AIX 5L 版本 5.2

命令参考大全，卷 6，v - z

S152-0347-01

AIX 5L 版本 5.2

命令参考大全，卷 6，v - z

S152-0347-01

注

在使用本资料及其支持的产品之前，请阅读第 269 页的附录 C，『声明』中的信息。

第五版（2003 年 5 月）

本版本适用于 AIX 5L V5.2 和本产品的所有后续发行版，直至在新产品中另有说明。

本出版物的后面提供了一张读者意见表。如果该表已除去，则将意见寄往：IBM 中国公司上海分公司汉化部，中国上海市淮海中路 333 号瑞安广场 10 楼，邮政编码：200021。要通过电子的形式提供意见，请使用此商业因特网地址：ctscrcf@cn.ibm.com。我们可能会使用您提供的任何信息，而无须对您承担任何责任。

© Copyright International Business Machines Corporation 1997, 2003. All rights reserved.

目录

关于本书 . . . . . **vii**
本书的使用者 vii
如何使用本书 vii
ISO 9000 ix
UNIX98 规范的 32 位和 64 位支持 x
相关信息 x

按字母排列的命令清单 **1**
vacation 命令 1
val 命令（SCCS） 3
varyoffvg 命令 4
varyonvg 命令 5
vc 命令 7
vgrind 命令 10
vi 或 vedit 命令 11
view 命令 27
vmh 命令 27
vmo 命令 28
vmstat 命令 39
vmtune 命令 46
vpdadd 命令 47
vpddel 命令 49
w 命令 50
wait 命令 51
wall 命令 51
ewallevent、wallevent 命令 52
watch 命令 54
wc 命令 55
what 命令 57
whatis 命令 58
whatnow 命令 59
whereis 命令 63
which 命令 64
which_fileset 命令 64
who 命令 66
whoami 命令 69
whodo 命令 69
whois 命令 71
whom 命令 72
wlmassign 命令 74
wlmcheck 命令 76
wlmcntrl 命令 77
wlmmon 和 wlmperf 命令 80
wlmstat 命令 84
write 命令 86
writesrv 守护程序 90
wsm 命令 91
wsmaccess 命令 92
wsmserver 命令 94

© Copyright IBM Corp. 1997, 2003

wtmpfix 命令 95
wump 命令 96
X 命令 97
x_add_fs_fpe 命令 107
x_add_nfs_fpe 命令 108
x_rm_fpe 命令 109
xargs 命令 110
xauth 命令 113
xclock 命令 115
xcmsdb 命令 117
xdm 命令 118
xfindproxy 命令 131
xfs 命令 132
xget 命令 134
xhost 命令 135
xinit 命令 136
xkbcomp 命令 138
xkbevd 守护程序 139
xkbprint 命令 141
xlock 命令 142
xlsfonts 命令 144
xmbind 命令 145
xmkmf 命令 146
xmodem 命令 147
xmodmap 命令 149
XNSquery 命令 150
XNSrouted 守护程序 151
xntpd 守护程序 152
xntpdc 命令 155
xpr 命令 161
xpreview 命令 163
xrdb 命令 165
xrx 命令 168
xsend 命令 169
xset 命令 170
xsetroot 命令 173
xss 命令 174
xstr 命令 175
xterm 命令 177
xwd 命令 198
xwud 命令 199
yacc 命令 201
yes 命令 203
ypbind 守护程序 203
ypcat 命令 205
ypinit 命令 206
ypmatch 命令 208
yppasswd 命令 209
yppasswdd 守护程序 210
yppoll 命令 211
yppush 命令 212
ypserv 守护程序 213

ypset 命令 215
ypupdated 守护程序 216
ypwhich 命令 217
ypxfr 命令 218
zcat 命令 220
zdump 命令 221
zic 命令 222

附录 A. 对大于 2 吉字节的文件的命令支持 227
不支持大于 2 吉字节的文件的命令 227
支持大于 2 吉字节的文件命令 227

附录 B. 命令的功能列表 229
通信 230
命令列表: 消息处理程序 233
文件和目录 242
常规操作 249
命令列表: 数值数据 256
命令列表: 性能调整 257
编程工具 265

附录 C. 声明 269
商标 270

索引 271

关于本书

本书是六卷《*AIX 5L V5.2 命令参考大全*》中的卷 6，它包含关于操作系统命令的参考信息。它描述每个命令执行的任务、如何修改命令、命令如何处理输入和输出、可由谁运行这些命令，并提供全部六卷的主索引。

命令的快速参考列表安排在功能组中，请参阅卷 6。

支持带 5200-01 推荐的维护软件包的 AIX 5L V5.2 发行版的版本。任何对该维护软件包特定的参考资料都表示为*带 5200-01 的 AIX 5.2*。

本书的使用者

本书适合于操作系统命令的使用者。

如何使用本书

命令是执行操作或运行程序的请求。您能使用命令来指示操作系统希望它执行的任务。当命令输入时，命令解释器（也称之 shell）将之译码，然后任务被处理。

某些命令能简单地通过输入一个词来输入。也可以组合命令以使一个命令的输出成为另一命令的输入。这称为流水线技术。

标志进一步定义命令操作。标志是命令行中与命令名一起使用的修饰符，前面通常加一破折号。

命令也可以组织起来并存储在文件中。这被称为 shell 过程或 shell 脚本。您可以不必个别地执行命令，而是执行包含命令的文件。

某些命令可使用 基于 Web 的系统管理器 应用程序或系统管理接口工具（SMIT）来构造。

突出显示

本书中使用了以下突出显示约定:

粗体	标识命令、子例程、关键字、文件、结构、目录和系统预定义名称的其它项。它也标识图形对象，例如用户选择的按钮、标签和图标。
斜体	标识由用户提供其实际名称或值的参数。
等宽体	标识特定数据值示例、与您所看到的显示的文本相类似的文本示例、与您作为程序员所写的相类似的部分程序代码示例、来自系统的信息或您应实际输入的信息。

格式

每个命令可包含任何下列部分:

用途	描述每个命令的主要功能。
语法	显示命令行选项的语法语句。
描述	详细描述命令功能和使用的讨论。
标志	命令行标志与相关变量的列表，并有标志如何修改命令操作的解释。
参数	命令行参数及其描述的列表。
子命令	说明子命令（对于交互式命令）使用的列表。
出口状态	命令返回出口值的描述。
安全性	指定运行命令所需的任何许可权。

© Copyright IBM Corp. 1997，2003

示例	关于您可如何使用命令的特定示例。
文件	命令所使用文件的列表。
相关信息	本书中相关命令和其它书中相关讨论的列表。

读取语法语句

语法语句是一种表示命令语法的方法，它由诸如中括号（[ ]），大括号（{ }）和竖线（|）等符号构成。以下是关于 **unget** 命令语法语句的一个示例。

unget [-r*SID*] [-**s**] [-**n**] *File* ...

在命令语法语句中，使用以下约定:

- 命令行中的必须照字面原意输入的项用**粗体字**。这些项包含命令名称、标志和文字字符。
- 表示必须用名称替换的变量的项用*斜体字*。这些项包含带有标志的参数和命令所要读取的参数，例如 *Files* 和 *Directories*。
- 中括号所包含的参数为可选项。
- 大括号所包含的参数为必需项。
- 括号、大括号都未包含的参数为必需项。
- 竖条表示您只能选择一个参数。例如，[a | b] 表示您*可以*选择 a、b 或者什么都不选。同样，{ a | b } 表示您*必须*选择 a 或者 b。
- 省略号（...）表示参数可以在命令行中重复。
- 破折号（-）代表标准输入。

可安装的软件包清单

要列出一个单独命令的可安装软件包（文件集），可使用带 **-w** 标志的 **lslpp** 命令。例如，要列出包含 **installp** 命令的文件集，请输入:

```
lslpp -w /usr/sbin/installp
```

输出显示与以下类似:

```
文件                                文件集                  类型
-------------------------------------------------------------------
/usr/sbin/installp                  bos.rte.install           File
```

要列出包含 `installp` 命令的全部文件名的文件集，请输入:

```
lslpp -w "*installp*"
```

输出显示与以下类似:

```
文件                                文件集                  类型
-------------------------------------------------------------------
/usr/sbin/installp                  bos.rte.install           File
/usr/clvm/sbin/linstallpv           prpq.clvm                 File
/usr/lpp/bos.sysmgt/nim/methods/c_installp
                                    bos.sysmgt.nim.client     File
```

在后台运行命令

如果要运行一个花费长时间进行处理的命令，您可指定命令在后台运行。后台处理是一种运行处理较慢的程序的有效方式。要运行一个后台命令，您可在命令尾处使用 `&` 运算符。

```
Command&
```

一旦进程在后台运行，您可继续工作并在系统中输入其它命令。

有时候，您也许想要在一个指定时间或特定日期运行一个命令。使用 **cron** 守护程序，您可调度命令自动地运行。或者，使用 **at** 和 **batch** 命令，您可在稍后时间或系统装入级别允许时运行命令。

输入命令

典型地，您在命令行中的 shell 提示之后输入命令。shell 提示可能有变更。在以下的示例中，$ 是提示。

为显示您当前目录的内容列表，您应输入 `ls` 并按下 Enter 键:

```
$ ls
```

当您输入一个命令并运行时，操作系统不显示 shell 提示。当命令完成操作时，系统会再一次显示提示。这表示您可以输入另一个命令。

输入命令的常规格式为:

Command Flag(s) Parameter

标志改变命令工作的方式。许多命令有几个标志。例如，如果您在 **ls** 命令后输入 **-l**（long）标志，系统将提供关于当前目录内容的附加信息。以下示例显示如何使用 **ls** 命令所带有的 **-l** 标志:

```
$ ls -l
```

