 解析程序中的结构规则。另见解析器 (*parser*)。

语法模板 (syntax template): 包含星号的词法构造, 在此构造中可以通过用文本代替星号来派生多个属性语法。

语法图表 (syntax diagram): 显示如何在命令行输入命令的命令的图表。

语法 (syntax): (1) 构造命令、语句或程序的语法规则。(2) XOM 中:

OM 语法不同的分类中任何一个, 其中对象管理规范静态地将基于它们的格式的值分组。这些种类是值的 OM 类型的额外情况。

属性值基于其形式而确定的范畴。

语句标号 (statement label): FORTRAN 中包含一至五位十进制数字的号码, 用于标识语句。语句标号通常用于传送控制、定义 DO 循环的范围或指 FORMAT 语句。另见标号 (*label*); 标签。

语句函数定义 (statement function definition): FORTRAN 中定义语句函数的语句。它的形式是语句函数后跟 = (等号), 再跟算术、逻辑或字符表达式。

语句函数 (statement function): FORTRAN 中其后跟着哑变量列表的名称, 与一个算术、逻辑或字符表达式相等。它可以在整个程序中代替表达式。另见宏 (*macro*)。

语句号 (statement number): 请参阅语句标号 (*statement label*)。

语句 (statement): (1) 程序或过程中的指令。(2) 编程语言中, 代表一系列操作或一组声明中一个步骤的语言构造。另见块语句 (*block statement*)。在 Ada 语句中, 语句指定一个或多个在程序执行过程中要执行的操作。

语言环境 (locale): 用户环境的一个子集, 它为一个特定的文化定义了约定, 例如: 时间格式、数值格式, 和字符分类、转换和整理。

语义错误 (semantic error)：由于常量和标识的不正确定义而在编译时引起的错误。另见 **语法错误 (syntax error)**。

语义学 (semantic)：字符之间或字符组和它们的含义之间的关系，独立于它们的解释和使用方式。

预安装 (preinstalled)：制造商安装的软件，并已准备好使用。

预处理器语句 (preprocessor statement)：在 C 语言中，以 # (磅符号) 开始的语句，并包含预处理器解释的说明。

预处理器 (preprocessor)：(1) 影响预备性计算或组织工作的功能单元。(2) 仿真技术中的一种程序，它将数据从被仿真系统的格式转换成仿真器能接受的格式。(3) 一种对源程序进行检查的程序，它检查源程序中的预处理器语句，并运行，从而对源程序进行必要的改动。(4) 一种程序，它在编译器处理源程序前修改，可能还会优化源程序。

预定义连接目标类 (Predefined Connection Object Class)：指定对设备可进行的连接种类和位置。

预定义设备对象类 (Predefined Devices Object Class)：根据类、子类和类型确定，代表每个设备类型。预定义设备对象类包含关于设备的基本信息，如设备方法名称和如何访问包含在其余对象类中的信息。

预定义数据库 (predefined database)：包含系统支持的所有可能设备的配置数据。另见 **设备配置数据库 (Device Configuration Database)** 和 **定制数据库 (Customized Database)**。

预定义约定 (predefined convention)：在 FORTRAN 中，数据项的隐式类型和长度规范基于其名称的初始字符（当没有给定显式规范时）。初始字符 I 到 N 暗示类型整数长度 4；初始字符 A 到 H, O 到 Z, \$ 和 _ 暗示类型实长度 4。

预订 (subscribe)：X.25 通信中，租借一条 X.25 线路，并指定所需设施。

预防性维护软件包 (Preventive Maintenance Package, PMP)：系统的维护包更新。一个 PMP 包含基本操作系统 (BOS) 的更新和安装在系统上的每个可选软件产品的更新。

预连接文件 (preconnected file)：安装时定义的单元或文件。例如，标准输入和标准输出是预连接文件。

域名 (domain name)：名称层次中在整个因特网上使用的层。

预写式记录日志 (write-ahead logging)：一种日志记录机制，其中所有与事务相关的日志记录都在事务实际提交之前写进日志。这保证了那些日志记录将存在于日志中，所以当事务提交的那个时候系统出故障时，它们可用来将可恢复数据恢复到正确的状态。

阈值 (threshold)：(1) 具有以下属性的逻辑运算符，如果 P 是一个语句， Q 是一个语句， R 是个语句等等，那么 P 、 Q 、 R 等等的阈值为真只要至少 N 个语句为真，阈值为假只要少于 N 的语句为真。 N 是调用阈值条件的非负整数。(2) 计算机图形中的一种级别，其级别以上的所有灰度图像可以显示为白色，而级别以下的所有灰度图像可以显示为黑色。

域 (domain)：(1) 网络中数据处理资源处于公共接受通用控制的部分。(2) 在数据库中，指一个属性或数据元素的所有可能值。(3) 在 TCP/IP 中，在分层网络中使用的命名系统。域名系统使用 DOMAIN 协议和 **named** 守护程序。在域系统中，一组主机在域和子域树状结构的层次中分别进行管理。

域 (realm)：一种单元，专门从安全性角度考虑；该术语在 Kerberos 规范中使用。

源程序 (source program)：以源语言表达的计算机程序。

源代码控制系统 (Source Code Control System, SCCS)：为正在开发的程序的源文件维护版本控制的程序。它存储对文件的更改而不是存储更改的文件，这就允许同个文件的多个版本存在于系统中。另见 **修订控制系统 (Revision Control System)**。

源代码 (source code)：到编译程序或汇编程序的输入，以源语言编写。对照 **目标代码 (object code)**。

元类 (metaclass)：在 AIXwindows 和“增强的 X-Windows”，不能实例化小窗口和小配件，但是可以将一系列单独的可继承的资源发送到类下的层次结构的子类中的一个对象类。每一种小窗口子类的实例具有该小窗口通用功能，而且可以将这些功能导出到该类的子小窗口。包含在该类种的是 **Core**、**Composite**、**Constraint**、**Primitive**、**Button**、**Manager**、**MenuMgr** 和 **MenuPane**。

源模块 (source module)：请参阅 **源程序 (source program)**。

原始磁带设备 (raw tape device)：请参阅 **原始设备 (raw device)**。

原始设备 (raw device)：一种设备，它将数据 I/O 作为一个连续流对待，而不考虑数据逻辑结构。例如，I/O 用于固定磁盘，流式磁带在与字符没有关系的字节单元中使用。

原始 I/O (raw I/O) : 对不利用核心缓冲区的块设备的面向字符的访问。

元数据 (metadata) : 和文件系统相关的结构数据, 如目录组织, inode 表和链接。元数据并不是由用户提供的数据; 它是关于用户数据结构的信息。

元素 (element) : (1) 在表或数组中最小的数据单位。(2) 在一个集合中, 一个对象、实体或拥有属性的概念中, 该属性定义了一个集合。与成员 (*member*) 同义。(3) 在 SNA 中, 分区中特定的资源, 它以元素地址来识别。(4) 数组、子区域、枚举或集合的组成部分。(5) 在 AIXwindows 中, 一个对象或类似的数据结构, 它有定义的类的性质。

源文档 (source documents) : 与原始软件一起, 由原始开发公司同时生产的语言信息。

源文件 (source file) : 包含源语句 (比如高级语言程序和数据描述规范) 的文件。包含输入数据或命令的文件。

原型文件 (prototype file) : 包含令牌的一个新文件系统中的一个文件。这些包含引导程序的名称、创建的文件系统的大小和根文件的规范。

原因代码 (cause code) : X.25 通信中, 一个包含在明文和复位指示信息包中的 1 字节代码, 指明发送源和发送原因。与明文原因 (*clear cause*) 同义。另见 **诊断代码 (diagnostic code)**。

原语: 在“增强的 X-Windows”中的一个原语类, 它为由管理器类管理的低层小窗口提供资源和功能。原语类小窗口不能有通常的子小窗口, 但可以有弹出子项小窗口。

源语句 (source statement) : 用编程语言写的语句。

原子操作 (atomic operation) : 一个在设置掩码和等待信号的操作间不会产生信号的操作。

原子 (atom) : (1) 与字符串名称相对应的唯一标识。原子用于标识属性、类型和选择。(2) 一个代表字符串值的 32 位数字。另见 **X 原子 (X Atom)**。

源 (source) : (1) 系统、系统中的程序或向目标提出请求的设备。对照 **目标 (target)**。(2) 高级程序间通信中启动另一个系统上作业的系统或程序。

远程处理 (teleprocessing) : 通过通信通道, 处理从远程位置接收到的或传输至远程位置的数据。与 **远程访问数据处理 (remote access data processing)** 同义。

远程存取数据处理 (remote access data processing) : 同 **远程处理 (teleprocessing)**。

远程打印 (remote print) : 发出打印作业到一个机器 (客户机) 上, 以在网络上的另一个机器 (服务器) 上打印。

远程登录 (remote login) : 在系统上初始化会话, 它通过通信线路进行访问。

远程调步 (remote pacing) : 由远程系统生成的调步尝试控制本地系统的输出。另见 **调步 (pacing)**。

远程过程调用运行时库 (Remote Procedure Call run-time library) : **rpc_\$** 系统调用集, 由 NCS 提供以实现远程过程调用机制。

远程过程调用 (remote procedure call, RPC) : 不同主机或同一主机上的进程间的远程操作的调用。

远程连接 (remote connection) : 本地系统和远程系统或设备间的通信链路。

远程事务处理程序名 (remote transaction program name, RTPN) : 对话的其它 (远程) 端的事务程序的名称。

远程通信 (telecommunication) : 在远程通信线路上的两个或更多位置上的计算机系统之间, 或计算机系统和远程设备之间控制信号与信息的传输。

远程统计信息接口 (Remote Statistics Interface, RSi) : 在 Performance Toolbox 中的管理器 API, 它允许一个应用程序通过网络接口从远程节点 (或本地主机) 访问统计信息。

远程系统 (remote system) : 通过某个通信线路连接到您系统的系统。

远程用户信息服务命令 (FINGER) : Name/Finger 协议。一个应用层的网际协议, 它提供在 **finger** 命令和 **fingered** 守护程序之间的接口。

远程主机 (remote host) : 除某特定操作员正在其上工作的主机外, 网络上的任一主机。同 **外部主机 (foreign host)**。

远程作业输入 (remote job entry, RJE) : 通过输入单元提交作业, 该输入单元通过数据链路对计算机有访问权。

远程 (remote) : 指一个通过通信线路访问的系统或设备。对照 **本地 (local)**。

约会编辑器 (Appointment Editor) : 在 CDE 的日历中, 一个用于调度、更改或删除约会的窗口。

约会列表 (Appointment List) : 在 CDE 中, 由日历显示的按年代顺序排列的约会列表。

约束小窗口 (constraint widget) : 在“增强的 X-Windows”中, 小窗口是组合小窗口的子类。它基于与每个子相关联的约束管理自身的几何外形。

约束 (constraint) : 在 Ada 语言中, 约束确定某类型值的一个子集。在该子集中的值满足约束。

越界 (bounds violation) : 试图使用引用到数组外存储器的索引或指针来访问数组。

运行时环境 (run-time environment) : 子例程和 shell 变量的集合, 提供常用函数和系统组件的信息。

运行时监控器 (run-time monitor) : 监视器的一部分, 它管理事务客户机和从用户和服务请求得到输入的服务器应用程序。

运行时间算法选择 (run-time algorithm selection) : 一种优化技术, 在其中某个操作的参数被求值以确定哪几个等价算法将最有效地执行该操作。

运行时检查 (run-time checks) : Ada 程序运行时发生的错误检查。如果检测到一个错误, 则程序引发一个异常。

运行文件 (run file) : 链接编辑器的输出。适合被装入主存储器和运行的格式中的一种程序文件。另见模块 (module)。

运行状态 (running state) : 当用户可登录和使用机器时该机器的条件。

运行 (run) : (1) 一个或多个作业或程序的性能。(2) 使某个程序、实用程序或其它机器功能被执行。

运算符函数 (operator function) : 过载 C++ 运算符或者是类的成员或者使用至少一个是类类型的参数或者是类类型的引用。

运算符 (operator) ; 操作员 : (1) 符号 (比如 +、- 或 *) 代表运算 (在这个案例中, 加法、减法、乘法)。(2) 操作设备的人。(3) 在 Ada 语言中, 运算符是有一个或二个操作数的运算。一元运算符写在操作数前; 在二个操作数间写二进制运算符。符号表示法是函数调用的特殊种类。可以把运算符声明为函数。许多运算符是通过类型声明来隐式的声明 (例如, 大多数类型说明隐含类型值的等同性运算符的声明)。

[Z]

载波信号 (carrier signal) : 一种具有固定频率的信号, 经调制后可传输数据信号。

载波 (carrier) : 数据通信中可以调制或记忆携带信息的信号的持续频率。

暂挂状态 (pending state) : 服务器程序的状况, 在此状况中, 服务器程序接收操作 (启动、停止或暂挂) 请求, 但还没有执行该操作。

暂挂状态 (suspended state) : (1) 资源暂时不能接收请求的状态。启动操作请求将资源返回至它暂挂前的状态。(2) 系统不分派任务且不争夺处理器的软件状态。

暂挂 (pending) : 等待, 当在暂挂的操作中时。

噪声 (noise) : (1) 影响信号并可能使信号携带的信息失真的一种干扰。(2) 任何实体, 例如电压、电流或数据的特征随机变量。(3) 已知振幅、分布和和谱密度统计属性的随机信号。

增量备份 (incremental backup) : 自上一次备份创建以来由于非只读访问的原因而复制打开的文件的过程, 该过程满足备份频率标准。请与完全备份 (full backup) 对照。

增强的 X-Windows : 开发各种应用程序环境种类的基本功能集合。工具箱功能管理工具包初始化、窗口小部件、内存、事件、几何、输入焦点、选择、资源、事件的转换、图形上下文、象素映射和错误。另见 AIXwindows 环境 (Environment)。

增强型图形适配器 (enhanced graphics adapter (EGA)) : 一种适配器, 诸如增强型图形适配器, 提供了高分辨率的图形, 允许使用彩色显示器进行文本处理和图形应用程序。

增强型小型设备接口 (ESDI) : 增强型小型设备接口 (Enhanced Small Device Interface)。

摘机 (off-hook) : 关于电话设置的激活。在公共交换机系统上, 自动应答的数据集被认为是执行摘机。对照挂机 (on-hook)。

摘要 (digest) : 被组织成能够提供对每部分数据进行快速访问格式的数据。

占位符 (placeholder) : 只用于标记有意实体位置的对象、组件或文件。

粘滞位 (sticky bit) : 一种访问许可位, 它使可执行程序保留在磁盘的交换区上。只有具有 root 权限的人员可以设置粘滞位。此位用于目录以指明只有文件所有者可以链接或断开链接该目录中的文件。

站 (station) : (1) 能够发送或接收数据的计算机或设备。(2) 系统的使用远程通信设施 (比如一个或多个系统、计算机、工作站、设备以及某个特定位置的相关程序) 在远程通信线路上发送或接收数据的输入或输出点。(3) 设备上执行操作的位置。(4) FDDI 中, 环路中可寻址的逻辑和物理连接, 此环能够传输、接收以及重复信息。(5) SNA 中的链路站。

帐户 (account) : 目录中的日志和其它给予用户访问系统的其它信息。

着色 (paint) : 在计算机图形学中, 用阴影来显示图像的某区域。

兆赫兹 (megahertz, MHz) : 度量频率的单位。一兆赫兹等于 1 000 000 赫兹。

照相排版机 (phototypesetter) : 一种排版机, 它是这样工作的: 将光通过字体字符的胶片矩阵投射到光敏纸或胶片上。

兆字节 (megabyte, MB) : 不精确的, 一百万字节。当考虑到半导体存储器容量, 十分之一的能力; 十进制表示为 1 048 576。当考虑到媒体设备存储器, 一兆字节是其 5/3 的能力 (1 000 000)。

折叠 (collapse) : 使用 CDS 浏览器从显示器 (关闭它) 除去目录内容。要折叠一个打开的目录, 请双击它的图标。双击关闭的目录可扩展。

折线 (polyline) : 在计算机图形学中, 一系列相邻的线。

真彩色 (true color) : “增强的 X-Windows” 中,

1. 直接色彩的一种退化情况, 其中像素值中的子字段直接对相应的 RGB 值进行编码。即, 色彩图已经预先定义了只读 RGB 值。这些值通常近直线不断倾斜。
2. 也代表 TrueColor, 一个值。
3. 指每像素使用 24 位直接 RGB, 其中 8 位 (256 级别) 红色、8 位 (256 级别) 绿色和 8 位 (256 级别) 蓝色, 总共 256*256*256 或 16,777,216 种红、绿、蓝强度的不同组合。可以同时显示的不同颜色像素的数量, 此数字受到帧缓冲区中可显示像素的数量的限制。

诊断代码 (diagnostic code) : 在 X.25 通信中的一个 1 字节代码, 包含在给出发送信息包的原因信息的空包和重设信息包中。另见原因代码 (cause code)。

诊断控制程序 (Diagnostic Control Program) : 用于诊断的顶层控制程序和配置管理器。它将遍历配置数据库, 测试资源和资源之间的依赖性。它分析来自诊断应用程序的结论并生成问题报告。

诊断输出 (diagnostic output) : 除了由标准输出生成的这些消息之外由进程生成的错误或状态消息。错误输出 (error output) 的同义词。

诊断助手 (diagnostic aid) : 用来检测和排除设备或程序故障或错误的工具 (过程、程序或参考手册)。

诊断 (diagnostic) : 与程序错误和设备故障的检测和排除有关。

振幅 (amplitude) : 电压或电流波形的大小或数量。

正常端口 (normal port) : 允许用户登录的异步端口。当该端口启用时, 它的外出作用不被允许。

正常连接停止 (normal attachment stop) : 请参阅正常停止 (normal stop)。

正常模式 (normal mode) : 请参阅多用户模式 (multiuser mode)。

正常停止 (normal stop) : 两种停止连接的方法之一。(另一种方法是强制停止。) 如果该连接或它们的任何连接处于暂挂状态, SNA Server 拒绝正常停止操作。

正交投影图 (orthographic projection) : 在表示中投影的行是并行的。正交投影图缺乏透视图透视缩小以及它附随的较强的深度感。因为它们绘制简单, 绘图员经常使用正交投影图。另见透视投影图 (perspective projection)。

正文 (body) : (1) 打印页面中包含主要文本的页面部分, 不包括页边距、标题、页脚和脚注。(2) 一本书中前页和后附页之间的部分。(3) Ada 语言中主体定义子程序、包或任务的执行。body stub 是主体的一种格式, 表示该执行定义在独立的编译器单元中。

争用作用域 (contention scope) : 一给定线程与一组线程组必须争用 CPU。如为本地, 该线程与同一进程中的其它线程竞争。如为全局, 该线程与系统中的所有其它线程竞争。

争用 (contention) : (1) 在局域网中, 当协议允许两个或更多个站在通信通道中并发发送而导致冲突风险的情况。(2) 在一个会话中, 当两个程序试图同时开始对话时发生的情况。

正则表达式 (regular expression) : (1) 一组字符、元字符和定义搜索模式中字符串或字符串组的运算符。(2) 一种字符串, 它包含通配符和定义一组一个或多个可能字符串的操作。对照文字字符串 (literal string)。另见整理元素 (collating element) 和次模式 (subpattern)。