由跟随在命令或标志后的字符串所构成的参数。它指定数据（如文件或目录的名称）或值。在以下示例中，名为 **/usr/bin** 的目录是一个参数:

```
$ ls -l /usr/bin
```

当输入命令时，重要的是记住下列内容:

- 命令通常以小写字母输入。
- 标志通常带有一个 -（负号标志）的前缀。
- 如果命令由 **;**（分号）分隔开，则在命令行中可输入多个命令。
- 命令的长序列可通过使用 \ (反斜杠) 在下一行继续。反斜杠应放置在第一行结束处。以下示例显示反斜杠的放置:

  ```
  $ cat /usr/ust/mydir/mydata > \
  /usr/usts/yourdir/yourdata
  ```

当输入某些命令时，会更改 shell 提示。因为有些命令事实上是程序（例如 **telnet** 命令），当您在命令中操作时，提示会更改。您在程序中所发出的任何命令称为子命令。当您退出程序时，提示返回到 shell 提示。

操作系统可用不同的 shell（例如 Bourne、C 或 Korn）操作，您输入的命令由 shell 来解释。因此，您必须知道所使用的 shell 以便输入格式正确的命令。

停止命令

如果您输入了命令后决定停止运行命令，您可以中断命令的进一步处理。要停止命令的处理，按下中断按键顺序（通常是 Ctrl-C 或 Alt-Pause）。当处理停止时，会返回 shell 提示，于是您可以输入另一个命令。

ISO 9000

本产品的开发和生产中使用了 ISO 9000 质量认证体系。

UNIX98 规范的 32 位和 64 位支持

从版本 4.3 开始，为获得基于 UNIX 的操作系统的可移植性，将操作系统设计成为支持 The Open Group 的 UNIX98 规范的。已经将许多新的接口和一些当前接口添加或增强为符合此规范，使版本 4.3 对于应用程序更开放和可移植。

同时，阻止操作系统的先前版本的兼容性。这由新环境变量的创建来完成，可用于在每个系统、每个用户或每个进程的基础上设置系统环境。

要确定开发 UNIX98 可移植应用程序的适当方式，可能需要参考 The Open Group 的 UNIX98 规范，它可能通过订购包含在 CD-ROM *Go Solo 2: The Authorized Guide to Version 2 of the Single UNIX Specification* 中，CD-ROM 中的该书包含 The Open Group 的 UNIX98 规范。

相关信息

下列书籍包含有关或相关命令的信息。

- *《AIX 5L V5.2 命令参考大全，卷 1》*
- *《AIX 5L V5.2 命令参考大全，卷 2》*
- *《AIX 5L V5.2 命令参考大全，卷 3》*
- *《AIX 5L V5.2 命令参考大全，卷 4》*
- *《AIX 5L V5.2 命令参考大全，卷 5》*
- *《AIX 5L V5.2 命令参考大全，卷 6》*
- *AIX 5L Version 5.2 Files Reference*
- *《AIX 5L V5.2 词汇表》*
- *AIX 5L Version 5.2 Guide to Printers and Printing*
- *《AIX 5L V5.2 安装指南与参考大全》*
- *《AIX 5L V5.2 分区环境中的 AIX 安装》*
- *AIX 5L Version 5.2 Network Information Services (NIS and NIS+) Guide*
- *《AIX 5L V5.2 操作系统安装：入门》*
- *《AIX 5L V5.2 性能管理指南》*
- *AIX 5L Version 5.2 Performance Tools Guide and Reference*
- *《AIX 5L V5.2 安全指南》*
- *AIX 5L Version 5.2 System Management Concepts: Operating System and Devices*
- *《AIX 5L V5.2 系统管理指南：操作系统与设备》*
- *《AIX 5L V5.2 系统管理指南：通信与网络》*
- *《AIX 5L V5.2 系统用户指南：通信与网络》*
- *《AIX 5L V5.2 系统用户指南：操作系统与设备》*
- *AIX 5L Version 5.2 Technical Reference: Base Operating System and Extensions Volume 1*
- *AIX 5L Version 5.2 Technical Reference: Base Operating System and Extensions Volume 2*
- *AIX 5L Version 5.2 Technical Reference: Communications Volume 1*
- *AIX 5L Version 5.2 Technical Reference: Communications Volume 2*
- *AIX 5L Version 5.2 Technical Reference: Kernel and Subsystems Volume 1*
- *AIX 5L Version 5.2 Technical Reference: Kernel and Subsystems Volume 2*
- *《AIX 5L V5.2 基于 Web 的系统管理器管理指南》*

- *Performance Toolbox Version 2 and 3 for AIX: Guide and Reference*

按字母排列的命令清单

vacation 命令

用途

返回一条消息给发送方表示邮件接收方正在度假。

语法

vacation [{ **-I** | *User* }] | [{ **-f** *Number* [*Unit*] | *User* }]

描述

vacation 命令返回一条消息给邮件消息的发送方，通知发送方：接收人正在度假。指定用法在 **$HOME/.forward** 文件中，它容许消息发送给您，同时也发送一条消息返回给发送方。

注：Sendmail V8.9.3 和后续发行版有安全性增强，如果存在以下两种情况中的一种，它将忽略 **.forward** 文件：

- **.forward** 文件具有组或通用的可写许可权。
- 任何 **.forward** 文件的父目录都具有组或通用的可写许可权。

如果认为 vacation 程序由于 **.forward** 文件被忽略而不运行，请检测许可权。如果必须有对 **.forward** 文件任何父目录的组或通用的写许可权，则使用正确的值设置 sendmail 配置文件中的 DontBlameSendmail 选项。

vacation 命令需要包含回送给每位发送者的消息的 **$HOME/.vacation.msg** 文件。如果不存在此文件，**vacation** 命令则查找 **/usr/share/lib/vacation.def**，这是系统范围的缺省 vacation 消息文件。它应当是一条完整的消息，包含任何所需的报头，如`From` 或 `Subject`。缺省情况下，该消息仅每星期一次发送给每位发送邮件给您的人。使用 **-f** 标志更改发送消息的频率间隔。发送消息的人名保存在文件 **$HOME/.vacation.pag** 和 **$HOME/.vacation.dir** 中。当使用 **-I**（大写 i）标志为用户标识进行 **vacation** 命令初始化时，系统创建这些文件。

如果没有指定 **-I** 标志，**vacation** 命令从标准输入读取首行来查找 `From` 行以确定发送方。标准输入中如果没有文本可用，则命令返回一条错误消息。所有正确格式化的来件都有一个 `From` 行。如果 `From` 标题行表明消息是来自 `Postmaster`、MAILER-DAEMON，或如果初始 `From` 行包含字符串 `REQUEST@`，或如果 `Precedence: bulk` 或 `Precedence: junk` 行包含在头中，则不发送消息。

标志

-I
初始化 **$HOME/.vacation.pag** 和 **$HOME/.vacation.dir** 文件。在修改 **$HOME/.forward** 文件之前，使用此标志执行 **vacation** 命令。

-f*Number* [*Unit*]
指定发送度假消息的频率间隔。*Number* 参数是一个整数值， *Unit* 参数指定时间单位。*Unit* 参数可以是以下值之一：

s	秒
m	分
h	小时
d	天
w	周

© Copyright IBM Corp. 1997，2003

注: **-f** 标志不能与 **-I** 标志一起使用。

示例

1. 在使用**vacation** 命令以返回一条表明正在度假的消息给发送方之前，您必须初始化 **$HOME/.vacation.pag** 和 **$HOME/.vacation.dir** 文件。要初始化这些文件，请输入:

   ```
   vacation  -I
   ```

2. 修改 **.forward** 文件。例如，Mark 在 **.forward** 文件中输入以下语句:

   ```
   mark,|"/usr/bin/vacation mark"
   ```

   发送方接收 **$HOME/.vacation.msg** 文件中的消息；或者如果该文件不存在，则接收 **/usr/share/lib/vacation.def** 文件中找到的缺省消息。如果这两个文件都不存在，不会给邮件消息发送方发送自动应答，且不生成错误消息。如果这两个文件中有一个存在，发送方每周会从 mark 收到一条 vacation 消息，不管发送方给mark 发送多少消息。

3. 如果 **.forward** 文件中包含以下条目:

   ```
   mark, |"/usr/bin/vacation -f10d mark"
   ```

   发送方每十天从 mark 接收一条 vacation 消息，不管发送方给 mark 发送多少消息。

4. 要创建与缺省度假消息不同的 vacation 消息，创建文件 **$HOME/.vacation.msg** 并将消息添加到此文件。以下是一条 vacation 消息的示例:

   ```
   From: mark@odin.valhalla (Mark Smith)
   Subject: I am on vacation.
   Delivered-By-The-Graces-Of: the Vacation program
   I am on vacation until October 1. If you have something urgent,
   please contact Jim Terry <terry@zeus.valhalla>.
      --mark
   ```

5. 要取消 vacation 消息，从 **$HOME**（登录）目录除去 **.forward** 文件、**.vacation.dir** 文件、**.vacation.pag** 文件和 **.vacation.msg** 文件:

   ```
   rm .forward .vacation.dir .vacation.pag .vacation.msg
   ```

文件

$HOME/.forward	包含想将邮件转发至的人名。
/usr/share/lib/vacation.def	包含系统范围内的缺省 **vacation** 消息。
$HOME/.vacation.dir	包含使用 **vacation** 命令时发送邮件给您的人名。
$HOME/.vacation.msg	包含个人化的 **vacation** 消息。
$HOME/.vacation.pag	包含使用 **vacation** 命令时发送邮件给您的人名。
/usr/bin/vacation	包含 **vacation** 命令。

相关信息

mail 命令、**sendmail** 命令。

.forward 文件。

*《AIX 5L V5.2 系统用户指南: 通信与网络》*中的『邮件概述』、『转发邮件』和『发送度假消息通知』。

*《AIX 5L V5.2 系统用户指南: 操作系统与设备》*中的『理解目录』。

val 命令（SCCS）

用途

验证 SCCS 文件。

语法

val [**-s**] [**-r***SID*] [**-m***Name*] [**-y***Type*] *File* ...

描述

val 命令读取指定文件以确定它是不是符合伴随标志指定的特征的源代码控制系统（SCCS）文件。如果指定一个 **-**（负号）为 *File* 值，**val** 程序读取标准输入并将标准输入的每一行解释成 **val** 标志和 SCCS 文件名。文件结束符用于终止输入。

val 命令为每一个被处理的文件将消息显示到标准输出。

标志

每个标志或标志组独立应用到每个指定的文件。标志可以按任何顺序出现。

-m *Name* 将 *Name* 值与指定文件中的 SCCS **31** 标识关键字进行比较。关于标识关键字的信息，请参阅 **get** 命令。
-r *SID* 指定要验证的文件的 SID。SID 必须有效并且无二义性。
-s 消除正常情况下写入标准输出的错误消息。
-y *Type* 指定一种类型与指定文件中的 SCCS 标识关键字进行比较。

出口状态

如果对所有文件都成功执行，**val** 命令返回 0，否则将返回一个从可能的错误中析取的 8 位代码。它作为位串解释，该位串中设置位按如下方式从左至右进行解释:

0x80 遗漏文件自变量。
0x40 选项未知或重复。
0x20 SCCS 文件损坏。
0x10 不能打开文件或文件不是 SCCS。
0x08 SID 无效或存在二义性。
0x04 SID 不存在。
0x02 y 不匹配。
0x01 31，m 不匹配。

> **注:** **val** 命令在一个给定命令行上可以处理两个或更多的文件，并能处理多个命令行（当读取标准输入时）。在上述情况下，返回的是一个聚集代码；该聚集代码是为每一命令行和文件处理生成的代码的逻辑“或”。

示例

要确定 `s.test.c` 文件是否是 SCCS 文本文件，请输入:

```
val -ytext s.test.c
```

相关信息

AIX 5L Version 5.2 General Programming Concepts: Writing and Debugging Programs 中的 List of SCCS Commands。

admin 命令、**delta** 命令、**get** 命令、**prs** 命令。

sccsfile 文件格式。

AIX 5L Version 5.2 General Programming Concepts: Writing and Debugging Programs 中的 Source Code Control System (SCCS) Overview。

---

varyoffvg 命令

用途

释放卷组。

语法

varyoffvg [**-s**] *VolumeGroup*

描述

varyoffvg 命令释放由 *VolumeGroup* 参数指定的卷组及其相关的逻辑卷。首先，必须关闭逻辑卷。例如，如果逻辑卷包含一个文件系统，就必须将它卸装。

要激活卷组，请使用 **varyonvg** 命令。

> **注**: 要使用此命令，必须拥有 root 用户权限，或者是 **system** 组成员。

可以使用系统管理接口工具（SMIT）来运行该命令。要使用 SMIT，请输入:

```
smit varyoffvg
```

> **注**: 当调页空间处于活动状态时，无法使一个有调页空间卷的卷组脱机。在释放有活动调页空间卷的卷组之前，请确保在系统初始化时调页空间没被自动激活，然后重新引导系统。

标志

-s 将卷组置于“系统管理”方式，使得在卷组上只能使用逻辑卷命令。在此方式中，用户不能打开或访问任何逻辑卷。

示例

1. 要释放卷组 `vg03`，请输入:

   ```
   varyoffvg vg03
   ```

2. 要释放卷组 `vg02` 但是容许逻辑卷命令继续有效，请输入:

   ```
   varyoffvg  -s vg02
   ```

   卷组中的逻辑卷无法打开，但是逻辑卷命令继续有效。

文件

/usr/sbin/varyoffvg 包含 **varyoffvg** 命令。

相关信息

exportvg 命令、**mount** 命令、**umount** 命令、**varyonvg** 命令。

AIX 5L Version 5.2 System Management Concepts: Operating System and Devices 中的 System Management Interface Tool 解释了结构、主菜单和 SMIT 完成的任务。

AIX 5L Version 5.2 System Management Concepts: Operating System and Devices 中的 Logical Volumes 解释了逻辑卷管理器、物理卷、逻辑卷、卷组、组织形式，以确保数据完整性和分配特征。

---

varyonvg 命令

用途

激活卷组。

语法

varyonvg [**-b**] [**-c**] [**-f**] [**-n**] [**-p**] [**-r**] [**-s**] [**-t**] [**-u**] *VolumeGroup*

描述

varyonvg 命令激活由 *VolumeGroup* 参数指定的卷组及其所有相关的逻辑卷。被激活的卷组是可以使用的。当卷组被激活时，如果物理分区不是当前的，则它们将被同步。

无论何时，当设备配置数据库与存储在逻辑卷管理器中的信息不符时，所有物理卷列表及其状态会显示到标准输出。卷组可以是联机的，也可以不是。必须仔细检查列表，并根据报告的状态采取适当的操作以保持系统的完整性。

当以并行方式联机时，如果 varyon 进程检测到有系统先前不知道的逻辑卷，则导入它们的定义。新设备专门文件的许可权和所有权被复制到卷组专门文件的许可权和所有权。如果已经在创建它的节点上更改了这些逻辑卷的设备专门文件的许可权和／或所有权，则必须在该节点执行同样的更改。

如果*卷组*因损失大部分物理卷而无法联机，则显示所有物理卷及其状态的列表。要在此情况下使这些*卷组*联机，必须使用强制选项。

如果大部分物理卷不可访问（没有 Quorum），**varyonvg** 将不能使卷组联机。即使已禁用 quorum 检查，也是如此。禁用 quorum 检查只能确保即使在 quorum 丢失的情况下卷组仍然联机。

如果有任何物理卷处于 `PV_MISSING` 状态且 quorum 检查被禁用，则*卷组*将不会联机。即使有一组可用磁盘，也是如此。要在此情况下联机，或者使用强制选项，或者将环境变量 `MISSINGPV_VARYON` 设置为 TRUE（如果该卷组在引导时丢失磁盘的情况下联机，在 **/etc/environment** 中设置此值。）

在以上情况中（使用强制联机选项和使用 `MISSINGPV_VARYON` 参数），您应对*卷组*完整性完全负责。

> **注:** 要使用此命令，必须拥有 root 用户权限，或者是 **system** 组的成员。

可以使用系统管理接口工具（SMIT）来运行此命令。要使用 SMIT，请输入:

标志

-b 中断因正常运行 **varyonvg** 命令而被锁定的磁盘上的磁盘保留。在已经联机的卷组上使用此标志。此标志仅适用于 AIX 4.2 或更新版本。

注:

- 此标志为一个给定卷组中所有的磁盘解锁。
- **-b** 标志打开使用 **SC_FORCED_OPEN** 标志的卷组中的磁盘。对于 SCSI 和 FC 磁盘，此标志强制打开该磁盘所驻留的目标地址上的所有 lun。因此，当使用此**联机**选项时，卷组不应该共享目标地址。
- 如果在包含活动调页空间的卷组上使用，**-b** 标志会导致系统挂起。

-c 以“并发”方式或“增强并发”方式使卷组联机。仅当卷组是“并发可行”或“增强并发可行”的，并且系统已装入 HACAMP 产品且可用时，才有可能实现上述操作。如果两者都不成立，卷组不能联机。

如果联机进程检测到在卷组中有一个新的逻辑卷，其名称已经被一个现有的逻辑卷使用，则联机失败。在试图再次联机之前，必须重命名此现有的逻辑卷。

注:

1. “增强并发”卷组使用“组服务”。“组服务”和 HACMP ES 一起提供，并且必须在以此方式激活卷组之前进行配置。
2. 当在 64 位内核上运行时，只支持“增强并发可行”的卷组。当在 64 位内核上运行时，不支持“并发可行”的卷组。

-f 允许激活在当前没有一组可用磁盘的卷组。所有不能转变为激活状态的磁盘将置于被除去状态。在卷组中至少必须有一个磁盘可用。

-n 禁止 *VolumeGroup* 内的旧物理分区的同步。

-p 要使用 **varyonvg** 命令，所有的物理卷都必须可用。

-r 以只读方式使卷组联机。此方式防止:

- 对逻辑卷的写操作
- LVM 元数据更新
- 旧分区同步

注: 所有要求 LVM 元数据更新的 LVM 高级命令的请求将在此方式下失败。

-s 使卷组只在“系统管理”方式中可用。逻辑卷命令能对卷组执行操作，但是不能为输入或输出打开逻辑卷。

注: 逻辑卷命令也不能读取或写入用 **-s** 标志联机的卷组中的逻辑卷。如果逻辑卷命令试图写入用 **-s** 标志联机的卷组内的某个逻辑卷（如 **chvg** 或 **mklvcopy**），那么可能显示错误消息，表明它们不能写入和／或读取逻辑卷。

-t 检查设备配置数据库和逻辑卷管理器中的时间戳记。如果时间戳记不符，则发出 **synclvodm** 命令使数据库同步。

注: 如果卷组以并发方式联机，会始终执行此检查。

-u 使卷组联机，但使组成卷组的磁盘处于未锁定状态。将该标志作为一个休眠卷组的初始联机的一部分。该标志仅适用于 AIX 4.2 或更新版本。

注意: LVM 的基本设计假定只有一个引发设备能访问卷组。HACMP 产品确实和 LVM 一起工作，以便使一个共享卷组的多节点访问同步。但是，多引发设备节点可以轻易地访问具有 **-b** 和 **-u** 标志的卷组，并且不使用 *HACMP*。必须意识到，因为这两个标志绕过磁盘保护（锁定），卷组状态信息可能折衷或变得模糊不清。如果使用 **-b** 和 **-u** 标志，数据和状态输出无法确保一致。

示例

1. 要激活卷组 `vg03`，请输入:

   ```
   varyonvg vg03
   ```

2. 要激活卷组 `vg03` 而不同步非当前分区，请输入:

   ```
   varyonvg -n vg03
   ```

文件

/usr/sbin	包含 **varyonvg** 命令目录。
/tmp	当命令运行时存储临时文件。

相关信息

chvg 命令、**lspv** 命令、**lslv** 命令、**lsvg** 命令、**varyoffvg** 命令。

AIX 5L Version 5.2 System Management Concepts: Operating System and Devices 中的 System Management Interface Tool 解释结构、主菜单和 SMIT 完成的任务。

AIX 5L Version 5.2 System Management Concepts: Operating System and Devices 中的 Logical Volumes 解释逻辑卷管理器、物理卷、逻辑卷、卷组、组织形式，确保数据完整性以及分配特征。

---

vc 命令

用途

用标识关键字代替已分配的值。

语法

vc [**-a**] [**-t**] [**-s**] [**-c***Character*] [*Keyword*=*Value*]...