整理表 (collation table) : 提供函数使用的有序字符集和字符同等类别。

整理序列 (collating sequence) : 字符在计算机内经过安排的序列, 用于排序、组合或比较。

整理元素 (collating element) : 与正则表达式 (regular expression) 中的某个序列匹配的一个或多个字符。

整理 (collation) : 基于字母顺序和同等类别进行字符和字符串排序的过程。

整数表达式 (integer expression) : 一个只有一个整数类型值的算术表达式。

整数常数 (integer constant) : (1) 一个不包含十进制小数点的十进制常数。(2) 一个八进制或者十六进制常数。(3) 一个不包含十进制小数点的十进制数字的字符串。

整数对象 (integral object) : 在 C 语言中, 一个字符对象, 一个枚举类型的对象, 或者一个对象具有 **short**、**int**、**long**、**unsigned short**、**unsigned int** 或 **unsigned long**。

整数类型 (integer type) : 一个由整数值组成的算术数据类型。在 Ada 语言中, 一个整数类型是离散类型, 它的值表示特定范围内的所有整数。

整数 (integer) : 正数、负数或者零。

帧窗口 (frame window) : 在 X.25 通信中, 可能是未经确认的未完成的帧数。另见数据包窗口 (*packet window*) 和窗口 (*window*)。

帧缓冲区 (frame buffer) : 用于存储显示在监视器上的图像的一些视频 RAM (VRAM)。帧缓冲区是绘制每个绘图图元的电子画布。它是图形流水线中最后的停止码元之一, 最终图像以数字编码的强度和亮度的形式驻留在其上。它们被转换成每秒 60 次的模拟电压信号并发送到监视器的电子枪。可通过 GL 更改帧缓冲区的大小。通常, 主帧缓冲区可能是垂直 1024 像素 × 水平 1280 像素 × 8 色彩位。重叠平面可能是 1024x1280x2。尽管 z 缓冲区从监视器不是直接可见的, 但将它当作是一个帧缓冲区。(没有一种直接的方法来显示 z 缓冲区的内容, 尽管这一点可以间接实现。) z 缓冲区的大小通常是 1024x1280x24。光标是帧缓冲区的一种极特殊形式; 它可以四处移动。典型光标大小为 32x32x2。请参阅光栅 (*raster*)。

帧级别 (frame level) : 在 X.25 通信中, 物理级别和信息包级别之间的那个级别, 它按照高级数据链路控制过程 (HDLC) 工作。其含义与数据链路级别 (*data-link level*) 和级别 2 (*level 2*) 相同。另见级别 (*level*)、信息包级别 (*packet level*) 和物理级别 (*physical level*)。

帧级接口 (frame-level interface) : 在 X.25 通信中, 与具有本地错误控制的数据包交换相关的, 以数据包方式操作的 DTE/DCE 接口的级别, 其中信息包包含在帧中。另见信息包级别接口 (*packet-level interface*)。

帧脉冲串 (frame burst) : 一组接收介质上的连续帧的发射器。

帧 (frame) : (1) 在高级数据链路控制 (HDLC) 中, 由开始标志和结束标志 (01111110) 序列括起来并包括标志序列在内的连续二进制的序列。(2) 一组连续的数字时间片断, 每一数字时间片断在组内的位置可参考帧同步信号来标识。(3) 在隔行扫描视频监视器中, 一帧由两个域组成, 称为“偶数帧”和“奇数帧”, 每个半帧的持续时间为 1/60 秒。(4) 在块多路复用通道适配器和 ESCON 传输中, 一个 CCW 中的 S/370 和工作站之间传送的数据

块。(5) 在一个环上的两个或多个站之间传送的相应于 FDDI 协议的信息块。一个 FDDI 帧类似于一个信息包。

支持单元 (supporters) : Ada 语言中, 语言允许编译一个单元而需要的所有编译单元。这由单元导入、它们的导入等等构成。

只读存储器 (ROS) 菜单 (Read Only Storage (ROS) menus) : 系统 ROS 在目标上显示的一种菜单。引导目标使用的信息由用户提供给 ROS 菜单。

只读文件 (read-only file) : 指文件系统挂装, 允许数据被读但不能被复制、打印或修改的条件。

只读引用传递 (pass-by-read-only-reference) : 与常量传递 (*pass-by-CONST*) 同义。

指令地址寄存器 (instruction address register, IAR) : 一个包含将运行的下一条指令地址的系统控制寄存器。该 IAR 可以以监控器状态中的一个监控调用方式来访问, 而不是在问题状态下直接寻址。与程序计数器 (*program counter*) 同义。另见位置计数器 (*location counter*)。

指令调度 (instruction scheduling) : 一种编译器优化, 调度指令到 CPU 的不同处理单元, 以产生最大程度的指令重叠。

指令高速缓存 (instruction cache) : 一个为处理器提供程序指令的高速缓存, 从这里获取指令要比从 RAM 中快得多。

指令指针 (instruction pointer, IP) : 请参阅指令地址寄存器 (*instruction address register*)。

指令 (instruction) : 一个指定了由计算机执行的一个操作以及操作数的值和位置的程序语句。这个语句表示了程序员要求处理器要执行的一个特定操作。

指示笔 (stylus) : 用来在图形输入板上选择一个特定位置的设备。

指示器 (indicator) ; 指示符 : (1) 在程序或过程的各部分之间的状态通信内部开关。(2) 一个可以被设置为指定状态的设备, 通常要求一个先前进程的结果或者在一个设备上的指定情形时。该设备通常用来表明状态的存在, 也可能用来确定对选择性的进程的选择。

指数 (exponent) : 表示另一个数 (底数) 将提升的幂次的数。

指向成员的指针 (pointer to member) : 用于访问 C++ 类的非静态成员地址。

指针类型 (pointer type) : 一种数据类型, 它定义包含地址的变量, 有时定义包含关于变量的其它信息的变量。

指针抓取 (pointer grabbing)：在“增强的 X-Windows”中，一个客户机可活动地抓取控制指针，以使按钮和移动事件发送到该客户机，而不是发送到这些事件通常要发送的客户机。另见抓取 (*grab*)、按钮抓取 (*button grabbing*) 和键抓取 (*key grabbing*)。

指针 (pointer)：(1) 保留数据对象地址的一种变量。(2) 唯一目标的一个物理标识或符号标识。(3) 在计算机图形学中，连接光标和在屏幕上跟踪的设备。

值传递 (pass-by-value)：在 Pascal 中，实参的值的拷贝传递到调用例程的参数传递机制。如果调用例程修改形式参数，则不影响相应的实参。

执行不到的代码 (unreachable code)：在程序执行中执行不到的代码。检测到并除去执行不到的代码是优化的一部分。

执行 (exec)：使用另一个可执行程序覆盖当前进程。另见派生 (*fork*)。

执行 (execute)：CDE 中，文件或文件夹（目录）的访问许可。对于文件，可执行许可允许用户运行程序或 shell 脚本文件。对于文件夹（目录），可执行许可允许用户访问文件夹（目录）的内容。

直接可见性 (direct visibility)：请参阅可见性 (*visibility*)。

直接连接 (direct connection)：系统、工作站或其它 I/O 设备的附属物，通过所选的通信接口和有限长度的电缆连接。不需要调制解调器。

直接色彩 (direct color)：(1) 在“增强的 X-Windows”中的一类色彩图，其像素值分解为三个独立的子字段以进行检索。一个子字段对数组进行索引并生成红色亮度值，第二个子字段对另一个数组进行索引并生成蓝色亮度值，第三个子字段则生成绿色亮度值。RGB 值可以动态更改。它与伪色彩图是互相排斥的。(2) 还有 DirectColor 也是值。

直接寻址 (direct addressing)：(1) 使用表达式作为操作数项来表示指令地址的寻址方法。(2) 一种寻址方法，其指令的地址部分包含直接地址。

直接映射的高速缓存 (direct-mapped cache)：一种高速缓存，它的一行与识别要查询的行的虚拟地址字段的每个可能的值相对应。

直接总线连接 (direct-bus attached)：通过直接连接到系统主板的固定磁盘驱动器进行使用，而不通过 SCSI 适配器卡使用。

值 (value)：(1) 与参数或名称相关的字符集或数量集。(2) 编程中变量或存储位置的内容。(3) 属性的特定具体

值，比如属性颜色的蓝色。(4) 指定给常量、变量、参数或符号的数量。(5) XOM 中，可作为对象的特征或属性查看的任何综合信息项。另见属性值 (*attribute value*)。

制表窗口 (tabulating window)：Performance Toolbox 中窗口的一种特殊形式，当接收到数据时，它将工具值制成表格，并且还能计算一行上每个值的加权平均值。

置换 (displacement)：(1) 可以添加到基寄存器的内容以计算有效地址的正数或负数值。(2) 从记录、块或段的开始到字段开始的距离。

制作镜像 (mirroring)：一个图元镜像映象的创建。

终端仿真器 (terminal emulator)：允许设备（比如微型计算机或个人计算机）从计算机系统输入或接收数据就好像它是某个特定类型的连接的终端。

终端屏幕 (terminal screen)：显示屏 (*display screen*)。

终端应用程序 (terminal application)：在工作站（终端）运行的应用程序。

终端映射 (terminal mapping)：在标准字符集和特定于终端的字符集之间转换。

终端 (Terminal)：CDE 的前面板中，“个人应用程序” (Personal Applications) 子面板的一个项，用于打开终端窗口。

终端 (terminal)：(1) 一种设备，通常装有键盘和显示设备，能够在通信线路上发送和接收信息。另见工作站 (*workstation*)。(2) 系统或通信网络中，可以于其上输入或除去数据的点。(3) curses 和扩展 curses 中，表示当前工作站显示屏如何显示的特殊屏幕。终端屏幕由名为 **curscr** 的窗口标识，用户不能直接访问此窗口。相反，用户对 **stdscr** 窗口（或用户定义的屏幕）进行修改，然后 **refresh**（或 **wrefresh**）窗口来更新终端。

中断包 (interrupt packet)：在 X.25 通信中，允许快速包覆盖正常数据包（由序列传送的包）。另见包 (*packet*)；信息包；分组。

中断确认包 (interrupt-confirmation packet)：在 X.25 通信中，一个用于确定中断包的接收方的信息包。另见包 (*packet*)；信息包；分组。

中断条件 (break condition)：TTY 子系统中，数据全为零的字符帧错误。

中断信号 (break signal)：远程连接上发送的信号，用来中断远程系统上的当前活动。

中断语句 (break statement)：C 语言控制语句，包含关键字 **break** 和一个分号。

中断值 (break value)： 在分配数据段空间过程中，超出当前数据段终止处的第一个位置的地址。

中断 (interrupt)： (1) 在数据通讯中，在接收站采取措施使发送站结束传输。(2) 临时停止一个进程。(3) 当错误发生或需要辅助以完成 I/O 时，由 I/O 设备发送到处理器的信号。一个中断经常暂挂当前正在运行的程序。对照异常 (exception) 和信号 (signal)。

中间节点 (intermediate nodes)： 在小窗口树上带一个或更多个子类的小窗口。另见 widget 树 (tree)。

中间数据类型 (intermediate data type)： 按照其它、直接的接口所定义数据类型的任一基本数据类型。

中间语言调用 (interlanguage call)： 在一个用特定语言编写的程序中，对采用另一种语言编写的过程的任一引用。

中央处理器 (central processing unit, CPU)： 计算机部件，包含控制指令解释和运行的电路。

终止符 (terminator)： 程序产品中执行必要的操作以结束作业或程序的部分。

众所周知的端口 (well-known port)： 支持相同协议的主机（无论这些主机是否处于同一个网络）使用的约定的端口指定。与触点端口 (contact port) 同义。

众所周知的主机名称 (well-known host name)： 特定网络上与“网际协议”地址相关的惯用名称（例如 nameserver 和 timeserver 服务器）。

周期时间 (cycle time)： (1) 一个处理器周期的过程时间。从一类处理器到另一类，周期时间变化很大。(2) 存储设备的两个连续读 / 写周期起始之间的最小时间间隔。

逐段线性曲线 (piecewise linear curve)： 用于非统一有理 B 样条 (NURBS) 表面，在参数空间中的坐标对的列表。这些点与直线连接，以形成路径。

主表达式 (primary expression)： 标识、带括号表达式、函数调用、数组元素说明、结构成员或联合成员说明。

主程序 (main program)： (1) 当程序运行时的接收控制的第一个程序单元。对照子程序 (subprogram)。(2) 执行主函数的程序，为执行更多的特定功能，将控制传送到例程和子例程。

主处理器 (master processor)： 在多处理器系统中，引导时所启动的第一个处理器。

主窗口 (home-window)： 显示缓冲区的内容得到完全显示的窗口。

主存储器 (main storage)： (1) 可寻址的程序存储器，通过它指令和其它的数据可被直接装入到寄存器以便后续的程序运行和处理。与系统存储器 (system memory) 同义。(2) 内部存储器的一部分，其中的指令和其它数据必须被装入以便运行和处理。(3) 程序运行所在处理单元的一部分。

主导航文本 (primary navigation article)： 软件文档的常规、顶级单元。

主服务器 (master server)： 在一个网络安装环境中，有权在该环境下的所有机器上执行命令的服务器。主服务器被设计为管理网络、客户机和网络安装数据中的资源对象。

主函数 (main function)： 一个函数，具有标识符 main。每个 C 语言程序必须含有一个函数名为 main。此函数是 C 语言程序的主程序。

主缓轮询 (primary slow poll)： 主链路站使用的，用于减少次链路站非生产性的轮询的技术。

主会话 (home session)： 在 CDE 中，指注销时指定特定会话的选项，不同于您当前所在的会话，而被看作是您下一次登录时会自动返回到其中的会话。请与当前会话 (current session) 对照。

主机标识 (host ID)： 用于主机的标识符。主机标识对网络上某一地址系列中的主机进行唯一标识，但不标识该网络。主机标识不一定能足以建立与主机的通信。

主机名称 (host name)： (1) 指定给与网络连接的计算机的名称。使用此术语可能会产生歧义，因为它有可能指计算机的简短名称形式，或者指计算机的全限定名称。(2) 网络中某台机器的因特网地址。也称为主机标识 (host ID)。

主机转储表 (master dump table)： 一个包含由内核组件生成的转储表项的结构。转储程序使用该表来定位应该包含在转储中的数据结构。

主机 (host)： (1) 通信网络中的主计算机或控制计算机。(2) 连接到网络的计算机。

主机 (master)： 在 NIM 环境中的唯一机器，它允许在其他 NIM 客户机上远程执行命令。

主叫地址扩展 (calling address extension)： 请参阅地址扩展 (address extension)。

主叫地址 (calling address)： 请参阅网络用户地址 (network user address)。

主叫约定 (calling conventions)： 用于相互交换数据的例程和子例程的指定方法。

主叫 DLS 用户 (calling DLS user)： 处于连接方式的数据链路服务 (DLS) 用户，初始化数据链路连接的建立。

主叫 NS 用户 (calling NS user)： 一初始化网络连接的网路服务 (NS) 用户。

主叫 (calling)： X.25 通信中，应用于主叫位置或用户的 adjective。

主列顺序 (column-major order)： 一种存储数组元素的方法，使得当访问内存相邻元素时最左边的下标变化最快。

主面板 (Main Panel)： 在 CDE 中，不包括子面板的前面板的一部分。

主目录 (home directory)： (1) 与单个用户关联的目录。(2) 在登录后或发出不带参数的 **cd** 命令之后用户的当前目录。(3) 为事务程序提供完整的主目录路径名的参数。

主企业系统连接管理程序 (Primary Enterprise Systems Connection Manager)： 在多 ESCM 环境中，ESCM 命令的源。

主体标识 (principal identifier)： 用于唯一标识主体的名称。

主题标识符 (subject identifier, SID)： 标识一个用户或一组用户的字符串。每个 SID 包含三个字段，其形式为 person.group.organization。在一个帐户中，每个字段必须有一个特定的值；在 ACL 条目中，一个或多个字段可能为一个通配符。

主题树 (topic tree)： CDE 中位于一般帮助对话框中的可扩展和可收缩主题列表，可以选择它来显示帮助信息。

主文件夹 (home folder)： CDE 中保存个人文件和附加目录的文件夹。缺省情况下，当首次打开“文件管理器”和“终端仿真器”窗口时将它们设置成主文件夹。在讨论命令行活动时，可以称为主目录 (*home directory*)。

主文件 (master file)： (1) 一个永久信息的集合，例如一个客户地址文件。(2) 在给定的作业中用作许可的相对固定的文件，尽管它的内容可能变化。

主文件 (prime file)： Pascal 中的一种文件，它含有 Pascal 编译器内部表格的预编译声明。主文件用于在编译开始前对编译器内部表进行初始化。

主选择文本 (primary selected text)： 一种选定的作为主目标或目的地的文本组，特别是在可传递给函数的文本字段中选定的文本。还可以是函数或语句中指定的文本的第一个块。另见主选择 (*primary selection*)。

主选择 (primary selection)： 在 AIXwindows 中的小窗口中选定的一种文本。主选择有一个由 **XmTextGetSelection** 函数检索的值。另见主选择文本 (*primary selected text*)。

主要设备号 (major device number)： 每个设备或设备类型的一个系统标识号。主要设备、次要设备和通道号唯一地标识一个硬件设备。另见次要设备号 (*minor device number*)。

主语言 (primary language)： 您要系统使用的，用于屏幕信息的主语言环境。

主站 (primary station)： (1) 在点到点通道上，首先获得通道控制的站。在多点通道上，控制通信的站。(2) 在高级数据链路控制 (HLDC) 中，数据站的一部分，它支持数据链路的主控制功能，生成传输命令并解释接收的响应。(3) 在 SNA 中，一个 SDLC 数据链路上负责数据链路控制的站。在一个数据链路上只能有一个主站。数据链路上的所有流量都是在主站和次级站之间进行的。

主组 (primary group)： 在并发组中，指定给您创建的文件组。

主 (primary)： 数据的一种不可缩小的单位。例如，一个常量、变量或数组元素。

注册表数据库 (Registry database)： 关于人员、组、组织和帐户信息的数据库。

注册 (registration)： 在网络安装数据库中创建某个对象，它唯一地向主控服务器标识网络安装环境中的客户机、网络或资源。

注册 (sign-on)： 在一个显示站上开始一个会话。

助记符准备 (mnemonic overstrike)： 通过将一个字放在另一个字符上所创建的非标准字符，以代表一个非打印字符。

助记符 (mnemonic)： (1) 为了帮助用户记住符号的意义所选择的符号。(2) 包含一个机器指令的首字母缩写词或缩略词的汇编指令字段。(3) 在 CDE 中，命令中的单个带下划线的字符。助记符表明，您可以通过输入那个字母选择命令。助记符更经常用在菜单命令中，然而，其它按钮也可以有助记符。