描述

vc 命令将各行从标准输入复制到标准输出。输入中的命令行和控制语句内的标志和关键字修改作为结果的输出。**vc** 命令将用户声明的关键字替换为命令行上分配的值。关键字可以在文本和控制语句内被替代。

控制语句

控制语句是以控制字符（缺省控制字符是一个 **:**(冒号)）开始的单个行。控制语句提供对输入的条件处理。允许的控制语句类型为:

:if *Condition*

Text

:end

只有条件为真时，才将 **:if** 语句和相匹配的 **:end** 之间的所有行写入标准输出。**:if** 和 **:end** 语句可以嵌套。但是，一旦条件为假，所有保留的嵌套 **:if** 和 **:end** 语句都会被忽略。有关条件语法和允许的操作，请参阅“条件语法”一节。

:dcl *Keyword*, [*Keyword* . . .]

声明指定的关键字。所有的关键字都必须加以声明。

:asg *Keyword*=*Value*
将指定的值赋给指定的关键字。**:asg** 语句优先于 **vc** 命令行中的关键字赋值。后面的 **:asg** 语句覆盖所有前面的相关关键字赋值。已声明但没有赋值 *Value* 的关键字具有空值。

:: *Text*
除去两个前导控制符，用它们各自的值替换关键字 *s*，然后将该行复制到标准输出。

:on 或 **:off**
打开或关闭所有行的关键字替换。

:ctl *Character*
将控制字符更改为 *Character* 值。

:msg *Message*
按以下格式将消息写入标准错误输出：Message(n)：message

其中，n 是消息出现的输入行的行号。

:err *Message*
将错误消息写入标准错误。**vc** 命令停止处理并返回出口值 1。该错误消息格式如下：

```
ERROR: message
ERROR: err statement on line n (vc15)
```

条件语法

以下是语法项和允许的语句：

```
条件          ::=OR 语句
                  ::=NOR 语句
OR 语句              ::=AND 语句
                  ::=AND 语句 | OR 语句
AND 语句      ::=表达式
                  ::=表达式 & AND 语句
表达式         ::=( OR 语句)
                  ::=值 运算符 值
运算符 值     ::= = 或 != 或 < 或 >
                  ::= ASCII 字符串
                  ::= 数字字符串
```

可用的条件运算符及其含义为：

=	等于
!=	不等于
&	AND（与）
&|	OR（或）
>	大于
<	小于
( )	用于逻辑分组
NOT	只可能紧跟 *if* 之后，并且使用时反转整个条件的值。

> 和 <(大于和小于)仅操作无符号整数值；例如，012 > 12 结果为假。所有其它的运算符将字符串作为修改量；例如，012 ! = 12 结果为真。运算符的优先级从最高到最低优先级排列如下：

- = ! = > <（所有优先权相等）
- &
- &|

圆括号可以用于更改优先级顺序。

值必须用至少一个空格或制表符与运算符或圆括号分开。

关键字替换

关键字必须以控制语句中使用的同一控制字符开始和结束。一个关键字可以最多有 9 个字母数字字符，其中第一个字符必须是字母。关键字可以是任意 ASCII 字符串。数据关键字 *Value* 是数字的无符号字符串。值不能包含制表符或空格。

标志

-a 在所有的文本行（不只是那些以两个控制字符开始的行），将控制字符包围的关键字 *s* 用其分配的值替换。

-c*Character* 将 *Character* 值用作控制字符。*Character* 参数必须指定一个 ASCII 字符。

-s 不显示通常显示到标准错误的警告消息。

-t 为检测控制语句，忽略从行首开始一直到第一个制表符（包括第一个制表符）的所有字符。如果 **vc** 命令查找到一个控制字符，则它忽略直到制表符（包括制表符）的所有字符。

出口状态

此命令返回下列出口值:

0 成功完成。

>0 发生错误。

示例

1. *Keyword***=***Value* 赋值的示例:

   ```
   numlines=4
   prog=acctg
   pass4=yes
   ```

   当将文本写入标准输出时，**vc** 命令从用两个控制符标记的输入文本行除去所有的控制字符和关键字。

2. 要防止解释控制字符，在它的前面插入一个反斜杠，如以下示例所示:

   ```
   ::the :prog: program includes several of the following\:
   ```

   :prog: 关键字替换为它的值，但是 **\:** 作为 **:**（冒号）传输到标准输出。

   以 \ (反斜杠) 开始、紧跟着控制符号的输入行不是控制行，而且除反斜杠之外被复制到标准输出。但是，**vc** 命令写入以反斜杠开始并且没有控制字符跟随的行，而不作任何更改（包括初始反斜杠）。

文件

/usr/bin/vc 包含 **vc** 命令。

相关信息

admin 命令、**delta** 命令、**get** 命令。

AIX 5L Version 5.2 General Programming Concepts: Writing and Debugging Programs 中的 List of SCCS Commands。

AIX 5L Version 5.2 General Programming Concepts: Writing and Debugging Programs 中的 Source Code Control System (SCCS) Overview。

vgrind 命令

用途

格式化易于阅读的程序清单。

语法

vgrind [**-f**] [**-n**] [**-t**] [**-x**] [**-P***Printdev*] [**-T***Name*] [**-**] [**-d***File*] [**-h** *Header*] [**-l***Language*] [**-s***Size*] [*File* ...]

描述

vgrind 命令使用 **troff** 命令以易于阅读的风格格式化（编排）由 *File* 参数指定的程序源文件。注释以斜体字显示、关键字以粗体字显示，当前函数的名称在对应的每页的页边空白中列出。

vgrind 命令以过滤器方式或正常方式运行。

在过滤器方式中，**vgrind** 命令以一种类似于 **tbl** 命令的方式充当过滤器。标准输入直接被传递到标准输出，除了用以下 **troff** 样式的宏括起来的行:

.vS 开始处理。
.vE 结束处理。

前面的行按照 **vgrind** 命令约定被格式化。来自过滤器的输出可以被传递到 **troff** 命令用作输出。**eqn** 或 **tbl** 命令没有特定的顺序。

在正常方式中，**vgrind** 命令接受输入文件、处理它们并依次传递到 **troff** 命令、适当的后处理器，然后传给打印机。

在两种方式中，**vgrind** 命令传递时不转换以小数点开始的行。

vgrind 命令仅支持标准 **/usr/share/lib/vgrindefs** 语言定义文件定义的 ASCII 关键字，或任何由 **-d** 标志另外指定的文件定义的 ASCII 关键字。

标志

-f 强制使用过滤器方式。
-n 强制不使用粗体关键字。
-t 使格式化文本写入标准输出。
-x 以一种易于阅读的格式输出索引文件。无论什么时候用 **index** 文件在当前目录运行 **vgrind**命令，索引文件就会自动产生。通过运行使用 **-x** 标志和 *File* 参数的 **vgrind** 命令，可以关闭功能定义索引。
-P*PrintDev* 将输出发送到使用 **qprt** 命令的 *Printdev* 打印机。如果没有指定此标志，则使用 **PRINTER** 环境变量。如果没有设置 **PRINTER** 环境变量，则使用系统缺省值。
-T*Name* 为由 *Name* 参数指定的 **troff** 设备创建输出。输出通过适当的后处理器送出。缺省为 **ibm3816** 后处理器。
- 强制从标准输入获取输入（如果指定 **-f** 标志，则为缺省值）。
-d*File* 指定一种备用语言定义文件（缺省为 **/usr/share/lib/vgrindefs** 文件）。
-h *Header* 指定特定的页眉以便打印在每个输出页上（缺省为文件名）。

注: 在 **-h** 标志之后，*Header* 变量之前，要求有一个空格。

-l*Language* 指定要使用的语言。当前已知的语言有:

c C（缺省）。函数名仅可以在一行上用空格、制表符或星号作为前导。圆括号选项也必须位于同一行。

csh CSH。

p PASCAL。诸函数名必须作为 **function** 或 **procedure** 关键字在同一行上显示。

m MODEL。诸函数名必须作为 **isbeginproc** 关键字短句在同一行上显示。

sh SHELL。

r RATFOR。

mod2 MODULA2。

yacc YACC。

isp ISP。

l ICON。

-s *Size* 指定在输出上使用的点大小（和 **.ps** 请求完全相同）。

文件

index	包含在其中创建索引源的文件。
/usr/bin/vgrind	包含 **vgrind** 命令。
/usr/share/lib/tmac/tmac.vgrind	包含宏数据包。
/usr/share/lib/vfontedpr	包含预处理器。
/usr/share/lib/vgrindefs	包含语言描述。

相关信息

qprt 命令、**tbl** 命令、**troff** 命令。

vgrindefs 文件格式。

---

vi 或 vedit 命令

用途

用全屏显示编辑文件。

语法

{ **vi** | **vedit** } [**-l**] [**-R**] [**-t***Tag*] [**-v**] [**-w***Number*] [**-y***Number*] [**-r** [*File*]] [{ **+** | **-c** } { *Subcommand* }] [*File* ...]