柱面坐标系统 (cylindrical coordinate system)： 一个阵列，其每一行的左边在功能上与同一行的右边相邻。

柱面 (cylinder)： 无需移动磁盘驱动器或磁盘驱动器读/写机构就可读或写的所有硬盘或软盘磁道。

注释 (annotation)： 参考许可证注释 (*license annotation*)。

注销 (log off) : 在一个显示站上结束与一个计算机系统的会话。

注销 (log out) : 请参阅退出 (log off)。

注销 (sign-off) : 在显示站结束会话。

抓取句柄 (grab handle) 或句柄 (handle) : 指 CDE 中显示在选定图形元素角落和中点的小方块。

抓取 (grab) : (1) 选择键盘键、键盘、指针按键、指针和客户机专用服务器的操作。一般来说, 这些工具并不是打算供普通应用程序使用, 而是打算用于各种输入和窗口管理器以实现多种样式的用户界面。另见活动抓取 (active grab)、被动抓取 (passive grab)、按钮抓取 (button grabbing)、指针抓取 (pointer grabbing) 和键抓取 (key grabbing)。(2) 一个过程, 窗口通过它执行对自身或其任何后代而发生的键或按钮事件。这消除了对所有窗口设置转换的必要性。

转储表项 (dump table entry) : 一个在主转储表中的记录, 它可以标识组件转储表的位置。这些需要由转储程序收集特殊数据的所有内核组件, 都需要产生一个转储表项。

转储数据 (dump data) : 由内核转储程序收集的数据。它是从由内核组件使用的存储位置获得的。

转储 (dump) : (1) 把全部或部分存储器里的内容拷贝到另一个数据介质或输出设备中。(2) 已经转储的数据。

转化后援缓冲区 (translation lookaside buffer, TLB) : CPU 中的一个表, 它包含最近引用内存页的虚拟地址和真实地址之间的交叉引用。

转换表列表 (translation table list) : 请参阅转换表 (translation table)。

转换表 (translation table) : (1) 指定事件映射或过程名称的事件顺序的表。(2) 包含将事件转换为过程调用列表的字符串。

转换代码 (conversion code) : 打印函数调用中的值类型规范, 如同该值即将被打印出来一样 (例如, 以八进制格式)。

转换规范 (conversion specification) : 打印函数调用中, 一个系统如何在输出流中放入零值或更多格式参数的规范。每个转换规范包含一个 % (百分号) 字符, 后跟转换修改符和转换代码。

转换函数 (conversion function) : C++ 成员函数, 指定从它的类类型到另一个类型的转换。

转换器高速缓存 (converter cache) : 一个高速缓冲存储区, 包含频繁地访问与客户机应用程序相关联的信息。转换器端高速缓存的首要目的是减少对关键信息的存取时间。

转换器 (converter) : 一个不更改基本信息就可将数据从一种格式转换为另一种格式的设备。

转换 (conversion) : (1) 编程语言中代表相同数据项, 但是属于不同数据类型的值之间的转换。(2) 值类型更改。例如, 当添加不同数据类型的值时, 编译程序在添加它们之前将两个值都转变为相同的形式。另见事务处理程序 (transaction program)。

转换 (diversion) : 在文本格式化中用来保存文本以便今后在文档 (比如脚注) 中打印的命令。

转换 (translations) : 调用事件或事件序列的操作过程。

转接延迟 (transit delay) : X.25 通信中, 信息包从一个 DTE 转移到另一个 DTE 所花的时间。

转向 (turnaround) : 将通信线路从传输方式更改至接收方式, 或从接收方式更改至传输方式。

撰写窗口 (Compose window) : 在 CDE 的邮件程序中, 用来新建电子邮件消息的窗口。撰写窗口菜单栏有几个消息撰写选项可用。

撰写 (compose) : 设置类型。

转义函数 (escape function) : 形如 “ESC Something” 的代码。一种多字节控制函数。

转义序列 (escape sequence) : (1) 前置一个 \ (反斜杠) 并解释为对操作系统有特定含义的字符。(2) 发送到终端以执行诸如移动光标、从正常转为反转视频和清除屏幕等操作的序列。terminfo 文件定义这些转义序列。与多字节控制 (multibyte control) 同义。(3) 指定一个字符串变量的多字符代码。另见非转义的 (unescaped)。

转义字符 (ESC) : 请参阅转义字符 (escape character)。

转义字符 (escape character, ESC) : (1) shell 编程和 TTY 编程中的 \ (反斜杠) 字符, 它表示下一个字符并不是通常赋予它的特定意义。(2) 一般地说, 是指一个抑制或为紧跟它的一个或多个字符选择一个特定含义的字符。

装订边 (binding edge) : 页边, 打孔自由装订, 装订、分类或钻孔。

装配 (mount) : 使一个文件系统可以访问。

装入模块 (load module) : 请参阅运行文件 (run file)。

状态工具 (state instrument)：在 Performance Toolbox 中，状态工具显示了系统资源的最新统计信息，可选的有加权平均值。然而它不按时显示统计信息，一些状态工具只有当您要将工具更改至记录工具时，才收集此数据。用于绘制这些记录（包含状态栏、圆形分格统计图表和速度计）的某些类型的图。对照记录工具 (*recording instrument*)。

状态信息 (state information)：关于小窗口或配件外观及行为的当前状态的信息。此信息在每个独立的小窗口和配件中记录并在必要时更新。

状态 (state)：(1) 在应用合适的脉冲前电路保持的状态。(2) 单独的可重启的程序部分，`runacct` 命令（主要日常记帐 shell 步骤）可在其中终止其处理。

状态 (status)：(1) 程序或设备的当前情况或状态。例如，打印机状态。(2) 硬件或软件的情况，通常以状态代码表示。(3) 在“增强的 X-Windows”中，许多 **Xlib** 子例程返回一个成功状态。如果子例程不成功，它的值无论何种情况都不会受到干扰。

追加 (append)：(1) 把数据添加到已存在数据末尾的操作。(2) 在字处理中，把一个文件附加到另一个文件的末尾。

桌面 (desktop)：系统中的一组对象的可视化表示，将它们放在一起以便组织工作。

咨询锁 (advisory lock)：锁的一种，进程拥有文件的某个区域以防止任何其它进程锁定该区域或覆盖区域。另见强制锁 (*enforced lock*)。

资源标识 (resource identifier, rid)：在“增强的 X-Windows”中，返回给某应用程序的一个整数，该整数标识一个已为该程序的使用而分配的资源。

资源对象 (resource object)：在网络安装管理环境中使用的代表文件或目录的资源。

资源对 (resource pair)：请参阅资源值 (*resource value*)。

资源管理器 (resource manager)：管理应用程序数据的组件。资源管理器与应用程序服务器进行通信。

资源利用率值 (resource utilization value)：一个进程当前在系统中使用的资源量。另见资源集合的作用域 (*scope of resource collection*)。

资源目标共享 (resource target share)：共享资源应该对工作负载管理类可用。这些共享与其它类共享一起使用，以确定在类之间进行期望的资源分布。

资源缺乏 (resource outage)：没有能力获取诸如链接、缓冲区或控制块存储这样的必需资源。

资源收集作用域 (scope of resource collection)：在“工作负荷管理” (*Workload Management*) 中，指定给某个类的资源收集作用域，它确定资源限制如何应用于该类中的进程。作用域级别可设置为类、进程、用户，或组。

资源数据库 (resource database)：为显示器、主机和应用程序的特定组合而组装的数据库。它可包含几个源的信息。

资源限制值 (resource limitation values)：一组值，工作负载管理尝试为一组资源使用值而维护它。资源限制值独立于 `setrlimit()` 指定的资源限制。

资源值字符串 (resource value string)：一种参数，它包含没有设置特定资源时而使用的缺省值。

资源值 (resource value)：资源确定小窗口或窗口的操作和诸如颜色和行为的属性。该资源与为每个资源而给出特定值的值列表关联。另见属性 (*attribute*)。

资源状态 (resource state)：一种状态，它表明资源是可用的或不可用的。

资源字符串 (resource string)：参数，如程序名的部分，标识某个应用程序的资源。

资源 (resource)：(1) 在 NIM 中，任何文件、目录、文件系统或需要执行 NIM 操作的设备。(2) 在“增强的 X-Windows”中，如窗口、像素图、光标、字体、图形上下文和色图这些项都被称为资源。为命名起见，每个项都有一个与之关联的唯一标识。资源的生命期由在其上创建资源的连接的生命期所定界。(3) 在“增强的 X-Windows”中，某个小窗口内的一条命名数据，它可由客户机、应用程序或用户缺省值来设置。(4) 在 AIXwindows 和“增强的 X-Windows”中，与一个特定的图形对象类相关联的外观或行为的唯一特性。在类层次结构中，资源可以从较高级类自上而下由其每个子类向下传递（继承）。

子菜单 (submenu)：通过 \rightarrow 符号从另一个菜单访问的菜单。只能从主菜单访问的相关菜单。AIXwindows 中子菜单的示例是 `XmCascadeMenu` 小窗口，当在主菜单项上侧向拖动鼠标，它从 `XmPopupMenu` 小窗口的一侧出现。

子程序 (subprogram)：(1) 另一个程序调用的程序，比如 `subshell`。(2) FORTRAN 中，第一条语句为 FUNCTION、SUBROUTINE 或 BLOCK DATA 语句的程序单元。对照主程序 (*main program*)。(3) Ada 语言中，子程序是一个过程或一个函数。过程指定操作的顺序，并由过程调用语句调用。函数指定操作的顺序，还返回称作结果的值，所以函数调用是一个表达式。子程序作为子程序声明编写，指定了它的名称、形式参数和（对于函数）它的结果；以及指定操作顺序的子程序主体。子程序调用

指定将与形式参数相关联的实际参数。子程序是各种程序单元种类之一。另见函数 (*function*)；功能和过程 (*procedure*)。

子串 (substring)： 字符串中连续的子区段。

子单元 (subunit)： 对于 Ada 编程，请参阅主体 (*body*)。

子地址 (subaddress)： X.25 通信中，国家终端号 (NTN) 末端未分配的数字。如果网络供应商将所有数字分配给 NTN，就可能没有子地址。

子堆 (subhea)： Pascal 中通过调用 MARK 定界的堆的那部分。在一个堆中，子堆以类似堆栈的方式处理。

子对象 (subobject)： 是给定对象从属关系的对象。

子分量 (subcomponent)： Ada 语言中，一个分量或另一个分量的分量。

子服务器 (subserver)： 由在“系统程序控制器”控制下运行的服务器程序直接控制的系统资源或程序。

子过程 (subprocess)： 由其它过程启动的过程。当子过程完成运行时，控制传回主过程。

子集 (subset)： (1) 一种集合，其中的每个元素是其它指定集合的元素。(2) 比原始语言具有更少功能或更多限制的编程语言的变体格式。(3) 远程通信中的订户集，比如电话机。

子界标量类型 (subrange scalar type)： 在 Pascal 中，定义值限于某些基本标量类型值的子集中的变量的类型。另见基本标量类型 (*base scalar type*)。

子进程 (child process)： 在操作系统中，由父进程启动的进程，共享父进程资源。

子空间 (children spacing)： 在 AIXwindows 和“增强的 X-Windows”中，由另一个小窗口管理的小窗口称为管理父小窗口的子。父小窗口通常控制子小窗口在父小窗口边界内的实际空间和布置。

子类型 (subtype)： (1) 标识正被查询的 DLC 的 IOCINFO ioctl 变量。(2) Ada 语言中某类型的子类型，它表现一个子集的此种类型的值的特征。此子集由此类型上的约束确定。子类型的值的集合中的每个值属于此子类型，并满足确定此子类型的约束。

子类 (subclass)： 从更高级类继承资源的小窗口的类。

子例程标识 (Subroutine ID)： 与每个包含在应用程序内的子例程相关的唯一标识。

子例程切换表 (subroutine switch table)： 包含处理例程的特定处理程序例程的地址。

子例程 (subroutine)： (1) 一组有序的语句或编码指令，它们可用于一个或多个的计算机程序且可处于计算机程序的一个或多个点上。(2) 可作为另一个例程的一部分的例程。另见例程 (*routine*)。(3) 活动进程提出的请求，它请求系统内核的服务。另见宏 (*macro*)。

子命令 (subcommand)： 由先前发出的命令要求的工作的作用域中的操作请求。

子目录 (subdirectory)： 文件系统层次结构中，包含在另一个目录中的目录。

子区节点 (subarea node)： 数据通信中的节点，它使用网络地址来路由，而且它的路由表受到网络配置更改的影响。子区节点可以为外围节点提供的边界功能支持。

子设备 (child device)： 一个层次位置术语。它指出什么可以连接到父设备。例如，SCSI 磁盘可以是 SCSI 适配器的子设备。

子树 (subtree)： 更低级别的目录结构。

子通道 (subchannel)： 在 S/370 体系结构中定义用来执行传送给设备的逻辑通信路径。

子网地址屏蔽 (subnet address mask)： 本地系统使用的位屏蔽，用它来确定目标是否和源在同一网络上，或确定是否可以通过本地接口之一直接访问目标。

子网地址 (subnet address)： 本地主机地址的再分的部分，保留它们来指示子网。子网寻址允许由多个网络组成的自主系统共享同一个因特网网络地址。

子网 (subnet)： 单个网络内一组多个逻辑网络部分之一，比如可以由“TCP/IP 接口程序” (TCP/IP Interface Program) 创建。与子网 (*subnetwork*) 同义。

子网 (subnetwork)： 子网 (*subnet*) 的同义词。

子文件夹 (subfolder)： CDE 中包含在另一个文件夹中的文件夹 (有时称作父文件夹 (*parent folder*))。当论及命令行活动时，这可能称作子目录 (*subdirectory*)。

子系统 (subsystem)： (1) 一种辅助或从属系统，通常能够独立操作或与控制系统同步操作。(2) 通信的一部分，它处理远程系统要求，并从应用程序分离大部分与系统相关的注意事项。

子小窗口 (child widget)： 在 AIXwindows 和“增强的 X-Windows”中，一个由另一个小窗口管理的小窗口称为管理父小窗口的子。例如，小窗口 **Composite** 典型地管理与它们连接的子小窗口 **Primitive**。父小窗口通常控制子小窗口的布置及何时如何映射。当删除父小窗口时，也同时删除所有由该父小窗口控制的子小窗口。另见子小配件 (*child gadget*)。

子小配件 (child gadget)：一个无窗口的子小窗口。另见子小窗口 (*child widget*)。

子主机 (subhost)：除了与另一个 (通常更高级别) 系统通信，还控制连接的工作站的通信系统。

子资源 (child resource)：属于一个安全资源 (文件或库)，使用父资源的用户列表。一个子资源只可以有一个父资源。

子 (child)：(1) 属于一个安全资源 (文件或库)，使用父资源的用户列表。一个子资源只可以有一个父资源。(2) 在操作系统中，子是由父进程启动的进程，共享父进程资源。对照父 (*parent*)。(3) 在“增强的 X-Windows”和 AIXwindows 中，一个第一级 (first-level) 子窗口。一个由另一个小窗口管理的小窗口称为管理父小窗口的子。例如，小窗口 **Composite** 通常管理与它们连接的子小窗口 **Primitive**。父小窗口通常控制子小窗口的布置及何时如何映射。

子 (children)：子 (*child*) 的复数形式。

自顶向下 (top-down)：解决问题的一种方法，它从抽象的最高级别开始，往最低级别继续。

自动编号引用 (autonumber reference)：从自动编号创建的硬拷贝或软拷贝引用。自动编号引用自动重新编号以反映其相应自动编号的状态。在软拷贝中，自动编号引用与超文本链接相关联，以便链接可以跳转到相应的自动编号。另见自动编号 (*autonumber*)。

自动编号 (autonumber)：在 Interleaf 桌面排版软件中，与每个组件前缀相关联的唯一号码。因为每个自动编号有唯一的值，所以这些号码用于创建文本中其它地方出现的组件的引用。在硬拷贝中，自动编号可用于创建页面引用，当其相关联的组件移动到不同页面时进行自动重新编号。在超文本中，自动编号用于创建跳转到相应组件的超文本链接。对于有序列表项组件和图片标题组件，自动编号是可见的并用于以正确的顺序对这些项系列编号。

自动变量 (automatic variable)：进入例程时分配，返回时取消分配的一种变量。对照静态变量 (*static variable*)。

自动拨号器 (ADU) (automatic dialing unit, ADU)：一个能够自动生成拨号数字的设备。

自动拨号器 (autodialer)：请参阅自动拨号器 (*automatic dialing unit*)。

自动滚动 (automatic scrolling)：当光标移动到窗格边界时，自动发生的滚动操作。

自动呼叫器 (ACU) (auto-call unit, ACU)：在 X.25 通信中，一个自动呼出及应答呼叫的设备。

自动呼叫器 (automatic calling unit, ACU)：一个允许主机自动拨远程设备号码的设备。

自动呼叫 (auto-call)：工作站无需操作员操作在交换线路上呼出呼叫的能力。对照手动呼叫 (*manual call*)。

自动换行 (word wrap)：在字处理中，如果文本超出页边距，而自动将文本移动至下一行的功能。

自动推 (autopush)：一种流机制，使得当流设备打开时自动把预先指定的模块列表推到流。

自动应答 (auto-answer)：工作站无需操作员操作接收来自交换线路的呼叫的能力。

自动执行 (autoexec)：登录时运行的一个或一系列命令。

字段 (field)：(1) 在记录或者面板中的一个区域用于包含特定类别的数据。(2) 记录的最小可识别组成部分。一个记录的单独可寻址的子部分，包含记录中数据的特定组成部分。例如，一个关于雇员的数据记录或许可分成包含雇员名称、身份证号和薪资的字段。(3) 在显示空间的一个区域，程序从此接受输入。(4) 在视频中，一个字段是一个半帧。字段的绘制是每秒 60 次。另见结构字段 (*structured field*)。

字符变量 (character variable)：C 语言中，在程序运行期间其值可以改变及数据类型为 **char** 或 **unsigned char** 的数据对象。

字符表达式 (character expression)：一字符常量或变量、字符数组元素、字符子串、字符值函数引用或以连接运算符分隔 (圆括号可选) 的前述元素序列。

字符常量 (character constant)：(1) 一常量值，其数据属性为字符。(2) 编程语言中用单引号扩起来的字符或转义序列。

字符串常量 (string constant)：括在双引号中的字符。

字符串寄存器 (string register)：拥有由记号调用的已定义的字符串值的寄存器。另见记号 (*token*)。

字符串值 (string value)：指定字符串的值。AIXwindows 中标识 Text 小窗口的字符串的值。

字符串 (character string)：连续的字符序列。C 编程语言中，字符串必须以空字符结束。

字符串 (string)：(1) 各实体 (比如字符或物理元素) 的线性顺序。字符串的示例有字母字符串、二进制元素字符串、位字符串、字符字符串、搜索字符串和符号字符串。(2) Pascal 中，预先定义类型为 **STRING** 的对象。(3) 用于存储和处理文本的编程语言中数据的格式。在 C 语言代码中，字符串被当作类型为 **char** 的一维数组。

字符单元 (character cell)：以像素表示的字体的实际宽度和高度。另见边界框 (*bounding box*)。

字符集 (character set)：因为特定原因而使用的一组字符；例如，打印机可以打印或键盘可支持的字符集。

字符键 (character key)：(1) 一键盘键，允许用户输入显示在键上的字符。另见功能键 (*function keys*)。 (2) 字处理中用来处理文本（一次一个字符）的控件。

字符类型 (character type)：由字母数字字符构成的数据类型。另见数据类型 (*data type*)。

字符类 (character class)：(1) 与输入流中的单一字符匹配的字符范围。(2) 按顺序或用方括弧 []、括号扩起来的字符集合。

字符删除 (character delete)：文本数据中擦除当前光标所在位置的字符然后把任何后续文本左移一个字符位置的操作。

字符设备 (character device)：一次处理一个字符的设备。另见字符特殊文件 (*character special file*)。

字符数据索引 (character data indexing)：一种输入/输出优化，通过在一个小索引文件中搜索字符串及使用获得的索引在主数据文件中查找记录来减少 I/O 访问时间。