描述

vi 命令启动基于基础 ex 编辑器的全屏编辑器。因此，ex 子命令可以在 vi 编辑器中使用。**vedit** 命令启动适用于初学者的 vi 编辑器版本。在 vedit 编辑器中，**报告**选项设置为 1，设置了 **showmode** 选项和 **novice** 选项，使其成为一个行编辑器。

您可以通过指定要编辑的文件名启动 vi 编辑器。如果在命令行上提供多个 *File* 参数，vi 编辑器按指定的顺序编辑每一个文件。现有文件的 vi 编辑器在屏幕底部显示文件名、行数和字符数。在多字节语言环境的情况下，字符数要解释成字节数。

由于 vi 编辑器是全屏编辑器，所以可以逐屏编辑文本。vi 编辑器在编辑缓冲区为正在编辑的文件制作副本，并且文件内容直到保存更改才会更改。光标在显示屏幕上的位置表明它在文件中的位置，并且子命令在光标位置影响该文件。

vi 编辑器的限制

以下列表提供了 vi 编辑器的最大限制。这些计数采用单字节字符集。

- 每行 {LINE_MAX} 个字符
- 每个全局命令列表 256 个字符
- 在以前被插入和删除的文本中 128 个字符
- 在一个 shell 转义命令中 128 个字符
- 在一个字符串值选项中 128 个字符
- 在一个标记名中 30 个字符
- 128 个映射宏，总共 2048 个字符
- 1,048,560 行，每行强制不能超过 {LINE_MAX} 个字符
- 宏名和宏文本被限制在 100 个字符内。

注: 对大于 64MB 的文件运行 vi 编辑器可能会显示以下的错误消息:

```
0602-103 文件太大而不能放置在 /tmp 中
```

vi 编辑方式

vi 编辑器按以下方式操作:

命令方式
当启动 vi 编辑器时，它处于命令方式。您可以输入任何子命令，除了那些指定仅用于文本输入方式的子命令。当子命令和其它方式结束时，vi 编辑器返回命令方式。按 Esc 键取消子命令。

文本输入方式
在此方式中使用 vi 编辑器来添加文本。用以下任意子命令进入文本输入方式: **a** 子命令、**A** 子命令、**i** 子命令、**I** 子命令、**o** 子命令、**O** 子命令、**c***x* 子命令（其中 *x* 代表子命令的范围）、**C** 子命令、**s** 子命令、**S** 子命令和 **R** 子命令。输入这些子命令之一后，即可输入文本到编辑缓冲区中。要返回命令方式，按 ESC 键正常退出或按“中断”（Ctrl-C 按键顺序）非正常结束。

最后行方式
带前缀 **:**（冒号）、**/**（斜杠）、**?**（问号）、**!**（感叹号）或 **!!**（两个感叹号）的子命令在屏幕底部的显示行上读取输入。当输入初始字符时，vi 编辑器将光标置于屏幕底部，在该处可以输入命令中剩余的字符。按 Enter 键运行该子命令，或按“中断”（Ctrl-C 按键顺序）取消它。当使用 **!!** 前缀时，光标只在两个感叹号都输入后才移动。当使用 **:** 前缀进入最后行方式时，如果在指定计数的命令之前使用以下字符，vi 编辑器赋予它们特定含义:

% 所有行，而不管光标的位置

$ 最后行

. 当前行

定制 vi 编辑器

可以按照以下定制 vi 编辑器:

- 设置 vi 编辑器选项

- 定义宏
- 映射键
- 设置缩写

设置 vi 编辑器选项: 以下列表描述可以使用 **set** 命令更改的 vi 编辑器选项。这些选项的缺省设置是 **off**。如果打开了这些开关选项中的一个，在选项之前输入 **no** 即可重新关闭它。如果想中断 **autowrite** vi 选项，则输入 **noaw**，其中 **no** 关闭选项，**aw** 指定 **autowrite** 选项。

注: 当输入 vi 选项时不要包含圆括号。

vi 选项（缩写）	**描述**
autoindent（ai）	通过使用 **shiftwidth** 选项指定的制表符停止位之间的空格，在文本输入方式中自动缩排为前一行的缩排格式。缺省值是 **noai**。将光标退回到前面的制表符停止位，然后按 Ctrl-D 按键顺序。该选项对于全局命令不起作用。
autoprin（ap）	在任何更改编辑缓冲区的命令后面打印当前行。缺省值是 **ap**。该选项只应用于一行上一系列命令中的最后一个命令，并且对全局命令不起作用。
autowrite（aw）	如果编辑缓冲区自上次 **write** 子命令以来发生更改，则在 **:n** 子命令、**:ta** 子命令、Ctrl-A 按键顺序和 **!** 子命令之前，将编辑缓冲区自动写入到文件。缺省值是 **noaw**。
beautifying text（bf）	在文本输入（除了制表符、换行符和换页指示符）的过程中，防止用户在编辑缓冲区中输入控制字符。缺省值是 **nobf**。此选项应用于命令输入。
closepunct（cp=）	处理结束标点列表，特别当文本自动换行（**wraptype** 选项）时。在多字符标点前指出字数，例如 `cp=3..;)}` 。当自动换行时，**vi** 命令不分割结束标点。
directory（dir=）	显示包含编辑缓冲区的目录。缺省值是 **dir = /var/tmp**。
edcompatible（ed）	在多重替换过程中，保留 **g**（全局）和 **c**（确认）子命令后缀，并使 **r**（读取）后缀像 **r** 子命令一样工作。缺省值是 **noed**。
exrc（exrc）	如果未设置，则在初始化过程中忽略当前目录下任何 **.exrc** 文件，除非当前目录是 **HOME** 环境变量命名的。缺省值是 **noexrc**。
hardtabs（ht=）	告诉 vi 编辑器显示屏幕上硬件制表符停止位之间的距离。（该选项必须与基础的终端或终端仿真器的制表符设置相匹配。）缺省值是 **ht=8**。
ignorecase（ic）	当搜索正则表达式时，忽略大写和小写之间的区别。缺省值是 **noic**。
linelimit（ll=）	设置最大行数，如每个 **-y** 命令行选项。该选项只有和 **.exrc** 文件或 **EXINIT** 环境变量一起使用时有效。
lisp（lisp）	除去 ( ), { }, [[和]] 的特殊意义，并为 s 表达式启用 =（格式化打印）运算符，以便能够编辑行处理（LISP）程序。缺省值是 **nolisp**。
list（list）	显示带制表符 (^I) 的文本和行的标记端（$）。缺省值是 **nolist**。
magic（magic）	当搜索一种模式时，将 .（句号）、[（左括号）和 *（星号）字符当成特殊字符。在关闭方式中，只有 (（圆括号）和 $（美元符）保持特殊意义。但是，可以通过在其它字符前面加上 \ (反斜杠) 来激活其特殊意义。缺省值是 **magic**。
mesg（mesg）	如果以可视方式设置，打开对终端的写入许可权。该选项只有和 **.exrc** 文件或**EXINIT** 环境变量一起使用时有效。缺省值是 **on**。
modeline（modeline）	如果在文件开始五行或最后五行找到，则运行 vi 编辑器命令行。vi 编辑器命令行可以位于一行中的任意位置。为了便于 vi 编辑器能辨认命令行，该行必须包含一个空格或制表符，后跟 `ex:` 或 `vi:` 字符串。命令行以第二个 **:**（冒号）结束。vi 编辑器尝试将第一个和第二个冒号之间的任意数据解释为 vi 编辑器命令。缺省值是 **nomodeline**。
novice	指明是否处于 **novice** 方式。您不能使用 **set** 命令更改该值。
number（nu）	显示以行号作为前缀的行。缺省值是 **nonu**。

vi 选项（缩写）	描述
optimize（opt）	加速缺少光标寻址的终端的操作。缺省值是 **noopt**。
paragraphs（para=）	定义开始段落的 vi 宏名。缺省值是 **para=IPLPPPQPP\ LIppIpipnpbp**。如果重新指定了一个段落，单字母 **nroff** 宏（如 **.P** 宏）必须将空格包含为被引用字符。
partialchar（pc=）	出现在最后的显示列，该列中双宽字符不能完全显示。缺省字符为 **-**（负号）。
prompt	当处于命令方式时，通过打印 **:**（冒号）提示输入新的 vi 编辑器命令。缺省值是 **on**。
readonly（ro）	设置永久的只读方式。缺省值是 **noreadonly**。
redraw（redraw）	在哑工作站上模仿智能工作站。缺省值是 **nore**。
remap	允许根据其它宏来定义宏。缺省值是 **on**。
report（re=）	设置在消息显示之前命令能重复的次数。对于产生许多消息的子命令，例如全局子命令，完成命令序列后显示消息。缺省值是 **report=5**。
scroll（scr=）	设置用户上下滚动窗口内容时滚动的行数。缺省值是窗口大小的 1/2，四舍五入。
sections（sect=）	定义开始段落的 vi 宏名。缺省值是 **sect=NHSHHH\ HUuhsh+c**。如果指定段落，单字母 **nroff** 宏（如 **.P** 宏）必须将空格包含为被引用字符。
shell（sh=）	为 **!** 子命令或 **:!** 子命令定义 shell。缺省值是登录 shell。
shiftwidth（sw=）	为 **autoindent** 选项、shift 命令（**>** 和 **<**）和文本输入命令（Ctrl-D 和 Ctrl-T 按键顺序）使用的软件制表符停止位设置间距。vi 选项只影响行首的缩排。缺省值是 **sw=8**。
showmatch（sm）	当如输入)（右圆括号）或 } (右花括号)时显示（（匹配的左圆括号）或 { (左花括号)。缺省值是 **nosm**。
showmode（smd）	当 vi 编辑器处于输入方式时显示消息加以指明。缺省值是 **nosmd**。
slowopen（slow）	在插入期间，延迟更新显示屏幕。缺省值是 **noslow**。
tabstop（ts=）	在显示的文件中设置制表符停止位之间的间距。缺省值是 **ts=8**。
tags（tags =）	为使用 **ctags** 命令创建的函数名的数据库文件定义搜索路径。缺省值是 **tags=tags\ /usr/lib/tags**。
term（term=）	设置正在使用的工作站类型。缺省值是 **term=$TERM**，其中 **$TERM** 是 **TERM** shell 变量的值。
terse（terse）	允许 vi 编辑器显示消息的短格式。缺省值是 **noterse**。
timeout（to）	为字符的输入设置两秒的时间限制。当设置 **timeout** 选项时，此限制容许宏中的字符作为单独的字符输入和处理。要恢复宏的使用，请设置 **notimeout** 选项。缺省值是 **to**。
ttytype	为正在使用的终端指明 tty 类型。从 vi 编辑器不能更改此值。
warn（warn）	在 **!** 之前显示警告消息。当在编辑缓冲区中已经作了更改，但还没有写入文件时，如果是第一次发送 shell 命令，那么子命令执行 shell 命令。缺省值是 **warn**。
window（wi=）	设置显示在一个文本窗口中的行数。缺省值取决于操作时使用的波特率：600 波特或更少，8 行；1200 波特，16 行；更高的速率，全屏减去 1 行。
wrapmargin（wm=）	设置从一行到下一行自动换行的页边距。缺省值是 **wm=0**。0 值关闭自动换行。
wrapscan（ws）	允许字符串搜索从编辑缓冲区结尾到开始进行自动换行。缺省值是 **ws**。

vi 选项（缩写）	描述
wraptype（wt=）	指明用于在行尾使单词换行的方法。缺省值是 **general**。可以指定下列四个值中的一个: **general** 允许在单词中断处（如两个字符之间的空白）自动换行。此设置为缺省值。 **word** 允许在单词上自动换行。 **rigid** 允许在列上和结束标点之前自动换行。 **flexible** 允许在列上自动换行，但是一个标点符号可以扩展到超过页边距。
writeany（wa）	关闭通常在 **write** 子命令之前进行的检查。缺省值是 **nowa**。

要查看根据缺省设置而更改的 vi 编辑器设置列表，请输入 `set` 并按空格键。按 Enter 键返回到命令方式。

要查看 vi 编辑器设置的完全列表，请输入 `set all`。按 Enter 键返回到命令方式。

要打开 vi 编辑器选项，请输入 `set Option`。此命令自动返回到命令方式。

要打开多个 vi 编辑器选项，请输入 `set Option Option Option`。此命令打开三个指定的 vi 编辑器选项并返回到命令方式。

要关闭一个 vi 选项，请输入 `set no Option`。此命令自动返回到命令方式。

要更改一个 vi 编辑器选项的值，请输入 `set Option=Value`。此命令自动返回到命令方式。

可以仅为此编辑会话使用 **:set** vi 编辑器子命令来设置选项，或为此编辑会话和所有将来的编辑会话设置选项。

要*只为此编辑会话*设置或更改 vi 编辑器选项，请从命令行输入 **:set** 子命令。

要为*所有编辑会话*设置 vi 选项，将 **:set** 子命令置于 **.profile** 文件的 **EXINIT** 环境变量中（该文件由登录的 shell 读取）或将 **set** 子命令置于 **.exrc** 文件中。vi 编辑器首先查找 **EXINIT** 环境变量并运行其命令。如果 **EXINIT** 环境变量不存在，则 vi 编辑器查找 **$HOME/.exrc** 文件并运行其命令。最后，不管以前的结果如何，vi 编辑器查找本地 **.exrc** 文件并运行其命令。

注: 除了 **tvi** 命令（可信的 vi）外，该过程都成立。在此情况下，vi 编辑器查找并只运行 **/etc/.exrc** 文件。

关于通过设置 **EXINIT** 环境变量来更改选项的信息，请参阅 **environment** 文件中对于环境变量的描述。

.exrc 文件可以包含 **set** *Option*=*Value;* 格式的子命令，例如:

```
set cp=3 . . ;
```

要在 **.exrc** 文件中包含注释，请将 ″（双引号）作为该行的第一个字符。

定义宏: 如果经常使用一条子命令或一系列子命令，可以使用 vi 编辑器来定义发出子命令或子命令序列的宏。

要定义一个宏，请将子命令序列输入到以字母表中一个字母命名的缓冲区中。小写字母 a 到 z 覆盖缓冲区的内容，大写字母 A 到 Z 将文本附加到缓冲区先前的内容，这样可以逐块构建宏。

例如，要定义命名为 c 的宏，该宏搜索单词 corner 并使单词 corner 后的第三行成为当前行，请输入以下命令:

```
o /corner/+3
```

然后按 ESC 键并输入以下命令:

```
"c
```

其中 c 是缓冲区宏的名称。

要添加文本到已定义的缓冲区中以前的内容，请输入 `o viSubcommand`，按 ESC 键，并输入 `"CapitalLetter`，其中 *CapitalLetter* 变量指定一个从 A 到 Z 的大写字母。例如，要构建名为 T 的缓冲区宏，该宏搜索单词 corner 并允许添加更多命令，请输入以下命令:

```
o corner
```

然后按 ESC 键并输入以下命令:

```
"T
```

其中 T 是缓冲区宏的名称。任何时候都可以重复本过程将更多的 vi 子命令添加到相同的缓冲区。

例如，要添加移动光标到前面行并删除该行的命令，请输入以下命令:

```
o -dd
```

这里 -（负号）意为将光标上移一行，而 dd 表示删除当前行。按 ESC 键并输入以下命令:

```
"Tdd
```

要启动宏，请输入 `@ Letter`，其中 *Letter* 变量指定想要使用的缓冲区宏的字母名。要再使用相同的宏，请输入 `@@` (两个位于符号)。例如，输入 `@T` 启动 T 缓冲区宏并运行**搜索**、**移动光标**和**删除行**命令。输入 `@@T` 以再次启动 T 缓冲区宏。

系统使用的字符集通过整理表定义。该表影响 vi 宏的性能。

映射键: 可以使用 **:map**、**:map!** 和 **:ab** 子命令将击键操作映射到命令或命令序列。**:map** 子命令在命令方式中使用。**:map!** 和 **:ab** 子命令在文本输入方式中使用。可以为此编辑会话和所有以后的编辑会话映射键，或者仅为当前编辑会话（以两种方式中的一种）映射键。

要*为所有以后的编辑会话*映射键，请将子命令置于 **$HOME/.exrc** 文件中。vi 编辑器每次启动时会读取此文件。映射保持对每一次编辑会话有效。

要*仅从命令方式为当前的编辑会话*映射键，请在 vi 编辑器会话期间启动子命令。要只通过*文本输入方式*为当前编辑会话映射键，请在 vi 编辑器会话期间在命令行上输入子命令。映射保持对当前编辑会话有效。

警告: 如果使用的是 IBM 3161 ASCII 显示站、IBM 3163 ASCII 显示站或 IBM 3101 ASCII 显示站，vi 编辑器的缺省键映射会导致数据丢失。要查看缺省映射，请发出 **:map** 子命令。ESC-J 或 Shift-J 按键顺序产生特殊的问题。这些按键顺序删除从光标的当前位置到文件尾所有的信息。为了避免问题出现，请使用 **.exrc** 文件来更改此按键顺序。

:map、**:map!** 和 **:ab** 子命令按以下定义:

:map 在命令方式中定义宏。在 vi 编辑器中，**:map** 子命令允许通过按单一键来运行指定的命令或命令序列。

要在命令方式中映射键，启动带空的编辑缓冲区的 vi 编辑器而不使用 **vi** 命令命名 vi 文件或在 vi 编辑器启动后输入任何东西到缓冲区。可以使用 **:map** 子命令完成以下任务:

- 要将字符映射到编辑命令序列，请输入:

  ```
  :map Letter viSubcommand
  ```

- 要在命令方式中取消先前映射的字符，请输入:

  ```
  :unmap Letter
  ```

- 要为命令方式显示当前映射列表，请输入:

  ```
  :map
  ```

vi 编辑器不使用以下键，但是可以在命令方式中和 **:map** 子命令一起使用。

- 字母 s、g、K、q、V 和 v
- 控制键序列 Ctrl-A、Ctrl-K、Ctrl-O、Ctrl-T、Ctrl-W 和 Ctrl-X
- 符号 _（下划线），*（星号），\ (反斜杠) 和 =（等号）

尽管可以映射已被 vi 编辑器使用的键，但只要映射有效，就无法使用键的常用功能。有些终端允许将命令序列映射到功能键。如果处于 LISP 方式，=（等号）不能使用，因为它已由 vi 编辑器使用。

要映射字母 v 到命令序列，该命令序列找到下一个出现的 map 单词并将它改成单词 MAP，请输入以下命令:

```
:map v /map<Ctrl-V><Enter>cwMAP<Ctrl-V><Esc><Ctrl-V><Enter>
```

前例指令 vi 编辑器找到下一个 map（/map<Ctrl-V><Enter>），将 map 更改为 MAP（cwMAP），结束 change-word 子命令（<Ctrl-V><Esc>），并输入命令（<Ctrl-V><Enter>）。