字符特殊文件 (character special file)：一个提供访问输入输出设备的特殊文件。字符接口用于不使用块 I/O 的设备。另见字符设备 (*character device*)、块文件 (*block file*)、特殊文件 (*special file*) 和块特殊文件 (*block special file*)。

字符图形 (character graphics)：(1) 字符的可视表示，以彩色或单色（像素）定义。(2) 由符号（以等宽字体打印）组成的图形。一些符号是独立的；其它的用于装配成更大的图形。

字符位置 (character position)：显示器上的字符位置。

字符文字 (character literal)：源程序中的一个符号、数量或常量，其自身就是数据，而不是对数据的引用。对照数字文字 (*numeric literal*)。

字符显示器 (character display)：一个使用字符发生器在屏幕上显示预定义图像字符框（字符）的显示器。这类显示器一次无法进行任何少于一个字符框的屏幕寻址。对照全点解析显示器 (*All Points Addressable display*)。

字符转换 (character translation)：在国际化字符支持中，命令 **dd** 和各种转换子例程在扩展字符和 ASCII 转义字符串之间转换以保持唯一的字符信息。

字符串子串 (character substring)：一字符串中的邻接部分。

字符 (character)：字母、数字或其它符号。

字节边界 (byte boundary)：基于 8 位间隔的内存寻址。每个内存位置包含一 8 位值，以十进制符号表示法的范围为 0 到 255。

字节常量 (byte constant)：FORTRAN 语言中字节类型的命名常量。

字节类型 (byte type)：FORTRAN 语言中，作为一个单元操作的二进制字符，通常比一个计算机字节。

字节顺序 (byte order)：在“增强的 X-Windows”中，由服务器为像素图或位图数据定义的字节顺序。具有不同本地字节顺序的客户机必要时必须交换字节。

字节 (byte)：(1) 表示 1 字符所需的存储量；一字节为 8 位。(2) 作为一个单元操作的二进制字符，通常比字短。(3) 包含某个位数（通常 8 位）的字符串，作为一个单元操作，表示一个字符。(4) 一组相邻的 8 个二进制数字，表示 1 个 EBCDIC 字符。(5) X.25 中，一字节称为一个八位元。另见位 (*bit*)。

字距调整 (kerning)：字符的布置，如它们的边界框（字符框）重叠。这允许更自然的字符间空白间距。

字母数字集 (alphanumeric set)：字符集包含大小写字母和数字，但不包含符号。另见字母数字字符 (*alphanumeric character*)。

字母数字字符 (alphanumeric character)：包含字母、数字和其它符号，如标点符号和数学符号。另见字母字符 (*alphabetic character*)。

字母字符 (alphabetic character)：一种语言中使用的字母或其它符号，不包括数字。通常为从 A 到 Z 的大小写字母加上特定语言允许的其它特殊符号（如 \$ 和 _）。另见字母数字字符 (*alphanumeric character*)。

字母 (letter)：字符集中从 A 到 Z 的一个大写或小写字符。

自然或社会科学格式 (natural or social science format)：受自然与社会科学的作者喜爱的书目样式。

字体表 (font table)：已装入系统的所有字体的列表。另见字体列表 (*font list*)。

字体单元 (font unit)：作为一个整体的任何字符组，特别用于值、变量和其它属于字体特征的数据的确定。

字体结构 (font structure)：一种数据结构，它包含所有创建字体集合所需的信息。

字体列表 (font list): 在 AIXwindows 中, 由 *fontlist* 参数指定的可用字体的列表。也即, 用于格式化源文档的字体列表。另见字体表 (*font table*)。

字体提高 (font ascent): 指定的字体的字符高度。

字体 (font): (1) 特定样式中的字符集。另见光栅字体 (*raster font*) 与图元字体 (*primitive font*)。 (2) 在“增强的 X-Windows”中, 符号的集合, 通常为字符。协议不转换或解释字符集。客户机说明用于存取字符数组的值。包含额外的度量信息来确定内部字符与内部行空间的一种字体。 (3) 具相同大小、样式与字型的图形字符的完整集合。例如, Times 12 点的字体与 Times 24 点的字体大小是不同的, 与 Helvetica 12 点字体的样式是不同的, 并与 Times 12 点粗体字的字型是不同的。

字形 (glyph): (1) 通常指某种字体的字符图像。 (2) 一种图形符号, 其外观传达信息; 例如指示控制光标移动方向的光标键上的垂直和水平箭头。

字形 (type style): 同一个字体集中给定大小、样式和设计的字符的形式。

自由表 (free list): 每个文件系统上可用块的列表。

自由块列表 (free-block list): 请参阅自由表 (*free list*)。

自治系统 (autonomous system): 一组网络和网关, 管理机构对其负有职责。一个自治系统可以较小或者很大。

字 (word): (1) 存储器中连续的 32 位 (四个字节) 的可寻址单元。字的第一个字节的地址可平均分为四份。与全字 (*fullword*)、机器字 (*machine word*) 和计算机字 (*computer word*) 同义。对照半字 (*halfword*)。 (2) 看作是用于给定用途的单元的字符串。

总线部件识别 (bus unit identification): I/O 段寄存器中的字段, 解码之后为特定硬件实现选择 IOCC。

总线 (bus): (1) 处理器中的物理设施, 通过其可将数据传送到所有目标位置, 但其中只有可寻址的目标位置可以通过适当约定来读取。 (2) 一种处理器互相连接成系列的计算机配置。 (3) 传输信号或电力的一个或多个导体。

纵向颠倒 (portrait upside-down): 一种页面方向, 当纸从打印机中出现时, 打印图像的顶部在纸末尾边缘。

纵向显示 (portrait display): 高比宽度长的矩形显示。另见横向显示 (*landscape display*)。

租用实施 (leased facility): 非交换线路 (*nonswitched line*) 的同义词。

组标识 (group ID, GID): 与某一特定组名相应的编号。通常, 组标识可以在将组名看作一个值的命令中进行替换。与组号 (*group number*) 同义。

组号 (group number): 组标识 (*group ID*) 的同义词。

组合管理器 (Composite Manager): 在 AIXwindows 中, 一个对于处理一个或多个自己的子小窗口具有特殊了解的管理器小窗口。通常管理器小窗口不知道自己的子 (窗口), 但是标题栏小窗口和滚动条小窗口可以注册为某类型组合管理器小窗口, 即面板小窗口, 面板小窗口可以正确控制标题栏和滚动条小窗口的位置。

组合类型 (composite type): 在 Ada 语言中, 组合类型是其值拥有组件的类型。两种组合类型: 数组类型和记录类型。

组合小窗口类 (Composite widget class): “增强的 X-Windows”中的元类, 它不实例化自身拥有的任何小窗口, 但是提供允许父小窗口管理布局和它们的子小窗口和小配件的映射的资源和功能。

组合小窗口 (composite widget): 在“增强的 X-Windows”中, 小窗口是子 (窗口) 的任意的、实现定义的集合的容器。这些子 (窗口) 可以由组合小窗口自身、其它客户机或两者组合来实例化。组合小窗口包含用来管理任何子小窗口几何图形 (布局) 的方法。组合小窗口是核心小窗口的子类。另见小窗口 (*widget*)。

组件栏 (component bar): 在 Interleaf 文档中, 位于左边距的黑色栏, 它包含包含在该文档中的组件的组件标识。

组件转储表 (component dump table): 一个结构, 内核组件用来识别应该由内核转储程序收集的数据结构。

组件 (component); 组成部分; 部件: (1) 结构化类型或值的一部分, 如数组元素或记录字段。 (2) 在 AIXwindows 或“增强的 X-Windows”中, 组成交互式用户界面的小窗口、小配件或其它图形对象。 (3) 在 Ada 语言中, 一个作为更大值的一部分的值, 或一个作为更大对象的一部分的对象。

组名 (group name): 使系统识别一组用户所用的唯一名称, 包含以字母、#、\$ 或 > 开头的一到八个字母数字字符。

祖先 (ancestor): 在“增强的 X-Windows”中, 具有下级小窗口的小窗口。换句话说, 一个下级小窗口的上级或祖先。如果 *W* 是 *A* 的下级, 那么 *A* 是 *W* 的祖先。

组相关高速缓存 (set-associative cache): 一种高速缓存, 其中两个或四个 (或更多) 线路对应虚拟地址字段的每个可能值 (这些字段标识在高速缓存查找过程中将被询问的线路)。

组相关性 (set associativity)：高速缓存设计的一方面，它确定了多少高速缓存线路可与给定内存位置相关联。四路组相关的高速缓存可以在四条高速缓存线路之一包含给定内存。另见高速缓存线路 (*cache line*)。

组帧错误 (framing error)：一种异步传输错误，通常由每个字符的位数引起，这些字符的位数在发送和接收工作站上设置得不相同。

组指定 (Group Appointment)：在 CDE 中，按照日历对一组人员进行指定，这组人员通过“组指定编辑器”来调度。

组织 (organization)：关联可以被授权公共访问权的用户组的数据，它通常与管理策略关联。同样，主体标识符的第三个字段。

组 (group)：(1) 可以共享受保护资源的访问权限的用户集合。(2) 合起来由单个名称辨认的名称列表。(3) 一系列记录逻辑上连接在一起。(4) 在全屏视窗或全屏面板上作为一个集合不断重复的许多行。(5) 在所有记录中，对于某特定字段具有相同值的相关记录的集合。

最大传输单元 (maximum transfer unit, MTU)：(1) 一个“网际协议”(IP) 数据报所包含的最大数量字节。(2) 可以用单一的帧被传输到一个特定的网络接口的最大数量的数据。

最低限度一致 (minimally consistent)：对象满足在类中提前定义的各种条件。

最短作业优先 (shortest-job-next, SJN)：排列作业的方法，其中最短作业首先打印。对照先来先服务 (*first-come-first-served*)。另见 规程 (*discipline*)。

最终生产者 (ultimate producer)：在输入和输出操作中的数据源。最终生产者可以是文件、设备或内存中的字节数组。

最终消费者 (ultimate consumer)：在输入和输出操作中的数据的目标。最终用户可以是文件、设备或内存中的字节数组。

作业队列 (job queue)：等待系统处理的作业列表。

作业号 (job number)：分配给一个作业的号码，使它在进入系统时和其它作业相区别。

作业 (job)：(1) 由用户定义的由系统执行的工作单元。术语“作业”有时代表任务，如一组程序、操作系统的文件和控制语句。(2) 一个或多个相关的过程或程序组成一个过程，由适当的作业控制语句来标识。另见 进程 (*process*)。

作用域运算符 (scope operator, (::))：在 C++ 中它定义右边自变量的作用域。如果左边自变量为空，则此作用域为全局。如果左边自变量是类名，则作用域在该类中。

作用域 (scope)：(1) 源程序的一部分，在其中变量可以传达其值。(2) 某说明可应用于其中的程序的部分。对于 Ada 编程，请参阅 声明 (*declaration*)。

左乘 (premultiplication)：在 GL 中，自左的矩阵乘法。如果矩阵 M 被矩阵 T 自左乘，则结果为 TM。

左调整 (left-adjust)：对在左页边距或在行最左边字符的制表位设置处的文本行或最左边位值处的文件进行对齐的过程。

左页边距 (left margin)：页面或屏幕上左边缘和最左面的字符位置之间的页面或屏幕区域。

坐标系 (coordinate system)：一给定的约定，用于在给定显示器或窗口定位像素，其中在 AIXwindows 中，X 是水平轴，Y 是垂直轴。取决于使用中的约定，原点为位于左上角或左下角的 [0, 0]。对于窗口，原点是位于边界内侧坐上或左下角（取决于使用中的约定）。坐标是离散的，以像素表示。每个窗口和像素图都有自己的坐标系。

特殊文字及符号

‘Hello, World’：介绍由 Tran-C 提供的基本构造的独立样本程序。

数字

2-1/2 维 (2-1/2 dimension)：一个用于描述正交视图间有限联合的绘图术语；透视图的两维一旦创建就无法修改，也无法旋转。允许半自动创建等轴视图。

3270 设备仿真 (3270 Device Emulation)：支持允许某一系统的本地或远程设备对另一个系统显示为 3270 设备。

3270 主机连接程序 (3270 Host Connection Program, HCON)：允许系统部件仿真连接到 System/370 主机的工作站或打印机的程序。另见 会话概要文件 (*session profile*) 和 HCONuser。

A

A：请参阅 安培 (*ampere*)。

ACK：请参阅 确认字符 (*acknowledgment character*)。

ACK0: 传输控制字符, 用于连贯主动应答; 说明收到的文本没有传输错误。

ACK1: 传输控制字符, 用于非连贯主动应答; 说明收到的文本没有传输错误。

ACL: 请参阅访问控制列表 (*access control list*)。

ACM: 美国计算机协会。

ACSE: 关联控制服务单元 (Association Control Service Element)。

ACU: 请参阅自动呼叫器 (*automatic calling unit*) 或自动呼叫器 (*auto-call unit*)。

ADK (应用程序开发者工具) (ADK, Application Developer's Kit): License Use Management 系统的组件, 软件开发者用来为软件产品定义及创建许可证。

ADU: 请参阅自动拨号器 (*automatic dialing unit*)。

AIXwindows 工具箱 (Toolkit): 一种面向对象的 C 语言数据结构和子例程的集合, 补充“增强的 X-Windows”工具箱并简化交互式客户机应用程序界面的创建。

AIXwindows 环境: 一基于 OSF/MOTIF 的软件图形用户界面环境, 包含 AIXwindows 工具箱、图形库、窗口管理器和运行在兼容操作系统顶层的桌面。

AIXwindows 用户界面定义 (User Interface Definition, UID): 一种包含包括在特定用户应用程序中的所有对象定义的文件。UID 是 AIXwindows 用户界面语言 (UIL) 的子集, 对于其所属的某个特定应用程序, UID 和 UIL 的功能类似。另见 *AIXwindows 用户界面语言 (User Interface Language, UIL)*。

AIXwindows 用户界面语言 (User Interface Language, UIL): 一种用来描述用户界面初始状态的精确编译语言。UIL 指定构成界面的小窗口、小配件和复合对象; 同时还确定任何时候因为用户交互作用而导致界面状态更改时要调用的子例程。另见 *AIXwindows 用户界面定义 (User Interface Definition, UID)*。

AIXwindows 资源管理器 (Resource Manager, MRM): 一种数据库管理系统, 允许用户有效检索及分配信息, 如特殊值和其它属性, 特别是允许用户选择颜色、字体和其它资源偏爱的应用程序。MRM 系统包含运行时访问 AIXwindows 用户界面语言 (UIL) 和创建用户界面的库子例程。另见 *AIXwindows 用户界面语言 (User Interface Language, UIL)*。

ALU: 逻辑运算单元。

amp: 请参阅安培 (*ampere*)。

ANSI: 请参阅美国国家标准协会 (*American National Standards Institute*)。

APA: 请参阅全点解析显示器 (*All Points Addressable Display*)。

APAR: 授权程序分析报告。由当前程序发行版中未更改的可疑缺陷产生的问题报告。

API: 请参阅应用程序接口 (*application program interface*)。

APL: 一编程语言。一种用于多种应用 (商业数据处理、系统设计、数学和科学计算、数据库应用和数学教学及其它学科) 的通用语言。

AppBuilder: CDE 中, 一种用于构造图形用户界面的软件应用程序。

APPC: 请参阅高级程序到程序通信 (*Advanced Program-To-Program Communications*)。

applet: 一种用于在因特网上分发的程序, 可包含在 HTML 页面中, 正如同包含图像一样。

ARK (管理员运行时工具包) (ARK, Administrator Runtime Kit): 在 License Use Management 中, 许可证软件产品的运行时环境。由网络许可证服务器守护程序 (*netlsd*) 和相关工具: *ls_admin*、*ls_rpt* 和 *ls_stat* 构成。

ARP: 请参阅地址解析协议 (*Address Resolution Protocol*)。

ARPA: (美国国防部) 高级研究计划署。

ARPA RFC 822: ARPA 因特网文本消息标准。

ASCII: 请参阅美国国家信息交换标准码 (*American National Standard Code for Information Interchange*)。

ASCII 平面文件 (ASCII flat file): 请参阅平面文件 (*flat file*)。

ASCII 字符 (ASCII characters): 组成 ASCII 字符集的字符。ASCII 字符集由 7 位 (7-bit) 控制字符和符号字符构成。另见美国国家信息交换标准码 (*American National Standard Code for Information Interchange*)。

ASCIIZ 格式 (ASCIIZ format): 以 null 字符结尾的字符串。

ASN.1: 请参阅抽象语法符号一 (*Abstract Syntax Notation One*)。

ATE: 请参阅异步终端仿真 (*Asynchronous Terminal Emulation*)。

ATM: 异步传送方式。一种信元交换 (cell-switching)、面向连接的技术。在 ATM 网络中, 连接到网络的终端工作站使用专用的全双工连接。

ATR: 请参阅地址转换寄存器 (Address Translation Register)。

AUTOLOG: 在由 3270 主机连接程序 2.1 和 1.3.3 中提供的一个菜单驱动的实用程序, 用于创建登录过程。

AZERTY 键盘 (AZERTY keyboard): 一种键盘, 其中从上往下第二行 (D 行) 的键标号为 (从左到右): A、Z、E、R、T、Y、U、I、O 和 P。另见 *QWERTY 键盘 (QWERTY keyboard)*。

azico: 在 Performance Toolbox 中, 一种用来分析性能记录的工具。

a.out: (1) 某些命令缺省产生的输出文件。缺省情况下, 该文件是可执行的并包含可用于跟踪调试程序的信息。(2) 由命令 **cc** 创建的目标文件格式, 并由子例程 **exec** 使用的目标文件格式。

B

B 样条三次曲线 (B-spline cubic curve): 计算机图形学中, 三次样条曲线近似于一四控制点集, 其属性为斜率和曲率连续通过控制点集。另见参数三次曲线 (*parametric cubic curve*)。

backfacing 多边形 (backfacing polygon): 在 GL 中, 多边形顶点在屏幕空间中以顺时针方向显示。如果启用 backface 选择, 此类多边形将不会绘出。

BASIC (初学者通用符号指令代码)
(**BASIC, beginner's all-purpose symbolic instruction code**): (1) 一种为交互式系统设计的编程语言。最早由 Dartmouth 大学为鼓励人们使用计算机进行简单问题求解操作而开发。(2) 一种语句数量少, 语法简单的高级编程语言。BASIC 设计成易学易用, 广泛应用于微机交互式应用程序。

BCUG: 请参阅双边闭合用户组 (*bilateral closed user group*)。

BEL: 响铃符。当需要人们注意时, 一个激活报警或其它注意设备的控制字符。与控制台报警符 (*console bell*) 同义。

BER: 请参阅基本编码规则 (*Basic Encoding Rules*)。

BGP: 边界网关协议。

bin 收集 (bin collection): 一种将审计记录写到临时 bin 文件的收集审计数据的方法。数据经守护程序 **auditbin** 处理之后, 把记录写到审计跟踪文件进行存储。

BIND 密码 (BIND password): 两个通信安全性密码之一。在 LU-LU 会话中, 系统检查远程系统的密码以验证用户连接的程序是正确的程序。另见节点验证 (*node verification*) 和通信权限密码 (*communications authority password*)。

BIOS (基本输入 / 输出系统) (BIOS, Basic Input/Output System): 在个人计算机 AT 中, 控制基本硬件操作 (例如与软盘驱动器、硬盘驱动器和键盘交互) 的微码。

bis: 描述替代主推荐的次推荐。

BIST: 内置自测试。

BIU 分段 (segmenting of BIUs): 路径控制的一个可选功能, 它将从传输控制接收到的基本信息单元 (BIU) 划分为两个或更多的路径信息单元 (PIU)。第一个 PIU 包含 BIU 的请求报头, 通常还有响应单元 (RU) 的一部分。剩余的一个或多个 PIU 包含 RU 的剩余部分。

blit: 位块传送。

BLT: 请参阅位块传送 (*bit block transfer*)。

BNC: 用于某些同轴电缆的连接器件。

BNU: 请参阅基本网络实用程序 (*Basic Networking Utilities*)。

BOS (基本操作系统) (BOS, Base Operating System): 控制计算机系统资源和操作的程序集合。

bosinst.data: 控制 BOS 安装程序操作的文件。

bpi: 位数/英寸, 存储产品的线密度度量。

bps: 位/秒。串行传输中设备或通道的传输一个字符的即时位速率。另见波特 (*baud*)。

BSC: 请参阅二进制同步通信 (*Binary Synchronous Communication*)。

BSD: 伯克利软件分发。

BTU: (1) 英国热量单位。(2) 组 SNA 中, 基本传输单位。

BUID: 请参阅总线单元识别 (*bus unit identification*)。

C

C: 摄氏或国家或地区。

C 接口 (C interface): 依据 ANSI 标准化的 C 变体这一级别来定义的接口。

C 库 (C library)：一种系统库，包含常用 C 语言文件访问子例程、字符串运算符、字符运算、内存分配和其它函数。

C 语言 (C language)：一种通用编程语言，是操作系统的主要语言。

CAD/CAM：计算机辅助设计 / 计算机辅助制造。

CAI：请参阅计算机辅助教学 (*computer-aided instruction*)。

CASE：计算机辅助软件工程 (Computer Assisted Software Engineering)。帮助开发复杂应用程序的工具或程序集。

CASE 标号 (case label)：字 case 之后跟一个常量表达式和一个冒号。当选择程序求值到常量表达式的值时，将处理跟随在 CASE 标号后面的语句。

CASE 标号 (CASE label)：Pascal 中，处在 CASE 语句分支中的语句之前的值或值范围。当选择程序求值到 CASE 标号值时，将处理跟随在 CASE 标号后面的语句。

case 子句 (case clause)：C For AIX 中的 switch 语句，case 标号后跟任意数量的语句。

cast：C 语言中把操作数值转换为指定标量数据类型的表达式 (运算符)。

catch 块 (catch block)：C++ 程序里的块，当与其参数的匹配异常抛出时接收控制。每个 catch 块与一个 try 块 (try block) 关联。

CBEMA：计算机和商业设备制造商协会 (Computer and Business Equipment Manufacturers Association)。

CCITT：国际电报电话咨询委员会 (Comite Consultatif International Telegraphique et Telephonique)。另见国际电话电报咨询委员会 (*Consultative Committee on International Telegraphy and Telephone*)。

CCW：通道控制字。定义好的控制编码，用来控制 System 360/370 通道上 I/O 部件的操作。

CD：载波检测。另见 DCD。

CDE：Common Desktop Environment (公共桌面环境) 的缩写，一种运行于 UNIX 上的图形用户界面。

CDLI：公共数据链路接口 (Common Data Link Interface)。与内核服务交互以便为套接字和流 (STREAMS) 接口提供支持的设备驱动程序。

CDS：单元目录服务 (Cell Directory Service)。

CDS 定义属性 (CDS-defined attribute)：一个 CDS 与名称关联的标准属性。一个具有相同意义的特定 CDS 定义

属性，不论与其关联的是何种类型的项 (交换所、目录和对象)。然而不同类型的项可以有不同的 CDS 定义属性。例如，每个 CDS 名称有如下 CDS 定义属性：创建时间戳记 (CDS_CTS)、更新时间戳记 (CDS_UTS) 和访问控制集 (CDS_ACS)。除这些属性之外，还有一个具有唯一的 CDS 定义属性的软链接，包含其失效时间和所指向的名称。

CDS 目录 (CDS directory)：CDS 名称空间中，一个用来存储位于一个名称 (目录名称) 下的项的逻辑单元。除对象项之外，目录可以包含软链接和子指针。可以对目录复制、删除及控制访问。目录的每个物理实例称为副本。

CDSPI：单元目录服务可移植接口 (Cell Directory Service Portable Interface)。

CDSTL：请参阅连接数据集至线路 (*connect data set to line*)。

CD-ROM：格式为光学读取光盘的高容量只读存储器。

CEC：中央电子柜 (Central Electronics Complex)。

CFM：立方英尺/秒 (Cubic feet per minute)。

CGA：彩色图形适配器 (Color Graphics Adapter)。

CGM：请参阅计算机图形元文件 (*Computer Graphics Metafile*)。

char：“对象数据库管理器”中用于把变量定义为固定长度、以空字符结尾的字符串的结束描述符。另见结束描述符 (*terminal descriptor*)。

char 说明符 (char specifier)：关键字 **char** 和 **unsigned char** 描述了变量表示的数据类型。

CHPID：通道路径标识 (Channel path identifier)。

CICS：请参阅客户信息控制系统 (*Customer Information Control System*)。

CID：请参阅连接标识 (*connection identifier*)。

CLA：请参阅通信线路适配器 (*communications line adapter*)。

CLAW：请参阅公共工作站链路访问 (*Common Link Access to Workstation*)。

CLK：请参阅时钟 (*clock*)。

CMOS：互补金属氧化物半导体 (Complementary metal-oxide semiconductor)。CMOS 是一种结合了 n-type 半导体和 p-type 半导体电属性的技术。

COBOL: 面向商业的通用语言 (Common business-oriented language)。一种基于英语的高级编程语言, 主要用于商业应用程序。

col: 缓冲排字输出以允许在不支持反卷的打印机和工作站上打印的排字机后处理程序。

confirm: X.25 通信中对到达的清除指示和复位指示信息包的响应。

CONFIRMED: 对 COMFIRM 请求的响应, 表示远程站点接收到该传输, 没有检测到任何错误。

Constraint: AIXwindows 中的一类对象, 从其可继承唯一的资源集。例如, **PanedWindow** 小窗口可通过使用继承的 **XtNmin** 和 **XtNmax** 约束资源指定自身子 (小窗口) 的大小。与每个小窗口相关联的引用资料指定从 **Constraint** 类继承的资源。

continuation 引用 (continuation reference): continuation 引用描述所有或部分操作的执行如何在不同的 DSA 上继续。

continue 语句 (continue statement): 一种 C 语言控制语句, 包含关键字 **continue** 和一个分号。

Core: 在 AIXwindows 中, **Core** 是顶层超类, 所有小窗口和小配件都从其派生。**Core** 由三个子类构成 (**Object**、**RectObject** 和 **WindowObj**), 选择性地提供 AIXwindows 工具箱中所有小窗口和小配件所需的外观资源和行为资源。

CP: 请参阅控制程序 (control program)。

CPS: 字符 / 秒。

CPU: 请参阅中央处理器 (central processing unit)。

CPU 时间 (CPU time): 程序在 CPU 中运行或由操作系统服务的时间量。不包括与程序 I/O 相关的时间或其它进程占去该程序的 CPU 使用时间。

CPU 锁 (CPU lock): 请参阅节点锁定许可证 (nodelocked license)。

CRC: 请参阅循环冗余校验字符 (cyclical redundancy check character)。

CRQ: 请参阅呼叫请求 (call request)。

CRT: 请参阅阴极射线管 (cathode ray tube)。

CSC: 请参阅核心顺序控制器 (core sequence controller)。

CSMA/CD: 请参阅带冲突检测的载波监听多路访问 (carrier sense multiple access with collision detection)。

CSX: 基于主机的诊断程序。

CTC: 通道到通道 (Channel-to-channel)。

CTS: 清除发送。与 EIA-232 协议一起使用。

CUD: 请参阅调用用户数据 (call user data)。

CUG: 请参阅闭合用户组 (closed user group)。

CUT: 请参阅控制单元终端方式 (control unit terminal mode)。

C++ 库 (C++ library): 一系统库, 包含常用 C++ 语言文件访问子例程、内存分配和其它函数。

C++ 语言语句 (C++ language statement): 一个 C++ 语言语句包含零个或多个表达式的。所有 C++ 语言语句 (除块语句) 都以符号 ; (分号) 结束。块语句以符号 { (左大括号) 开始, 以符号 } (右大括号) 结束, 可包含任意数量的语句。

C++ 语言 (C++ language): 一种基于 C 语言的编程语言, 基于 C 语言, 由 Bjarne Stroustrup 开发。C++ 定义的数据类型称为类。类提供数据抽象, 由数据 (数据成员) 和对这些数据操作的操作 (成员函数) 组成。C++ 也实现了面向对象编程的继承概念 (通过类派生) 和多态性概念 (通过虚拟函数和重载)。

C-ISAM: C 语言索引顺序存取方法 (C-language Indexed Sequential Access Method)。

C-stub: 实现与通信网络连接的 DUA 部件。

C.A.T: 计算机辅助排字 (Computer-assisted typesetting)。

D

D 位 (D-bit): 在 X.25 通信中, 如果接收方需要端到端确认 (递送确认), 则数据包或呼叫请求包中的位设置为 1。

DAC: 请参阅数模转换器 (digital-to-analog converter)。

DAF: 目标地址字段 (Destination Address Field)。

DAS: 双连接站 (Dual-attachment station)。一种同时连接到主和次 FDDI 环的站。

DASD: 直接访问存储设备。一种存取时间有效地独立于数据位置的设备。信息的输入和检索无需引用以前访问的数据。DASD 同时包括固定和可移动存储设备。

DB: 数据库。

dB: 请参阅分贝 (decibel)。

DBA: 请参阅直接总线连接 (*direct-bus attached*)。

DBCS: 请参阅双字节字符集 (*Double-Byte Character Set*)。

DC: 请参阅DCU。

DCD: EIA-232 协议使用的数据载波检测。另见 CD 和解码 (*decode*)。

DCE: 请参阅分布式计算环境 (*Distributed Computing Environment*) 和数据电路终端设备 (*data-circuit-terminating equipment*)。

DCN: 分布式计算机网络 (*Distributed Computer Network*)。

DCU: 数据高速缓存单元。另见高速缓存 (*cache*)。

DD: 请参阅设备驱动程序 (*device driver*)。

DDN: 国防部网络 (*Department of Defense Network*)。

DEALLOCATE: 一个从本地事务程序除去指定对话分配的请求。

default 子句 (default clause): 在 C switch 语句中, 关键字 default 后跟冒号及一个或多个 C 语句。当 switch 语句中指定 case 标号的条件没有保留时, 则选择 default 子句。

define: 在“ODM 定制设备数据库” (*ODM Customized Devices Database*) 中创建条目并建立父设备和连接位置。

define 方法 (define method): 用来在“ODM 定制数据库”中创建设备实例。它将设备从未定义或不存在的状态转为已定义状态。

define 语句 (define statement): 预处理器语句, 它使预处理器用指定代码代替标识符或宏调用。

DEL: 请参阅delete 字符 (*delete character*)。

delete 字符 (DEL): (1) 主要用来删除错误字符或不需要的字符的控制字符。(2) 识别要从文件中删除的记录的字

delta: 变量的有限增量。

DES: 请参阅数据加密标准 (*Data Encryption Standard*)。

DFT: 请参阅分布式功能终端 (*distributed function terminal*)。

DHCP: 动态主机配置协议 (*Dynamic Host Configuration Protocol*)。一个应用程序层协议, 允许网络上的机器 (客户机) 从服务器获取 IP 地址和其它配置参数。

Dialog: 在 AIXwindows 中应用程序和其用户之间的双向文本接口。接口的形式为小窗口和小配件的集合, 包括 **DialogShell** 小窗口、**BulletinBoard** 小窗口 (或 **BulletinBoard** 小窗口的子类或某个其它容器小窗口), 加上各种子窗口, 包括 **Label**、**PushButton** 和 **Text** 小窗口。

DISC: 取消连接 (*Disconnect*)。

DIX 连接器 (DIX connector): 把以太网网络适配器连接到标准 (“厚”) 以太网本地局域网的设备。它的名称是依据以太网 (*Digital Equipment Corporation*、*Intel* 和 *Xerox*) 的主要开发者的名称命名。

DLC: 请参阅数据链路控制协议 (*data link control protocol*)。

DLE: 请参阅数据链路转义字符 (*data link escape character*)。

DLPI: 数据链路提供者接口 (*Data Link Provider Interface*)。

DLS: 数据链路服务 (*Data link service*)。

DLS 提供者 (DLS provider): 在 X.25 通信中, 指提供数据链路提供者接口 (*Data Link Provider Interface*, *DLPI*) 服务的数据链路层协议。

DLS 用户: 在 X.25 通信中, 指访问数据链路层服务的用户级应用程序或用户级 (或内核级) 协议。

DLSAP (数据链路服务访问点, data-link-service access point): 在 X.25 通信中, 指数据链路服务 (DLS) 用户使自己与 DLS 提供者连起来以访问数据链路服务的点。

DLSAP 地址 (DLSAP address): 在 X.25 通信中, 用于区分和定位指向 DLS 提供者的特定 DLS 用户访问点的标识符。

DM: 请参阅断开连接方式 (*disconnected mode*)。

DO 变量 (DO variable): 在 FORTRAN 中, 在 DO 语句中指定的变量, 与之相关的 DO 循环每执行一次迭代, 这个变量就被加或减, 从而控制循环的迭代次数。

Do 循环的范围 (range of a DO loop): 实际跟随 DO 语句, 直到包含 DO 语句指定的最后一个语句的那些语句。

DO 循环 (DO loop): 在 FORTRAN 中, 由 DO 语句重复调用的语句范围。

do 语句 (do statement) : (1) 在 C 语言中, 包含后跟一条语句 (操作)、关键字 **while**、括在括号中的条件表达式的关键字 **do** 的循环语句。 (2) 用于把很多语句组到一个过程中的语句。

DOS (磁盘操作系统, Disk Operating System) : 个人计算机使用的磁盘操作系统。

DPS: 请参阅显示 *PostScript (Display PostScript)*。

DR1I: 限定响应 1 指示器。

DR2I: 限定响应 2 指示器。

DRAM: 请参阅动态随机访问存储器 (*dynamic random access*)。

DSI: 请参阅数据存储中断 (*data storage interrupt*)。

DSR: 数据集就绪; 与 EIA - 232 协议一起使用。关于更多信息请参阅调试解调器手册。

DTE: 请参阅数据终端设备 (*data terminal equipment*)。

DTR: 请参阅数据终端就绪 (*data terminal ready*)。

DWM: 请参阅无盘工作站管理器 (*Diskless Workstation Manager*)。

E

EBCDIC: 请参阅扩展二进制编码的十进制代码 (*extended binary-coded decimal interchange code*)。

EBCDIC 字符 (EBCDIC character) : 包含在 8 位 EBCDIC 集里的任何符号之一。

EC: 工程更改级别。

ECB: (1) 请参阅事件控制位 (*event control bit*)。 (2) 事件控制块。 (3) 电子密码本

ECC: (1) 请参阅错误校验和更正 (*error-checking and correction*)。 (2) 错误更正代码。

ECM: 实体协调管理。

ECMA: 欧洲计算机制造商联合会。

EGA: 增强型图形适配器。

EGP: 外部网关协议。该机制允许一个自治系统的外部网关与另一个自治系统的外部网关共享路由信息。

EIA: 电子工业联合会。

EIA-232D: EIA 接口标准, 定义了接口线路的物理、电子和功能特性, 该接口线路连接了通讯设备和相关联的工作站。它使用了 25 针的连接器和非对称线路电压。

EIA-422A: EIA 接口标准, 定义了接口线路的物理、电子和功能特性, 该接口线路将计算机连接到通信设备。它使用对称线路电压来降低噪声, 有更远距离的传输能力。系统单元通过 EIA 接口定义的 40 针组来使用收发针。

else 语句 (else clause) : **if** 语句的一部分, 包括关键字 **else** 后跟一个语句。当 **if** 条件的值求出是 0 (假) 时才执行 **else** 子句。

em: 在特定类型大小里, 一个 **em** 有与特定大小相同数量的点。请参阅 *en*。

EMC: 请参阅电磁兼容性 (*electromagnetic compatibility*)。

en: 在 **troff** 命令中, **en** 是半个 **em** 点的大小。在 **nroff** 命令中, **em** 和 **en** 有相同的大小。请参阅 *em*。

Enter 键 (enter) ; 输入: (1) 按 Enter 键, 发送信息给计算机。 (2) 在行中放置一条从终端传递到计算机的消息。

ESCD 控制台 (ESCD console) : 用于在 ESCD 执行连接性任务的 ESCD 输入 / 输出设备。

ESN: 请参阅项序列号 (*entry sequence number*)。

EXEC: 远程命令执行协议 (*Remote Command Execution Protocol*)。

F

F: 华氏 (温度)。

FCC: 联邦通讯委员会。

FCFS: 请参阅先来先服务 (*first-come-first-served*)。

FD: 全双工。请参阅双工 (*duplex*)。

FDDI: 光纤分布式数据接口 (*Fiber-optic Distributed Data Interface*)。一个 100 兆位 / 秒光笔 LAN 接口。一个 ANSI 标准, 用于高速、100 兆位 / 秒、通用的计算机互相连接的网络、网络和以双环配置的使用光纤电缆的外围设备。FDDI 可以最多连接 500 站点, 最大链路到链路距离是 2 千米, 总 LAN 圆周总长是 100 千米。

FDX: 全双工。请参阅双工 (*duplex*)。

FIFO: 请参阅先进先出 (*first-in-first-out*)。

filetab: 一个内核参数, 用于建立可以同时打开的文件的最大个数。

FLIH: 请参阅一级别中断处理程序 (*first-level interrupt handler*)。

FM: 请参阅频率调制 (*frequency modulation*)。

FM 头 (FM Header): 功能管理头部 (Function Management Header)。

for 语句 (for statement): 在编程语言中, 一条对一或多个变量分配值集的每一个值执行一或多条语句的语句。

FORTRAN (FORMula TRANslation): 一种高级编程语言, 主要用于科学、工程与数学应用程序。

FP: 请参阅 *FPU*。

FPU (FP): 浮点单元。

FTP: 文件传送程序。另见文件传送协议 (*File Transfer Protocol*)。

FX: 请参阅 *FXU*。

FXU (FX): 定点单元。

G

GB: 请参阅吉字节 (*gigabyte*)。

GC: 请参阅图形上下文 (*graphics context*)。

GC 高速缓存 (GC caching): 在“增强的 X-Windows”中, 允许将独立的更改请求合并到一个协议请求中。

GContext: 请参阅图形上下文 (*graphics context*)。

GCP: 图形控制处理器 (Graphics Control Processor)。

GCR: 组码记录 (Group Code Recording), 一种磁带记录格式, 其密度为 6250 bpi。

GDA: 全局目录代理 (Global Directory Agent)。

GDLC: 请参阅类属数据链路控制 (*generic data link control*)。

GGP: 请参阅网关至网关协议 (*Gateway-to-Gateway Protocol*)。

GID: 请参阅组号 (*group number*)。

GL: 图形库 (Graphics Library)。

GLB: 请参阅全局位置代理 (*Global Location Broker*)。

glbd: 全局位置代理守护程序 (The Global Location Broker daemon)。

goto 语句 (goto statement): 在编程语言中, 将控制权转移给程序中另一点的语句。

Gouraud 阴影法: 基于多边形顶点的亮度平滑多边形阴影的方法。沿着每条边均匀插入色彩, 然后沿着每条扫描线均匀插入边缘值。对于逼真的阴影, 色彩应经过伽玛校正。