注: 为了防止 vi 编辑器解释 Enter 键，当它被映射时，必须在前面输入 Ctrl-V 按键顺序。此情况对 Esc、Backspace 和 Delete 键也成立。

要映射控制字符 Ctrl-A、Ctrl-K 和 Ctrl-O，同时按 Ctrl 键和相应的字母。例如，要将 Ctrl-A 按键顺序映射到保存文件和编辑系列中下一个文件的命令序列，请输入以下命令:

```
:map <Ctrl-A> :w<Ctrl-V><Enter>:n<Ctrl-V><Enter>
```

要映射控制字符 Ctrl-T、Ctrl-W 和 Ctrl-X，必须首先用 Ctrl-V 按键顺序序为它们转义。

要映射 |（管道符），首先必须用两个 Ctrl-V 按键顺序为它转义，这通过以下示例说明：将字符 g 映射到命令序列，该命令序列转义到 shell，连接文件 **/etc/motd** 并输出到 **wc** 命令：

```
:map g :!cat /etc/motd <Ctrl-V><Ctrl-V>| wc<Ctrl-V><Enter>
```

如果终端允许映射功能键，必须用 *#number* 按键顺序引用它们，以指定想映射的功能键的号码。在下例中，F1 功能键映射到删除一个单词并将光标下移三个单词的命令序列：

```
:map #1 dwwww
```

为了使功能键映射起作用，功能键给终端类型的输出必须和 **terminfo** 文件中定义的输出相匹配。这些定义由 kf*number* 条目表示，其中，kf1 代表 F1 功能键，kf2 代表 F2 功能键，等等。如果当按功能键时所获取的输出与此条目不匹配，在任何映射可以发生之前，必须使用终端的设置方式去纠正设置以便和这些终端数据库条目相匹配。

也可以映射某些键盘特殊键，如 Home、End、Page Up 和 Page Down 键。对于大多数的终端，这些键已经在 vi 编辑器中映射。可以通过使用 **:map** 子命令验证此映射。如果这些键还没有映射，可以如下使用 **:map** 子命令：

```
:map <Ctrl-V><End> G
:map <Ctrl-V><Home> 1G
:map <Ctrl-V><PageUp> <Ctrl-F>
:map <Ctrl-V><PageDown> <Ctrl-B>
```

要在命令方式中获取所有当前映射列表，请输入 **:map** 子命令。则前例显示如下：

```
v         v          /map<Ctrl-M>cwMAP<Ctrl-[>Ctrl-M>
<Ctrl-A>  <Ctrl-A>   :w<Ctrl-M>:n<Ctrl-M>
g         g          :!cat /etc/motd | wc <Ctrl-M>
```

注：Ctrl-V 和 Enter 按键顺序显示为 Ctrl-M 按键顺序，而 Ctrl-V 和 Esc 按键顺序显示为 Ctrl-[按键顺序。

:map!

当处于文本输入方式时，将字符串映射到单个键。要在文本输入方式中映射键，启动带空的编辑缓冲区的 vi 编辑器而不使用 **vi** 命令命名 vi 文件或在 vi 编辑器启动后输入任何东西到缓冲区。可以使用 **:map!** 子命令完成以下任务：

- 要在文本输入方式中映射一个字母到一个或多个 vi 字符串，请输入：

  ```
  :map! Letter String
  ```

- 要在文本输入方式中取消先前映射的字母，请输入：

  ```
  :unmap! Letter
  ```

- 要显示在文本输入方式中已映射到特殊键的现有字符串列表，请输入：

  ```
  :map!
  ```

当处于文本输入方式时，输入映射键会生成指定的字符串。Ctrl-V 和 Esc 按键顺序进入命令方式，回到当前单词的开始（**bbw**），并启动 **cw**（change-word）子命令。例如：

```
:map! % <Ctrl-V><Esc>bbwcw
```

当输入文本时，如果意识到已经输错了一个单词，可以通过按 %（百分号）键并重新输入该单词。编辑器自动返回到插入方式。

注：当选择将用于 **:map!** 子命令的键时要特别当心。一旦键已映射，除非先发出 **:unmap!** 子命令，它们就不能再按文本输入。

:ab 将键或按键顺序映射到字符串以用于文本输入方式。当输入具有几个重复的短句、名称或标题时，**:ab** 子命令十分有用。

以下示例将 `city` 替换为短句 `Austin, Texas 78759`，只要它是用文本输入方式输入并且后面紧跟一个空格、句号或冒号:

```
:ab city Austin, Texas 78759
```

例如，如果在输入文本时，您输入以下内容:

```
My current residence is city.
```

按 Tab 键扩展单词 `city` 读入:

```
My current residence is Austin, Texas 78759.
```

缩写在单词中不扩展。例如，如果您输入 `My current residence iscity`，单词 iscity 不扩展。

如果 **:map!** 子命令用于映射用于插入方式的缩写，则出现的所有缩写都被扩展，而不管它出现在什么位置。如果在前例中使用 **:map!** 子命令（`:map! city Austin, Texas 78759`），那么无论什么时候输入单词 `city`，无论前面或后面是什么，该单词将扩展为 `Austin, Texas 78759`。因此，单词 `iscity` 成为 `isAustin, Texas 78759`。

注: 当选择用于 **:ab** 子命令的键时，需要小心。一旦键已定义，除非先发出 **:unab** 子命令，它们就不能再按文本输入。

设置缩写: **set** 命令具有与 **map!** 命令类似的行为，除了 **set** 命令仅当缩写是单个单词时以缩写形式代替字符串。可以使用 vi 编辑器的 **set** 命令完成以下任务:

- 列出现有的缩写
- 除去缩写
- 设置（定义）缩写

注: 用空编辑缓冲区启动 vi 编辑器。不要使用 **vi** 命令命名 vi 文件或在 vi 编辑器启动后输入任何东西到缓冲区。按 Esc 键以确保处于命令方式。

要列出缩写

输入 **:ab** 命令来列出现有的缩写。按 Enter 键返回到命令方式。

要除去缩写

输入 **:anab** *Abbreviation* 命令以除去缩写，其中 *Abbreviation* 变量指定不再想缩写的字符串。

要设置（定义）缩写

输入 **:ab** *Abbreviation String* 命令来设置缩写，其中 *Abbreviation* 变量指定定义为缩写的字符串，而 *String* 变量指定被缩写的字符串。 只有当缩写是单独的单词时，缩写可以被字符串替代。

例如， 如果输入 **:ab kn upper** 命令，然后当处于文本输入方式时输入 `acknowledge`，因为单词 acknowledge 中的 kn 字符串不是单独的单词，所以设置缩写字符串命令不启动。

然而，如果输入 **:ab kn upper** 命令，然后当处于文本输入方式时输入 `make the kn line all kncase`，结果是 `make the upper line all uppercase`。

标志

-c*Subcommand*	在开始用 **vi** 查看之前，执行 ex 编辑器子命令。光标移到被最近执行的命令影响的行。当输入一个空操作数时，如 **-c' '**，vi 编辑器将光标置于文件的第一行。**-c** 标志不能与 **+** 标志兼容。不要同时指定这两个标志。
-l	在 LISP 方式中进入 vi 编辑器。在此方式中，vi 编辑器适当地为 LISP 代码创建缩排，修改 **(**、**)**、**{**、**}**、**[[** 和 **]]** 子命令以便适当地为 LISP 进行操作。
-r[*File*]	在 vi 编辑器或系统故障之后恢复文件。如果不指定 *File* 变量， vi 编辑器显示所有保存过的文件列表。
-R	设置**只读**选项以保护文件使其不被覆盖。
-t*Tag*	编辑包含 *Tag* 变量的文件并将 vi 编辑器定位于它的定义。要使用该标志，必须首先使用 **ctags** 命令创建一个函数名及其位置的数据库。
-v	在详细方式中进入 vi 编辑器。
-w*Number*	将缺省窗口大小设置为 *Number* 变量指定的值。当在低速线路上使用 vi 编辑器时，此标志十分有用。
-y*Number*	用任何大于 1024 的值覆盖 1,048,560 的最大行设置。由于 vi 编辑器为进行缓冲区而控制使用额外的行，请求的行数应该是需要的两倍。
+[*Subcommand*]	在开始编辑之前，执行 ex 编辑器子命令。如果没有指定 *Subcommand* 变量，光标将置于文件的首行。此 **+** 标志与 **-c** 标志不兼容。不要同时指定这两个标志。

vi 常规子命令语法

使用以下常规语法来输入子命令:

[*Named_Buffer*] [*Operator*] [*Number*] *Object*

注: 方括号标识任选项。

[*Named_Buffer*]	指定临时文本存储区域。
[*Operator*]	指定子命令或操作；指导 vi 编辑器。
[*Number*]	将操作范围或行地址指定为一个整数。
Object	指定操作对象，如文本对象（字符、单词、句子、段落、节、字符串）或文本位置（一行、在当前行中的位置、屏幕位置）。

在子命令之前计数

可以在许多子命令前放一个数字。vi 编辑器以下面方式之一解释此数字:

- 转至由 *Number* 参数指定的行:

```
5G
10Z
```

- 转至由 *Number* 参数指定的列。

```
25|
```

- 上下滚动由 *Number* 参数指定的行数:

```
10Ctrl-U
10Ctrl-D
```

vi 编辑器子命令

使用子命令执行以下种类的操作:

- 移动光标
- 编辑文本

- 控制文件
- 其它操作

移动光标

按以下方式使用子命令在文件内移动光标:

- 在行内移动
- 按字符位置在行内移动
- 移到单词
- 按行位置移动
- 移到句、段落或节
- 通过重绘屏幕移动
- 翻页和滚动
- 搜索模式
- 在文件中标记特殊位置并返回