GPR: 请参阅通用寄存器 (*general purpose register*)。

guard 表达式 (guard expression): 置于布尔表达式开始处的表达式, 用来检查能否执行其它操作。

H

HCON: 请参阅 3270 Host Connection Program 2.1 和 1.3.3 (3270 Host Connection Program 2.1 and 1.3.3)。

HCONMRI: 3270 Host Connection Program 2.1 和 1.3.3 消息目录 (3270 Host Connection Program 2.1 and 1.3.3 Message Catalog)。

HCONuser: 已授予使用 3270 Host Connection Program 2.1 和 1.3.3 (HCON) 必需的特定权限的用户。另见 3270 Host Connection Program 2.1 和 1.3.3 (3270 Host Connection Program 2.1 and 1.3.3)。

HD: 请参阅半双工 (*half duplex*)。

HDLC: 请参阅高级数据链路控制 (*high-level data link control*)。

HDR: 请参阅首标 (*header label*)。

HDX: 请参阅半双工 (*half duplex*)。

HELLO: DCN 本地网络协议。设计用于自主系统的内部网关协议。

HIA: 请参阅 *System/370* 主机接口适配器 (*System/370 Host Interface Adapter*)。

hog 因子 (hog factor): 在系统记帐中, 对每个命令运行多少次、占用了多少处理器时间和内存以及使用的密集程度的分析。

Hollerith 常数 (Hollerith constant): 指 FORTRAN 中能在处理器中表示的任何字符串, 以 *nH* 开头, 其中 *n* 是该字符串中的字符数。

HOME: 请参阅 *\$HOME*。

hook 标识 (hook ID): 指定给特定跟踪点的唯一号码。所有跟踪项在其头部包含有起始跟踪点的 hook 标识。一个 hook 标识是一个 12 位的值。对于用户程序, hook 标识可

能是从 0x010 到 0x0FF 的值。Hook 标识在 `/usr/include/sys/trchid.h` 文件中定义。

HTML: 超文本标记语言 (HyperText Markup Language, HTML) 是 Web 浏览器用来解释和显示文档的标记语言。

Hz: 请参阅赫兹 (hertz)。

H&J: 在 C.A.T 术语中, (通常指自动) 的连字符连接和对齐过程。

I

I 和 D 混合高速缓存: 一同时包含指令和数据的高速缓存, 仅以高速缓存行标记区分。

IAR: 请参阅指令地址寄存器 (instruction address register)。

IC: 请参阅 ICU。

ICCC: 客户机间通信约定 (Inter-Client Communications Convention)。是一本手册的标准缩写, 该手册包含有使用“增强的 X-Windows”系统和 AIXwindows 工具箱的客户机应用程序的详细的准则集。另见 ICCCM。

ICCCM: 客户机间通信约定手册。另见 ICCCM。

ICMP: 请参阅因特网控制报文协议 (Internet Control Message Protocol)。

ICU (IC): 指令高速缓存单元。

ID: 标识 (Identification)。请参阅标识符 (identifier)。

IDP: 网际数据报协议 (Internet Datagram Protocol)。一种简单而不可靠的数据报协议, 用于支持“网际协议”系列的 SOCK_DGRAM 抽象。

IEEE: 电气和电子工程师协会 (Institute of Electrical and Electronics Engineers)。

IEEE 754: 二进制浮点标准 (Binary Floating Point Standard)。

IEEE 802.3: 以太网 LAN 规范。

IEEE 802.5: 令牌环 LAN 规范。

IETF: 因特网工程任务组 (Internet Engineering Task Force)。

IF 表达式 (IF expression): 一个过程中测试某个条件的表达式。由该过程执行的操作取决于测试的结果。

if 语句 (if statement): (1) C 语言条件语句, 包含关键字 **if**, 后面是一个用括号括起来的表达式 (条件)、一条语句 (操作) 及一个可选的 **else** 子句 (备选操作)。(2) 条件语句, 指定待测试条件和当条件满足时要采取的操作。

IMPL: 初始微程序装入 (Initial microprogram load)。

in: 请参阅英寸 (inch)。

include 语句 (include statement): 一种计算机语言的预处理器语句, 它指示处理器检索包含指令的特定文件和程序可能需要的数据。另见包含文件 (include file)。

inodetab: 用来在内存中为存储所有活动文件的 i-node 副本建立一个表的内核参数。

InputOutput 窗口 (InputOutput window): 在图形环境中, 一类用于输入和输出的不透明的窗口。InputOutput 窗口可以将 InputOutput 窗口和 InputOnly 窗口看作下级窗口。

int 说明符 (int specifier): 在 C 语言中, 字 **int**、**short**、**short int**、**long**、**long int**、**unsigned**、**unsigned int**、**unsigned short**、**unsigned short int**、**unsigned long** 或 **unsigned long int** 之一, 用以描述变量代表的的数据类型。

intern: 定义原子的过程。

IO: 请参阅 IOU。

IOCC: 请参阅输入/输出通道控制器 (input/output channel controller)。

IOCDS: I/O 配置数据集。

IOT 故障 (IOT fault): 异常终止进程的信号 (SIGIOT)。

IOU (IO): 输入/输出单元。

IP: 请参阅网际协议 (Internet Protocol)。

IP 地址 (IP address): 网际协议所采用的网络地址。是一个 32 位整数, 通常用点分十进制书写, 其中每个连续的八进制被转换成整数和由小数点与其它部分分开。

IP 套接字 (IP socket): 连接网际协议 (IP) 地址的端口。

IPC: 请参阅进程间通信 (inter-process communication)。

IPL: 请参阅初始程序装入 (initial program load)。

IPL 设备 (IPL device): 初始程序装入设备。

ips: 英尺/秒, 磁带机速度和性能的度量。

ISC: 请参阅初始序列控制器 (*initial sequence controller*)。

ISO: 请参阅国际标准化组织 (*International Standards Organization*)。

ISO 7776: LAPB-兼容 DTE 数据链路过程的 ISO 描述。

ISO 8208: 请参阅 X.25。

ITE: 在 CDE 中的内部终端仿真器。ITE 允许采用位图显示作为终端 (通过登录屏幕的命令行方式)。

I-field: 信息字段。

I-field 字节 (I-field byte): 某个发送或接收序列数据帧的信息字段中的数据。

i-list: 在基本操作系统的文件系统中, 块 2 到 n 组成 i-list, 其包含使文件与数据块或磁盘联系起来的各种结构 (i-node)。i-list 的大小取决于所安装文件系统的大小。另见 i-node 和 超级块 (*superblock*)。

i-node 号 (i-node number): 一个指定文件系统中的特定的 i-node 文件的数。请参阅 i-node。

I/O: 请参阅输入 / 输出 (*input/output*)。

I/O 错误 (I/O error): 输入或输出设备数据处理中的错误。

I/O 流库 (I/O Stream Library): 一个 C++ 类库, 提供处理多种输入和输出变量的设施。

I/O 配置 (I/O configuration): 通道路径、控制单元和连接到处理器单元的 I/O 设备的集合。

I/O 通道 (I/O channel): 请参阅输入 / 输出通道 (*input/output channel*)。

J

Java 语言 (Java Language): 一种基于 C++ 语言, 由 Sun Microsystems 开发的编程语言。Java 开发为包含网际数据操作的方法。Java 应用程序可以一次编写, 并可运行在任一机器上, 此机器必须拥有作为操作系统一部分的 Java 虚拟机。

JIS: 请参阅日本工业标准 (*Japanese Industry Standard*)。

K

kanji: 一种由符号组成的图形字符集, 这些符号用于日本表意符字母中。

KB: 千字节

Kb: 千位

kg: 请参阅千克 (*kilogram*)。

kprocs: 一个建立内核能同时运行的进程的最大数量的内核参数。

KSR: 请参阅键盘的发送 / 接收方式 (*keyboard send-receive mode*)。

KTS: 按键式电话系统。一个要求手动选择外线的私有电话系统。

kVA: 千伏安, 或者 1000 伏安。

L

L1 高速缓存 (L1 cache): 当存储器引用发生时, 第一个要访问的高速缓存。

L2 高速缓存 (L2 cache): 在某些的 RS/6000 型号上, 如果 L1 高速缓存查找导致高速缓存未命中时访问的高速缓存。通常的, L2 高速缓存的容量比 L1 高速缓存大, 速度要慢, 但是比 RAM 要快。

LAN: 请参阅局域网 (*local area network*)。

LAP: 请参阅链路访问过程 (*link-access procedures*)。

LAPB: 链路访问过程平衡 (*Link-access procedure balanced*)。另见链路访问过程 (*link-access procedures*)。

lb: 请参阅磅 (*pound*)。

LC: 请参阅链路控制 (*link control*)。

LCN: 请参阅逻辑通道号 (*logical channel number*)。

leaf delta: 对保存在源代码控制系统 (SCCS) 文件中的源代码所作的一系列更改。

LED: 请参阅发光二极管 (*light-emitting diodes*)。

License Use Management: 一个基于网络许可系统的 Gradient 技术版本 2.0.1 (版本 1.1.2a) 的运行时许可管理应用程序。该系统允许软件供应商将他们的软件与遵守机制捆绑在一起。在跟踪许可证的使用上, License Use Management 允许客户容易地遵守他们的软件许可证协议。

License Use Management 测试产品 (Test Product): 该产品通过使用 **ls_tv** 工具来验证许可证服务器运行是否正常。

LIS: 逻辑 IP 子网。一个 LIS 由一定数量的 ATM 站组成。LIS 类似于传统的 LAN 段。

LISP: 设计为表处理的一个广泛用来处理人工智能问题的编程语言。

LISP 代码 (LISP code): 用 LISP 编程语言写的程序指令。

LISP 方式 (LISP mode): 对于文本编辑器, 用在 LISP 编程语言中的字符方式, 如: ((左圆括号) 和]] (双右中括号), 作为符号来创建, 而不是作为编辑器命令来创建。

LLB: 请参阅本地位置代理 (*Local Location Broker*)。

llbd: 本地位置代理守护程序。

LLC: 请参阅逻辑链接控制 (*logical link control*)。

LMS: 线路监视器系统 (*Line monitoring system*)。

LNS: 请参阅 LU 网络服务组件 (*LU network services component*)。

long: (1) 在 ODM 中, 用于描述一个变量作为一个带符号的 4 字节号码的结束描述符类型。另见结束描述符 (*terminal descriptor*)。 (2) 一个带符号的 4 字节号码。

LP: 请参阅许可程序 (*licensed program*)。

LPFK: 请参阅发光可编程功能键盘 (*lighted programmable function keyboard*)。

LPM: 行 / 秒。打印机在一分钟内打印的行数。

LPR: 行打印机服务器。

LQ: 铅字质量。

LRU: 少量最近使用的。

LSB: 最不重要的字节。

lsb: 最不重要的位。

ls_admin: 在 License Use Management 中, 用来修改许可证服务器数据库的软件程序, 调用命令 **ls_admin**, 此命令位于 **/usr/lib/netls/bin** 目录。

ls_rpt: 在 License Use Management 中, 报告许可证服务器事件历史的软件程序, 调用命令 **ls_rpt**, 此命令位于 **/usr/lib/netls/bin** 目录。

ls_stat: 在 License Use Management 中, 报告许可证状态的软件程序, 调用命令 **ls_stat** 此命令位于 **/usr/lib/netls/bin** 目录。

ls_tv: 在 License Use Management 中, 网络许可证服务器守护程序的测试和验证工具, 调用命令 **ls_tv**, 此命令位于 **/usr/lib/netls/bin** 目录。

LU: 请参阅逻辑单元 (*logical unit*)。

LU 网络服务组件 (LU network services component, LNS): 开始和结束 LU-LU 会话响应于资源管理器的请求和远程 LU 的请求。它还激活和取消激活 CP-LU 会话。

LU, 独立 (LU, independent): 一个逻辑单元, 可以启动与另一个逻辑单元进行对话。

LU, 依赖 (LU, dependent): 一个逻辑单元, 必须等到主机系统启动对话后才能启动对话。

LU1: 请参阅逻辑单元类型 1 (*Logical Unit Type 1*)。

LU2: 请参阅逻辑单元类型 2 (*Logical Unit Type 2*)。

LU3: 请参阅逻辑单元类型 3 (*Logical Unit Type 3*)。

LU6.2: 请参阅逻辑单元类型 6.2 (*Logical Unit Type 6.2*)。

LU-LU 会话 (LU-LU session): 在 SNA Server 中, 一种在两个相同类型逻辑单元 (LUs) 之间的会话, 此对话支持两个最终用户通信, 也支持一个最终用户和一个 LU 服务组件之间的会话。

LV: 请参阅逻辑卷 (*logical volume*)。

lvalue: (1) 一个表达式, 表示一个对数据对象可被检查和更改。 (2) 表达式的左边部分。

LVM: 请参阅逻辑卷管理器 (*Logical Volume Manager*)。

M

m: 请参阅仪表 (*meter*)。

MAC: (1) 请参阅介质访问控制 (*medium access control*)。 (2) 强制访问控制。

main: 在 FORTRAN 语言中, 如果编程器没有提供名称, 它就是缺省的主程序名。

MAP: 请参阅维护分析过程 (*maintenance analysis procedure*)。

MB: 兆字节。

Mb: 兆位。

mbuf: mbuf 管理工具为操作系统上通信软件不同层提供的一个很小的 (256 字节) 缓冲区。

MC: 请参阅 MCU。

MCU (MC): 内存控制单元。

MHz: 请参阅兆赫兹 (*megahertz*)。

MIB 变量 (MIB Variable): 定义在管理信息库 (MIB) 中的一个受管对象。一个文本名称和相应的对象标识符, 一个句法, 一个访问方式, 一个状态, 和一个受管对象的句法描述定义了该受管对象。MIB 变量包含永久的管理信息, 通过定义的访问方式可访问该信息。

mil: 一个厚度度量单位, 1/1000 英寸。

mm: (1) 用于准备脚本的一个宏软件包, 它支持 **eqn** 和 **troff** 命令或 **neqn** 和 **nroff** 命令和功能注释、脚注、索引和表 (通过支持 **tbl** 命令)。(2) 毫米。

MOM: 请参阅监视器方式 (*monitor mode*)。

mond: 请参阅监视调度守护程序 (*monitor scheduling daemon*)。

ms: 一个用于原稿和论文准备的宏软件包, 它有自动脚注编号和目录的能力。

MSB: 最高有效字节 (Most significant byte)。

msb: 最高有效位 (Most significant bit)。

msqid: 请参阅消息队列标识 (*message queue ID*)。

MS-DOS: 微软磁盘操作系统 (Microsoft Disk Operating System)。

MTU: 请参阅最大传输单元 (*maximum transfer unit*)。

MVS: 多重虚拟存储器 (Multiple Virtual Storage)。

MVS/TSO: 用在 System/370 计算机上的一种操作系统。

mwm: AIXwindows窗口管理器。另见窗口管理器 (*window manager*)。

M-bit: 在 X.25 通信中, 一个表示在另一个数据包后有更多的数据 (当一个消息比一个包大时) 的数据包中的位。

M-byte: 请参阅兆字节 (*megabyte*)。

N

n: 请参阅 *en*。

NA: 不能应用的。

NaN: 请参阅非数 (*not-a-number*)。

NCA: 请参阅网络计算体系结构 (*Network Computing Architecture*)。

NCCF: 请参阅网络通信控制设施 (*Network Communications Control Facility*)。

NCK: 请参阅 网络计算内核 (*Network Computing Kernel*)。

NCS: 请参阅网络计算系统 (*Network Computing System*)。

NCS 单元 (NCS cell): 把 NCS 网络中一或多个机器编组的逻辑概念。属于备用单元的任何节点只能由那个单元的 License Use Management 服务器满足它们的许可证请求。在该单元外的节点不可以向另一单元的服务器发许可证请求。两种类型的单元, 缺省的或备用的, 用于提供两种访问 License Use Management 服务器的方式。

NDC: 请参阅规格化设备坐标 (*normalized device coordinates*)。

NETASCII: 一种八位 ASCII 码, 第一位设置为高, 用于错误检查。

NetBios: 基本输入/输出系统的网络版本。

NetLS: 请参阅 *License Use Management*。

netlsd: 用于调用网络许可证服务器守护程序的命令。

new: 在 C++ 中, 标识分配自由存储空间的操作符。**new** 操作符能被用于创建类对象。

NFS: 请参阅网络文件系统 (*Network File System*)。

NIC: 请参阅网络信息中心 (*Network Information Center*)。

NIM: 请参阅网络安装管理 (*Network Installation Management*)。

NIM 路由 (NIM routing): 定义 NIM 环境中哪些网络可以相互通信以及他们使用哪些网关来使通信变得简便的信息。NIM 路由用来表示总体网络环境中 LAN 的 TCP/IP 路由。

NIS: 网络信息服务。允许您通过您的网络维护一致的配置文件分布式数据库。

NL: 请参阅换行符 (*new-line character*)。

nodelock 文件 (nodelock file): 用户节点的文本文件 (而不是服务器许可证的节点), 在它此处添加节点锁定许可证。**nodelock** 文件位于 */usr/lib/netls/conf* 目录。

nroff: 为驱动 37 型电传打字工作站设计的排版实用程序。它现在驱动大量的退格和非退格 tty 型行式打印机和仿真 tty 打印机。

NRZ: 请参阅非归零 (*non-return-to-zero*)。

NS: 网络系统。同样地, 网络服务系统。

NSA: 下一站寻址。

NSI: 名称服务接口。

NTN: 请参阅国家终端号 (*national terminal number*)。

NTP: 请参阅网络计时协议 (*Network Time Protocol*)。

NTSC: 美国电视标准委员会定义的国家电视工业广播标准, 使用在美国、加拿大、日本和其它国家。美国广播标准的视频显示和定时格式。大多数视频磁带录制机录制和回放 NTSC 信号。需要专用的硬件把 RGB 监视器的输出转换为 NTSC 信号。另见 PAL 和 SECAM。

NTSC 信号 (NTSC signal): 国家电视标准委员会定义的信号标准。也叫作合成视频 (*composite video*)。

NTU: 请参阅网络终止单元 (*network terminating unit*)。

NUA: 请参阅网络用户 (*network user*) 和地址 (*address*)。

NUI: 请参阅网络用户识别 (*network user identification*)。

NUL: 请参阅空字符 (*null character*)。

NUL 字符 (NUL character): 在 XPG4 系统接口中, 所有位设置为零的字符。

NULL: 在 C 语言中, 保证不指向数据对象的指针。

null (空): 空、没有值、什么也不包含。

NURBS (Non-Uniform Rational B-spline): 可以被非一致有理 B 样条曲线和分段线形曲线修饰的参数表面。另见修饰循环 (*trimming loops*)。

NVRAM: 请参阅非可变随机存取存储器 (*nonvolatile random access memory*)。

O

O: 组织。

OCS: 卡板载序列器。

OCT: 请参阅对象类表 (*Object Class Table*)。

ODM: 请参阅对象数据管理器 (*Object Data Manager*)。

OM: 请参阅 XOM。

ORed: 已执行逻辑 OR 运算。

OS: 请参阅操作系统 (*operating system*)。

OSF: 开放软件基金会。

OSI: 请参阅开放系统互连 (*Open Systems Interconnection*)。

OSPF: 开放最短路径优先协议。

OSS: OSI 会话服务。

OS/2: 操作系统 / 2。

OU: 组织单元。

P

PAL: (1) 可编程阵列逻辑 (*Programmable array logic*)。
(2) 在欧洲和其它某些国家或地区使用的国家电视工业广播标准。另见 NTSC 和 SECAM。

PAL 信号 (PAL signal): 相位模拟锁信号, 也称为混合视频。混合视频的欧洲标准。

Pascal: 一种高级的、通用的编程语言。用 Pascal 写的程序是分块结构的, 由独立的例程组成。

PBX: 专用分支交换机。执行自动选择外线的专用电话系统。

PC: 个人计算机。

PCI: 程序控制中断。

PCM: 物理连接管理。

PCS: 见可编程字符集 (*programmable character set*)。

PDN: 见公用数据网 (*public data network*)。

PE: 编码的短语, 密度为 1600 bpi 的磁带记录格式。

PEP: 包交换协议。用户级别库实现的数据报服务, 使用 IDP 数据报套接字。

PEX: 支持三维空间图形的协议。

PEXlib: PEX 协议的程序员接口。

PFM: 见程序故障管理 (*program fault management*)。

PHIGS: 见程序分级交互图形系统 (*Programmers' Hierarchical Interactive Graphics System*)。

PID: 见进程标识 (*process ID*)。

PIO: 请参阅可编程输入 / 输出操作 (*programmable input/output operation*)。

PIP: 请参阅程序初始化参数 (*Program Initialization Parameters*)。

PList: 一种指针数组, 它带有以各种方法用于添加和除去元素的操作套件。

PMF: 参数管理框架。

PMP: 请参阅预防性维护软件包 (*Preventive Maintenance Package*)。

PMR: 请参阅问题管理记录 (*Problem Management Record*)。

POR: 请参阅加电复位 (*power-on reset*)。

POSIX: 请参阅计算机环境的可移植操作系统接口 (*Portable Operating System Interface For Computer Environments*)。

POST: 见加电自测 (*power-on self-test*)。

PostScript: 用于驱动文本和图形输出的图形语言。Adobe Systems, Inc. 的商标。

PPA (连接的物理点) (PPA, physical point of attachment): 一个点, 在此处系统将自己连接到物理通信介质。

PPA 标识 (PPA identifier): 通信在其上发生的特定物理介质的标识。

PROM: 可编程只读存储器。

P(R): 在 X.25 通信中, 包接收序列号。

PSAP: 请参阅表示服务存取点 (*Presentation Service Access Point*)。

PSDN (包交换数据网络) (PSDN, packet-switching data network): PSDN 是一组互相连接的交换节点, 该节点使订户能使用标准协议和包交换技术交换数据。这样的网络在由许多网络用户共享的电路上传递分到包中的消息。一根进入办公室的物理线路能处理许多并发连接。

PSN: 请参阅公用交换网络 (*public switched network*)。

PSTN: 请参阅公用交换电话网络 (*public switched telephone network*)。

P(S): 在 X.25 通信中, 发送序列号的信息包。

PTF: 请参阅程序临时性修订 (*program temporary fix*)。

PTN: 请参阅公用电话网络 (*public telephone network*)。

PTT: 公布、电报和电话权限。

PTY: 请参阅伪终端 (*pseudo terminal*)。

PU: 请参阅物理单元 (*physical unit*)。

PUT: 程序更新磁带。

PUT 2.0 或 PUT 2.1 (PUT 2.0 or PUT 2.1): 在 SNA 中, 有有限寻址和路径控制路由能力的一种外围节点。PUT 2.0 节点根据子区节点 (PUT 4 和 PUT 5) 以在其本地寻址和网络寻址间转换。PUT 2.0 不支持 LU6.2 的全部能力; PUT 2.1 支持。SNA Server 仅操作 PUT 2.0 或 PUT 2.1 外围节点。

PUT 4 或 PUT 5 (PUT 4 or PUT 5): 一种子区节点, 它提供网络范围寻址和子区内的控制数据流 (子区节点和所有连接到它的外围节点)。PUT 4 不包含 SSCP 组件; PUT 5 包含。SNA Server 不能执行 PUT 4 或 PUT 5 子区节点的功能。

PVC: 请参阅永久虚拟电路 (*permanent virtual circuit*)。

Q

Q 位 (Q-bit): 在 X.25 通信中, 数据包中的一种位, 可通过发送 DTE 以用某种对接收 DTE 有意义的方法限定用户数据来设置它。

qdaemon: 一种守护进程, 它维护一系列未完成作业, 并在适当的时间将它们发送到指定的设备。

QIC: 四分之一英寸盒式磁带。

QID: 队列标识。

QLLC: 见限定逻辑链路控制 (*qualified logical link control*)。

QWERTY 键盘 (QWERTY keyboard): 有标准键盘布局的键盘。该术语派生于字母键的第一行的前六个键的序列。另见 AZERTY 键盘 (*AZERTY keyboard*)。

R

RAM: 随机存取内存。特别是用于系统内存的内存。有时该内存指主存储器。

RC: 路由控制。

RCM: (1) 请参阅实时控制微码 (*real-time control microcode*)。 (2) 输出上下文管理器。

RCS: 请参阅修订控制系统 (*Revision Control System*)。

RD: 接收与 EIA-232 协议一起使用的的数据。

REC: 请参阅锁定服务 (*Lock Service*)。

RFC: 请求注释。

RFI: 请参阅无线电频率干扰 (*radio frequency interference*)。

RGB 方式 (RGB mode): 一种硬件配置, 它允许帧缓冲区中存储的值被解释为压缩的 RGB 值。在帧缓冲区中找到的值被直接传递到显示监视器的红、绿和蓝枪中。这些值不首先通过色图传递。(但是, 每个颜色都独立地通过伽玛斜坡发送, 以对其亮度进行最后校正。)另见色彩图方式 (*color map mode*)。

RGB 信号 (RGB signal): 红 / 绿 / 蓝信号。

RGB 颜色 (RGB color): 一种颜色值刻度, 它由主值 (红、绿和蓝) 组成。

RGB 值 (RGB value): 组成颜色的红、绿和蓝亮度的集, 即颜色的 RGB 值。

RGBA 值 (RGBA value): 在 GL 中, 红、绿、蓝和组成颜色的 alpha 亮度的集合。Alpha 值仅在有 alpha 位平面的机器上可用。

RIC: 实时接口协处理器。

rid: 请参阅资源标识 (*resource identifier*)。

RIP: 请参阅路由信息协议 (*Routing Information Protocol*)。

RISC: 请参阅减少指令集计算机 (*Reduced Instruction Set Computer*)。

RJE: 请参阅远程作业项 (*remote job entry*)。

RMT: 环形管理。

RNR 包 (RNR packet): 在 X.25 通信中, 一种接收未就绪包。对照 RR 包 (*RR packet*)。

RNR 帧 (RNR frame): 在 X.25 通信中, 一种接收未就绪帧。对照 RR 帧 (*RR frame*)。

ROM: 只读存储器。

Roman 字体 (Roman font): 普通类型字体。在许多铅字样中, 这是缺省字体, 管理大多数文本。它最常用于关闭斜体字或粗体字。

root 用户权限 (root user authority): 无限制地访问和修改操作系统的任何部分的能力, 通常与管理系统的用户关联。另见 root 用户 (*root user*)。

root 用户 (root user): 能无限制地访问和修改操作系统的任何部分的人, 通常是管理系统的用户。另见 root 用户权限 (*root user authority*)。

ROS: 远程操作服务。

ROSE: 远程操作服务元素。

RPC: 请参阅远程过程调用 (*remote procedure call*)。

rpm: 每分钟转数。

RPN: 请参阅实页号 (*real page number*)。

RPOA: 请参阅可识别专用操作代理 (*recognized private operating agency*)。

RPQ: 报价请求。

RR: (1) 资源记录 (DNS)。(2) 取整 (调度)。

RR 包 (RR packet): 在 X.25 通信中, 由 DTE 或 DCE 使用的一种包, 表明它已对接收窗口内的数据包准备就绪。对照 RNR 包 (*RNR packet*)。

RR 帧 (RR frame): 在 X.25 通信中, 一种接收就绪帧。对照 RNR 帧 (*RNR frame*)。

RSi: 请参阅远程统计信息接口 (*Remote Statistics Interface*)。

RS-232C: 请参阅 EIA-232D。

RS-422A: 请参阅 EIA-422A。

RTI: 响应类型指示符。

RTPN: 请参阅远程事务处理程序名 (*remote transaction program name*)。

RTS: 与 EIA-232 协议一起使用的准备发送。

RW 锁定 (RW lock): 读共享 / 写互斥锁定 (*read shared/write exclusive lock*) 的缩写。任何数目的线程可同步保留用于读的锁, 但是如果一个线程保留用于写的锁, 则所有其它线程被拒绝读或写锁定的资源。

S

S 接口 (S-connector): 使用超级视频的接口类型。类似于 Y/C 接口。它是一个 4-pin-min 型号尺寸的接口。

S 视频 (S-video): 超级视频。用来改进视频图像的质量, 它是一种在磁带上录制或在监视器上显示之前维持独立的 Y/C 信号的技术。

SABM: 请参阅设置异步平衡方式 (*set asynchronous balanced mode*)。

SAK: 请参阅安全注意密钥 (*secure attention key*)。

SAP: 请参阅服务访问点 (*service access point*)。

SAS: 单一连接站。通过布线集线器仅连接至主环或以背对背布线配置连接至另一个 SAS 的站。

SCCS: 请参阅源代码控制系统 (*Source Code Control System*) 和修订控制系统 (*Revision Control System*)。

SCCS 标识 (SID): 在 SCCS 中指定给每个程序的版本的号码。

SCCS 增量 (SCCS delta): 对 SCCS 文件所做的一组更改。创建新增量仅保存所做的更改。

SCSI 适配器 (SCSI Adapter): 请参阅小型计算机系统接口适配器 (*Small Computer Systems Interface Adapter*)。

sdb: 请参阅符号调试器 (*symbolic debugger*)。

SDLC: 请参阅同步数据链路控制 (*synchronous data link control*)。

SDLC 辅站 (SDLC secondary station): 响应另一个站 (主站) 的请求的站, 它对数据链路操作有极少的控制权。

SDLC 主站 (primary station): 负责数据链路的站。它向辅站发布命令。

SDT: 请参阅静态调试器陷阱 (*static debugger trap*)。

SECAM: 用于法国、苏联 (USSR) 和其它一些国家或地区的国家电视工业广播标准。另见 NTSC 和 PAL。

semid: 请参阅信号标识 (*semaphore ID*)。

set 标志 (set flags): 可以通过 shell 的 set 命令而生效的标志。

setgid: 请参阅设置组标识方式位 (*set-group-ID mode bit*)。

setuid: 请参阅设置用户标识方式位 (*set-user-ID mode bit*)。

shell: (1) 用户和计算机操作系统之间的一种软件界面。shell 程序解释命令和用户在设备上 (比如键盘、指示器和触敏屏幕) 的交互作用, 并且将它们传达至操作系统。(2) 允许内核程序在不同操作系统环境下运行的软件。(3) 向用户提供内核界面的命令解释器。另见 shell 程序 (*shell program*) 和命令解释器 (*command interpreter*)。与界面 (*interface*) 同义。(4) shell 是一种命令解释器, 它在用户和操作系统之间担当界面作用。shell 可以包含另一个嵌套在其中的 shell, 在这种情况下, 外部的 shell 是父 shell, 内部的 shell 是子 shell。(5) 在 AIXwindows 中, Shell 小窗口是最高级别的小窗口, 它们是内部的且不能被例示, 但是它们向窗口管理器提供了必要的界面。另见 shell 小窗口 (*shell widget*)。

shell 变量 (shell variables): shell 的用于向常量名称指定变量值的工具。

shell 步骤 (shell procedure): 组合在一个文件中的一系列命令, 当文件运行或当文件被指定为 sh 命令的一个值时, 这些命令执行某个特定的功能。与 shell 脚本 (*shell script*) 同义。

shell 程序 (shell program): 接受并解释操作系统命令的程序。shell 的同义词。

shell 脚本 (shell script): shell 步骤 (*shell procedure*) 的同义词。

shell 控制命令 (shell control command): 使用户能够将控制传递给 shell 步骤各个部分或控制一个程序步骤如何结束的命令。

shell 框 (shell box): 一种几何学管理技术, 其中某类型的边界框只能有一个与 shell 相同大小的字框。

shell 命令名称 (shell command names): 操作系统命令。

shell 提示符 (shell prompt): 命令行上的字符串, 它指明系统可以接受一个命令 (通常是 \$ 字符)。

shell 小窗口 (shell widget): 在“增强的 X-Windows”中, 它拥有与窗口管理器直接通信的最高级别的小窗口。这些小窗口没有父窗口。与 shell 同义。另见小窗口 (*widget*)。

shmid: 请参阅共享内存标识 (*shared memory ID*)。

short: ODM 中用于将变量定义为带符号 2 字节数的结束描述符。参阅结束描述符 (*terminal descriptor*)。

SiCounter: Performance Toolbox 中持续增加的值。工具显示各次观察之间值的增量 (更改), 这些观察根据已逝时间划分, 每秒表示一个速率。

SID: SCCS 标识。指定给增量的名称。

SiQuantity 值 (SiQuantity value): 在 Performance Toolbox 中表示一个级别, 比如已用内存或可用的磁盘空间。实际观察值由工具显示。

SJIS: 请参阅日文 Shift-JIS 编码 (*Shift-Japanese Industrial Standard*)。

SJN: 请参阅最短作业优先 (*shortest-job-next*)。

SLA: 串行链路适配器。另见 SOCC。

SLIH: 请参阅二级中断处理程序 (*second-level interrupt handler*)。

SLIP: 串行线路接口协议。当通过串行连接操作时 TCP/IP 使用的协议。

SMT: 系统管理界面工具 (System Management Interface Tool)。

SMP: 请参阅对称多处理器系统 (*symmetrical multiprocessor system*)。

SMP 安全 (SMP safe): 在程序中避免会在 SMP 环境中造成功能性问题的任何操作, 比如非串行访问共享数据。当单独使用此术语时, 通常指仅经历更正 SMP 环境中功能所需的最小更改的程序。

SMP 利用 (SMP exploiting): 向程序添加功能, 这些功能特别意在有效地利用 SMP 环境。描述为 SMP 利用的程序通常也假定为 SMP 安全和 SMP 有效。

SMP 有效 (SMP efficient): 在程序中避免会在 SMP 环境中造成功能性或性能问题的任何操作。描述为 SMP 有效的程序通常也假定为 SMP 安全。SMP 有效的程序通常经历附加的更改以最小化初期瓶颈。

SMT: 站管理。

SMTP: 请参阅简单邮件传送协议 (*Simple Mail Transfer Protocol*)。

SNA: 请参阅系统网络体系结构 (*System Network Architecture*)。

SNA 网络 (network): 用户应用程序网络中符合系统网络体系结构 (*System Network Architecture*, SNA) 中格式和协议的部分。它使最终用户间能进行可靠的数据传送, 并且提供了控制各种网络配置资源的协议。SNA 网络由网络可寻址单元 (NAU)、边界功能组件以及路径控制网络构成。

SNBU: 请参阅交换网备份 (*switched network backup*)。

SNMP: 简单网络管理协议 (*Simple Network Management Protocol*)。网络主机用来在网络管理中交换信息的协议。SNMP 网络管理基于在基于 TCP/IP 的网络应用程序中广泛使用的客户机服务器型号。

SNOBOL: 为字符串处理与模式匹配设计的编程语言。

SOCC: 串行光通道转换器 (*Serial Optical Channel Converter*)。220 Mbps/sec 光学的点对点链路。

SPC: 请参阅系统程序控制器 (*System Program Controller*)。

SPI: stub 编程接口 (*Stub programming interface*)。专用 RPC 运行时的接口, 它的例程不可用于应用程序代码。

Spmi: 请参阅系统性能评估界面 (*System Performance Measurement Interface*)。

SPOT: 请参阅共享产品对象树 (*Shared Product Object Tree*)。

SPP: 顺序分组协议 (*Sequence packet protocol*)。“Xerox 网络系统”中的主要传输层协议。它为应用程序提供了可靠的、控制流的、二路数据传输。它是一个字节流协议, 用于支持 SOCK_STREAM 抽象。SPP 协议使用标准“网络系统” (*Network System*, NS) 地址格式。

SRC: 系统资源控制器 (*System Resource Controller*)。

SRF: 标准记录格式 (*Standard record format*)。

SRN: 请参阅服务请求号 (*Service request number*)。

SRT: 请参阅结构规则表 (*Structure Rule Table*)。

SSCP: 请参阅系统服务控制点 (*system services control point*)。

STDERR: 请参阅标准错误 (*standard error*)。

STDIN: 请参阅标准输入 (*standard input*)。

STDOUT: 请参阅标准输出 (*standard output*)。

STREAMS: 支持网络服务和数据通信驱动程序开发的内核机制。它为内核中以及内核与用户级别间的字符输入和输出定义了接口标准。STREAMS 机制由整函数、实用程序例程、内核工具以及一组结构构成。

stub: (1) NCS 中, 在客户机和服务器之间传送远程过程调用和响应的程序模块。stub 执行 (数据) 组织、数据分解和数据格式转换。客户机和服务器都有 stub。编译程序从接口定义生成客户机和服务器 stub 代码。另见数据 (组织) (*marshal*)。 (2) 用作协议的扩展来为“增强的 X-Windows”生成协议请求的挂钩函数。挂钩例程 (*hooking routines*) 的同义词。 (3) 当定义接口时, 编译程序产生 RPC 调用。产生两组 stub, 客户机 stub 和服务器 stub。应用程序代码调用 stub, 然后 RPC 机制将此 stub 转化为调用远程机器上适当功能的调用。

subshell: 从现有 shell 程序启动的 shell 程序的实例。

subwidget: AIXwindows 和“增强的 X-Windows”中, 直接位于小窗口 / 小配件层次结构中更高层小窗口类下面的小窗口类。

SVC: 请参阅交换式虚拟电路 (*switched virtual circuit*) 和管理程序调用 (*Supervisor call*)。

swidge: 请参阅阴影小窗口 (*shadow widget*)。

switch 表达式 (switch expression) : (1) 位于关键字 **switch** 和 **switch** 语句主体之间的表达式。 (2) C 语言中, **switch** 语句的控制表达式。

SWVPD: 请参阅软件重要产品数据 (*Software Vital Product Data*)。

SYN: 请参阅同步字符 (*synchronization character*)。

System/370 主机接口适配器 (System/370 Host Interface Adapter, HIA) : 允许 POWERstation 或 POWERserver 连接至 5088 图形控制单元 (Graphics Control Unit) 的适配器。

T

tag-in: 在 “System 360/370 并行通道接口” 上将信号从控制单元传递到主机的控制线路总线。

tag-out: 在 “System 360/370 并行通道接口” 上将信号从主机传递到控制单元的控制线路总线。

tbl: 为 **nroff** 和 **troff** 命令格式化表的预处理器。

TCB: 请参阅可信计算基础 (*trusted computing base*)。

TCP: 请参阅传输控制协议 (*Transmission Control Protocol*)。

TCP/IP: 请参阅传输控制协议 / 网际协议 (*Transmission Control Protocol/Internet Protocol*)。

TCW: 转换控制单字。

TD: 用 EIA-232 协议的传输使用的数据。

Telnet: TCP/IP 中, 打开到系统的连接的协议。

tfptp: 请参阅普通文件传送协议 (*Trivial File Transfer Protocol*)。

this: C++ 关键字, 标识引用成员函数中类对象的特殊类型的指针。

throw 表达式 (throw expression) : 抛出异常的自变量。

TIC: 通道中的传送。

TIDL: 事务接口定义语言。

TIFF: 请参阅标记输入格式文件 (*Tagged Input Format File*)。

tile: (1) 一种像素图。 (2) 用像素图填充一个区域。 (3) 在二维中复制像素图。

TIMED: 时间同步服务器协议。用于将一台主机时间和其它主机时间同步的协议。

TLB: 请参阅转化后援缓冲区 (*translation lookaside buffer*)。

TLB 未命中 (TLB miss) : 当内存位置被引用, 且包含该内存位置的页在适当的转化后援缓冲区中没有条目 (指令或数据) 时发生的内存延迟。

TLI: 传输层界面。

TOD: 一天的时间。

TPN: 请参阅事务处理程序名称 (*transaction program name*)。

TRAN: 分布式交易服务 (*Distributed Transaction Service*)。

troff: 一种照相排版实用程序, 最初设计用来支持 “图形系统” 照相排版机器, 但现在能够支持各种照相排版。

try 块 (try block) : 一种 C++ 块操作, 其中已知异常被传递到处理程序。

TS: 请参阅传输服务 (*transmission services*)。

TSO: 时间共享选项 (*Time Sharing Option*)。

tty: 操作系统中任何使用 **termio** 标准终端设备接口的设备。 **tty** 设备通常以逐字符为基础执行输入和输出。

tty 设备 (tty device) : 请参阅 **tty**。

U

UA: 请参阅无编号确认 (*unnumbered acknowledge*)。

UART: 请参阅通用异步接收 / 传输 (*Universal Asynchronous Receive/Transmit*)。

ublock: 请参阅用户块 (*user block*)。

UCT: 请参阅通用协调时间 (*Universal Coordinated Time*)。

UDP: 请参阅用户数据报协议 (*User Datagram Protocol*)。

UDP/IP: 用户数据报协议 / 网际协议 (*User Datagram Protocol/Internet Protocol*)。

UFS: 请参阅 UNIX 文件系统 (*UNIX File System*)。

UI: 请参阅无编号信息帧 (*unnumbered information frame*)。

UID: 请参阅用户编号 (*user number*) 和 AIXwindows 用户界面定义 (*User Interface Definition*)。

UIL: 请参阅 AIXwindows 用户界面语言 (*User Interface Language*)。

umask: 文件方式创建屏蔽。当文件创建时自动设置的缺省许可。这些缺省可以通过在系统概要文件中包括进适当的 **umask** 命令来更改。

UNIX 文件系统 (UNIX File System, UFS): UNIX 文件树的一部分, 它被物理地包含在单个设备或磁盘分区中且可以单独装入、卸装和管理。

UNIX 至 UNIX 复制程序 (UNIX-to-UNIX Copy Program, UUCP): (1) 存在于大多数 UNIX 系统上的一组命令、程序和文件, 它们允许用户通过专用线路或电话线与另一个 UNIX 系统通信。另见基本网络实用程序 (*Basic Networking Utilities*)。 (2) 启动从一个或多个源到单个目标的文件复制的命令 (**uucp**)。

use 子句 (use clause): Ada 语言中, 实现出现在可见命名包中的声明的直接可见性的子句。

USOC-RJ11: 微型电话机插孔。

UTC: 请参阅通用协调时间 (*Universal Coordinated Time*)。

UTS: 请参阅更新时间戳记 (*Update Timestamp*)。

UUCP: 请参阅 UNIX 至 UNIX 复制程序 (*UNIX-to-UNIX Copy Program*)。

UUCP 登录标识 (UUCP login ID): 随“基本网络实用程序” (*Basic Networking Utilities, BNU*) 一起提供的登录名, 它具有完全访问所有 BNU 文件和目录的权限。另见基础网络实用程序 (*Basic Networking Utilities*)。

UUID: 请参阅通用唯一标识符 (*Universal Unique Identifier*)。

V

V: 伏特 (*Volt*)。

VC: 请参阅虚拟电路 (*virtual circuit*)。

vchar: ODM 中用于将变量定义为具有变量长度、null 结束字符串的结束描述符类型。另见结束描述符 (*terminal descriptor*)。

VCI: ATM 网络中的虚拟通道。

VDD: 请参阅虚拟设备驱动程序 (*virtual device driver*)。

vhandle: VEOS 中每个对象的标识符。vhandles 用于图像编程来标识 VEOS 中指定的软件对象。vhandle 可以比它标识的对象存在更长时间。当软件对象比如 swidget 遭到破坏, 该对象的 vhandle 保持已定义。

Virtual Storage Extended/Enterprise SystemsArchitecture (VSE/ESA): 在 ESA 环境运行的操作系统。ESA 是 System/370 体系结构的扩展, 且包含提供存取寄存器的高级可寻址功能部件。

Virtual Storage Extended/System Product (VSE/SP): 提供 VSE 操作系统支持的许可程序。

VLSI: 超大规模集成。

VLSI 电路 (VLSI circuit): 超大规模集成电路。

VLT: 请参阅视频查找表 (*video lookup table*)。

VM: 虚拟机 (*Virtual machine*)。

VMTP: 请参阅通用报文事务处理协议 (*Versatile Message Transaction Protocol*)。

VM/CMS: 用在 System/370 计算机上的一种操作系统。

VOL: 请参阅卷服务 (*Volume Service*)。

Vol ID: 请参阅卷标识 (*Volume ID*)。

volatile 寄存器 (volatile register): C 语言程序中, 其条目上的值不需要在调用的例程返回时保存的寄存器。

volatile 属性 (volatile attribute): 位于定义、声明或造型中的关键字 **volatile**。它使 C 语言编译程序将数据对象的值置于存储中, 并在每次引用此数据对象的时候重新装入该值。

VPD: 请参阅软件重要产品数据 (*Software Vital Product Data, SWVPD*)。

VPI: ATM 网络中的虚拟路径标识符。

VRAM: 视频随机存取存储器。

VSE: 请参阅虚拟存储扩展 (*Virtual Storage Extended*)。

VSE/ESA: 请参阅 *Virtual Storage Extended/Enterprise Systems Architecture*。

VSE/SP: 请参阅 *Virtual Storage Extended/System Product*。

VTAM: 请参阅虚拟远程通信访问方法 (*Virtual Telecommunication Access Method, VTAM*)。

VTL: 供应商技术逻辑 (*Vendor Technology Logic*)。

v-node: 虚拟 i-node。文件系统中表示文件的对象。不像 i-node, v-node 和文件系统中没有一对一通信; 多个 v-nodes 可以指单个的文件 (单个 i-node)。V-node 用于文件系统的上半层 (逻辑文件系统) 和文件系统实现 (比如日志文件系统或网络文件系统) 之间的通信。

V.24: V 系列中的第 24 本 CCITT 建议书, 它列出了 DTE 和 DCE 之间互换电路的定义。

V.35: V 系列中的第 35 本 CCITT 建议书, 它定义了使用 60-180 kHz 组带区电路的每秒 48 千比特的数据传输。

W

WAN: 请参阅广域网 (*wide area network*)。

WAN 链接 (WAN links): 分布在巨大地理区域上的计算机组之间的通信连接。调制解调器连接、T1 线路和卫星转播是一些常见示例。另见广域网 (*wide area network*)。

while 语句 (while statement): 一种 C 语言循环语句, 它包含关键字 **while**, 后跟一个在括号中的表达式 (条件) 和一个语句 (操作)。

with 子句 (with clause): 用于 Ada 编程, 请参阅编译单元 (*compilation unit*)。

X

X 服务器 (X Server): 请参阅服务器 (*server*)。

X 工具箱 (X Toolkit): 请参阅 AIXwindows 工具箱 (*Toolkit*)。

X 原子 (X Atom): 一个“增强的 X-Windows”的原子。另见原子 (*atom*)。

X 资源文件 (X resource file): AIXwindows 中包含所有在“小窗口属性编辑器”中指定为 Public 属性值的文件。最终用户可以编辑这样的文件。

X 资源 (X resource): 在 AIXwindows 中与资源 (*resource*) 同义。

X11 客户机 (X11 client): 可以和“增强的 X-Windows”的 VX11 兼容的应用程序。另见 X-Windows。

XCOFF: 请参阅扩展公共目标文件格式 (*extended common object file format*)。

xcomms: 为其它 X.25 命令 (**xmanage**、**xroute** 和 **xtalk** 命令) 提供菜单面板的命令。

XDR: 请参阅外部的数据表示 (*External Data Representation*)。

XDS: X/Open 目录服务 (*X/Open Directory Service*)。

XGSL: 请参阅图形支持库 (*Graphics Support Library*)。

XID: 请参阅交换标识 (*exchange identification*)。

XID 节点标识 (XID Node ID): 提供物理单元的节点标识的字段。此值就是当在 XID 操作过程中建立了第一个连接时与远程物理单元交换的标识。

Xlib: “增强的 X-Windows”中客户机用来连接视窗系统的 C 语言子例程库。

Xlib 调用 (Xlib call): 请参阅 Xlib。

xmanage: 使管理器可以管理 X.25 端口的命令。

Xmodem 协议 (Xmodem protocol): 其中的消息以 128 个字符 (并由控制字符包围) 的块的形式发送的通信协议。此协议还使用 **checksum** 程序来检查错误。xmodem 协议可以检测数据传输错误, 然后重新传输数据。

xmonitor: 使系统管理器能够监视 X.25 端口上活动的命令。

xmpeek: 允许您向任何一台主机询问其 **xmservd** 守护程序的程序。

XNS: Xerox 网络系统 (*Xerox Network Systems*)。Xerox Corporation 于二十世纪七十年代开发的网络体系结构。“XNS 网际协议”组与 TCP/IP 组相似。然而, 它使用不同的包格式和术语。

XOM: X/Open 对象管理 (*X/Open Object Management*)。

xroute: 使系统管理器可以管理 X.25 路由列表的命令。

Xt 本征 (Xt intrinsics): 请参阅本征 (*intrinsics*)。

xtalk: 使用户能够使用 X.25 与其它用户对话、交换消息和文件以及管理 X.25 地址列表的命令。

XTI: X/Open 传输界面 (*X/open Transport Interface*)。一种由“X/OPEN 传输接口”的“X/OPEN CAE 规范”指定的库实现, 它完全符合“X/OPEN 和 XPG4 公共应用程序环境” (CAE), 并定义了一组独立于任何特定传输提供程序的协议或其框架的传输层服务。

XTISO: 套接字上的 XTI。

XY 格式 (XY format): 一种组织成表示独立位平面 (这些位平面出现在“最重要至最不重要”位顺序中) 的一组位图的像素图的格式。另见 Z 格式 (*Z format*)。

X-Windows: 由 MIT 开发的网络透明的视窗系统。它是“增强的 X-Windows”的基础。请参阅 X11 客户机 (*X11 client*)。

X.121: X 系列的第 121 本 CCITT 建议书，它定义了网络用户地址（NUA）的约定。

X.21: 数据通信中，X 系列的第 21 本 CCITT 建议书，它为租用的数字和电路交换服务将数据终端设备的连接定义至 X.21 公共数据网络。X.25 通信中 X.21 只可在物理级别上使用。

X.21 bis: 一本 CCITT 建议书，它定义了为同步 V 系列调制解调器的接口而设计的 DTE 的公共数据网络上的使用。

X.25: 第 25 本 CCITT 建议书，它定义了数据终端设备和包交换数据网络间的接口。

X.25 端口 (X.25 port): /dev 目录中对应一条 X.25 链路的设备。另见 X.25 链路 (X.25 link)。

X.25 接口协处理器 / 2: 将系统设备连接至 X.25 包交换数据网络的可单独订购的适配器卡。与 X.25 适配器 (X.25 adapter) 同义。

X.25 链路 (X.25 link): 从 X.25 适配器到网络终止设备的 X.25 线路。在 X.25 API 中，链路等同于一个 X.25 端口。另见 X.25 端口 (X.25 port) 和数据链路 (data link)。

X.25 适配器 (X.25 adapter): X.25 接口协处理器 / 2 的同义词。

X.25 网络 (X.25 network): 提供符合 CCITT 采用的“建议书 X.25”的包交换数据传输的服务。

X.25 线路 (X.25 line): X.25 通信中，DTE 和 DCE 之间的物理链路以及订购的服务。另见通信线路 (communications line)。

X.28: 指定异步（起 / 止）设备与装拆器（PAD）之间的用户界面的 CCITT 建议书。

X.29: 指定 DTE 和远程包装拆器（PAD）之间的用户界面的 CCITT 建议书。

X.3: 通过包装拆器（PAD）将提供的服务指定给一台异步（起 / 止）设备的 CCITT 建议书。

Y

yacc 源文件 (yacc source file): 包含 yacc 编程语言代码的文件。

Y/C 信号 (Y/C signal): 一种黄色 / 色度信号。Y（黄色）表示亮度而 C 表示色度（颜色）。亮度包含图像详细信息和强度级别，而色度包含了图像的颜色信息。

Z

Z 格式 (Z format): 组织成在扫描线顺序中的一组像素的像素图格式。另见 XY 格式 (XY format)。

zombie 进程 (zombie process): 一个结束的进程，其条目保留在进程表中，但是用户和内核不分配这些条目。当进程发出 **exit** 子例程或发生以下情况时，进程变成 zombie 进程：它的父进程不运行 **wait** 子例程且没有将它的 **SIGCLD** 信号操作设置到 **SIG_IGN** 之中以表明它不打算等待它的子进程结束。

ZOOM 缩放因子 (zoom factor): 用于确定特定屏幕矩形扩大数量的乘数。 x ZOOM 缩放因子确定了 x 轴方向的扩大； y ZOOM 缩放因子确定了 y 轴方向的扩大。

特别字符

: 获得。例如，从文件抽取信息。

:

特别字符

\$HOME: 系统设置的一个环境变量，用来指定用户主目录。许多程序使用此变量来指定存储临时工作文件的目录。

/ 文件系统 (/ file system): 根文件系统；包含含有特定于机器的配置数据的文件。

/tmp 文件系统 (/tmp file system): 一个共享的文件存储位置。

/usr 文件系统 (/usr file system): 包含操作机器所必需的文件和程序。

/var 文件系统 (/var file system): 包含随每个客户机而不同的文件，如 spool 和邮件文件。

附录. 声明

本信息是为在美国提供的产品和服务编写的。

IBM 可能在其它国家或地区不提供本文档中讨论的产品、服务或功能特性。有关您当前所在区域的产品和服务的信息，请向您当地的 IBM 代表咨询。任何对 IBM 产品、程序或服务的引用并非意在明示或暗示只能使用 IBM 的产品、程序或服务。只要不侵犯 IBM 的知识产权，任何同等功能的产品、程序或服务，都可以代替 IBM 产品、程序或服务。但是，评估和验证任何非 IBM 产品、程序或服务，则由用户自行负责。

IBM 公司可能已拥有或正在申请与本文档内容有关的各项专利。提供本文档并未授予用户使用这些专利的任何许可证。您可以用书面方式将许可证查询寄往：

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

本条款不适用联合王国或任何这样的条款与当地法律不一致的国家或地区：国际商业机器公司以“按现状”的基础提供本出版物，不附有任何形式的（无论明示的，还是默示的）保证，包括（但不限于）对非侵权性、适销性和适用于某特定用途的默示保证。某些国家或地区在某些交易中不允许免除明示或默示的保证。因此本条款可能不适用于您。

本信息中可能包含技术方面不够准确的地方或印刷错误。此处的信息将定期更改；这些更改将编入本资料的新版本中。IBM 可以随时对本资料中描述的产品和 / 或程序进行改进和 / 或更改，而不另行通知。

本程序的被许可方如果要了解有关程序的信息以达到如下目的：(i) 允许在独立创建的程序和其它程序（包括本程序）之间进行信息交换，以及 (ii) 允许对已经交换的信息进行相互使用，请与下列地址联系：

IBM Corporation
Dept. LRAS/Bldg. 003
11400 Burnet Road
Austin, TX 78758-3498
U.S.A.

只要遵守适当的条件和条款，包括某些情形下的一定数量的付费，都可获得这方面的信息。

本资料中描述的许可程序及其所有可用的许可资料均由 IBM 依据 IBM 客户协议、IBM 国际程序许可证协议或任何同等协议中的条款提供。

有关双字节（DBCS）信息的许可证查询，请与您所在国家或地区的 IBM 知识产权部门联系，或用书面方式将查询寄往：

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

IBM 可以按它认为适当的任何方式来使用或分发您所提供的任何信息而无须对您承担任何责任。

涉及非 IBM 产品的信息可从这些产品的供应商、其出版说明或其它可公开获得的资料中获取。IBM 没有对这些产品进行测试，也无法确认其性能的精确性、兼容性或其它任何关于非 IBM 产品的声明。有关非 IBM 产品性能的问题应当向这些产品的供应商提出。

本信息中对非 IBM Web 站点的任何引用都只是为了方便起见才提供的，不以任何方式充当对那些 Web 站点的保证。该 Web 站点中的资料不是 IBM 产品资料的一部分，使用那些 Web 站点带来的风险将由您自行承担。

本信息包含用于日常商业操作的数据和报告的示例。要尽可能对它们进行举例说明，示例应包含个人、公司、商标和产品名。所有名称都是虚构的，实际的商业公司所使用的任何与此相似的名称和地址纯属巧合。

商标

以下各项是国际商业机器公司在美国和 / 或其它国家或地区的商标:

AIX
CICS
CICS/MVS
CICS/VSE
ESCON
IBM
OS/2
OS/2
PAL
S/370
S/390
系统应用程序体系结构
VSE/ESA
VTAM

UNIX 是 The Open Group 在美国和其它国家或地区的注册商标。

Microsoft、MS-DOS、Windows、和 Windows NT 是 Microsoft Corporation 在美国和 / 或其他国家或地区的商标。

其它公司、产品或服务名称可能是其它公司的商标或服务标记。

读者意见表

AIX 5L 版本 5.2
词汇表

姓名
单位及部门
电话号码

地址

读者意见表



请沿此线
撕下或折起

折起并封口

请勿使用钉书机

折起并封口

在此
贴上
邮票

IBM Corporation
Information Development
Department H6DS-905-6C006
11501 Burnet Road
Austin, TX
78758-3493

折起并封口

请勿使用钉书机

折起并封口

请沿此线
撕下或折起



中国印刷