在行内移动: 在命令方式中输入以下子命令。可以通过按 Esc 键取消不完整的命令。如果需要关于 vi 子命令格式的信息，请参阅“vi 常规子命令语法”。

左箭头或 **h** 或 **Ctrl-H**	将光标左移一个字符。
下箭头或 **j** 或 **Ctrl-J** 或 **Ctrl-N**	将光标下移一行（它保持在同一列）。
上箭头或 **k** 或 **Ctrl-P**	将光标上移一行（它保持在同一列）。
右箭头或 **l**	将光标右移一个字符。

按字符位置在行中移动: 在命令方式中输入以下子命令。可以通过按 Esc 键取消不完整的命令。如果需要关于 vi 子命令格式的信息，请参阅“vi 常规子命令语法”。

^	将光标移到第一个非空字符。
0	将光标移到行的开始。
$	将光移到行尾。
f*x*	将光标移到下一个 *x* 字符。
F*x*	将光标移到最后一个 *x* 字符。
t*x*	将光标移到下一个 *x* 字符的前一列。
T*x*	将光标移到最后一个 *x*字符的后一列。
;	重复最近的 **f**、**F**、**t** 或 **T** 子命令。
,	以反方向重复最近的 **f**、**F**、**t** 或 **T** 子命令。
*Number***|**	将光标移到指定列。

移到单词: 在命令方式中输入以下子命令。如果需要关于 vi 子命令格式的信息，请参阅“vi 常规子命令语法”。

w	将光标移到下一个小字。
b	将光标移到前一个小字。
e	将光标移到小字的倒数第二个字母。
W	将光标移到下一个大字。
B	将光标移到前一个大字。
E	将光标移到大字的倒数第二个字母。

按行位置移动: 在命令方式中输入以下子命令。如果需要关于 vi 子命令格式的信息，请参阅“vi 常规子命令语法”。

H 将光标移到屏幕的顶行。
L 将光标移到屏幕的最后一行。
M 将光标移到屏幕的中间行。
+ 将光标移到它的第一个非空字符的下一行。
- 将光标移到它的第一个非空字符的前一行。
Enter 将光标移到它的第一个非空字符的下一行。

移到句、段落或节: 在命令方式中输入以下子命令。可以通过按 Esc 键取消不完整的子命令。如果需要关于 vi 子命令格式的信息，请参阅“vi 常规子命令语法”。

(将光标置于前一句的开始，或前一个 s 表达式（如果处于 LISP 方式）。
) 将光标置于下一句的开始，或下一个 s 表达式（如果处于 LISP 方式）。
{ 将光标置于前一个段落的开始，或下一个列表（如果处于 LISP 方式）。
} 将光标置于下一个段落的开始，如果处于 C 方式，则置于下一节，或者如果处于 LISP 方式则置于下一列表。
]] 将光标置于下一节，或者函数（如果处于 LISP 方式）。
[[将光标置于前一节，或者函数（如果处于 LISP 方式）。

通过重绘屏幕移动: 在命令方式中输入以下子命令。可以通过按 Esc 键取消不完整的子命令。如果需要关于 vi 子命令格式的信息，请参阅“vi 常规子命令语法”。

z 用屏幕顶部的当前行重绘屏幕。
z- 用屏幕底部的当前行重绘屏幕。
z. 用屏幕中间的当前行重绘屏幕。
/*Pattern***/z-** 用屏幕底部的包含由 *Pattern* 参数指定的字符串的行重绘屏幕。

翻页和滚动: 在命令方式中输入以下子命令。可以通过按 Esc 键取消不完整的子命令。如果需要关于 vi 子命令格式的信息，请参阅“vi 常规子命令语法”。

Ctrl-U 向上滚动 1/2 屏幕。
Ctrl-D 向下滚动 1/2 屏幕。
Ctrl-F 向前滚动一个屏幕。
Ctrl-B 向后滚动一个屏幕。
Ctrl-E 将窗口向下滚动一行。
Ctrl-Y 将窗口向上滚动一行。
z+ 向上翻页
z^ 向下翻页

搜索模式: 在命令方式中输入以下子命令。可以通过按 Esc 键取消不完整的子命令。如果需要关于 vi 子命令格式的信息，请参阅“vi 常规子命令语法”。

[*Number***]G** 将光标置于由 *Number* 参数指定的行号，或者最后一行（如果没有指定 *Number* 参数）。
/*Pattern* 将光标置于包含由 *Pattern* 参数指定的字符串的行的下一行。
?*Pattern* 将光标置于包含由 *Pattern* 参数指定的字符串的行的上一行。
n 在同一方向重复上次对 *Pattern* 参数指定的文本的搜索。
N 在相反方向重复上次对 *Pattern* 参数指定的文本的搜索。
/*Pattern***/+***Number* 将光标置于与由 *Pattern* 参数指定的字符串相匹配的行后面的指定行数。
?*Pattern***?-***Number* 将光标置于与由 *Pattern* 参数指定的字符串相匹配的行前面的指定行数。
% 查找与当前光标位置的括号或大括号相匹配的括号或大括号。

编辑文本

用于编辑的子命令使您能执行以下任务:

- 在文件中标记具体位置并返回
- 添加文本到文件
- 当处于输入方式时更改文本
- 从命令方式更改文本
- 复制和移动文本
- 恢复和重复更改

在文件中标记具体位置并返回: 在命令方式中输入以下子命令。可以通过按 Esc 键取消不完整的子命令。如果需要关于 vi 子命令格式的信息，请参阅“vi 常规子命令语法”。

″	将光标移到当前行的前一位置。
″	将光标移到包含当前行前一位置的行的开始。
m*x*	用 *x* 参数指定的字母标记当前位置。
`*x*	将光标移到由 *x* 参数指定的标记。
'*x*	将光标移到包含由 *x* 参数指定的标记的行的开始。

添加文本到文件（文本输入方式）: 在命令方式中输入以下子命令以使得 vi 编辑器切换进入文本输入方式。如果需要关于 vi 子命令格式的信息，请参阅“vi 常规子命令语法”。

a*Text*	在光标后面插入由 *Text* 参数指定的文本。按 Esc 键结束文本输入方式。
A*Text*	添加由 *Text* 参数指定的文本到行尾。按 Esc 键结束文本输入方式。
i*Text*	在光标前面插入由 *Text* 参数指定的文本。按 Esc 键结束文本输入方式。
I*Text*	在行中的第一个非空字符前插入由 *Text* 参数指定的文本。按 Esc 键结束文本输入方式。
o	在当前行的下面添加一个空行。按 Esc 键结束文本输入方式。
O	在当前行的上面添加一个空行。按 Esc 键结束文本输入方式。

当处于输入方式时更改文本: 仅当处于文本输入方式时使用以下子命令。这些命令在命令方式中有不同的意思。如果需要关于 vi 子命令格式的信息，请参阅“vi 常规子命令语法”。

Ctrl-D	回退到前一个自动缩排停止位。
^ Ctrl-D	只结束该行的自动缩排。
0Ctrl-D	将光标退回到左页边。
Esc	结束插入并返回到命令状态。
Ctrl-H	擦除上一个字符。
Ctrl-Q	如果 xon 禁用则输入任意字符。
Ctrl-V	输入任意字符
Ctrl-W	擦除上一个小字。
****	引用擦除和行删除符。
Ctrl-?	中断并结束插入或 Ctrl-D 按键顺序。

从命令方式更改文本: 在命令方式中使用以下子命令。可以通过按 Esc 键取消不完整的子命令。如果需要关于 vi 子命令格式的信息，请参阅“vi 常规子命令语法”。

C	更改行的剩余部分（同 **c$** 一样）。
cc	更改一行
cw	更改一个单词
cw*Text*	将字更改为 *Text* 参数指定的文本。
D	删除行的剩余部分（同 **d$** 一样）。

dd 删除一行。
dw 删除一个单词。
J 连接行。
r*x* 用 *x* 指定的字符替换当前字符。
R*Text* 用 *Text* 参数指定的文本覆盖字符。
s 替代字符（同 **cl** 一样）。
S 替代行（同 **cc** 一样）。
u 撤销上次更改。
x 删除光标位置的字符。
X 删除光标前的一个字符（同 **dh** 一样）。
<< 将一行向左移位。
<L 将从光标位置开始的所有行向左移到屏幕末尾。
>> 将一行向右移位。
>L 将从光标位置开始的所有行向右移到屏幕末尾。
~ 将光标位置的字母更改为相反的大小写格式。
! 缩排 LISP。

复制和移动文本: 在命令方式中使用以下子命令。可以通过按 Esc 键取消不完整的子命令。如果需要关于 vi 子命令格式的信息，请参阅“vi 常规子命令语法”。

p 将撤销缓冲区中的文本还原到光标之后。
P 将撤销缓冲区中的文本还原到光标之前。
″*x***p** 将 *x* 缓冲区中的文本还原
″*x***d** 将文本删除到 *x* 缓冲区。
y 将跟随的对象（例如 word 为**w**）放到撤销缓冲区。
″*x***y** 将跟随的对象放到 *x* 缓冲区，其中 *x* 为任意字母。
Y 将行放到撤销缓冲区。

恢复和重复更改: 在命令方式中使用以下子命令。可以通过按 Esc 键取消不完整的子命令。如果需要关于 vi 子命令格式的信息，请参阅“vi 常规子命令语法”。

u 撤销最后一次更改。

注: 撤销后，光标移到更新过的当前行的第一个非空字符。
U 如果最后一次更改以来光标还没有离开该行，则恢复当前行。
. 重复上次更改或增加 ″*n***p** 命令。

注:

1. 这条子命令将重复最后一次的更改，包括撤销命令。因此，一次撤销后，重复一次撤销而不是重复最后一次更改。
2. 这条子命令不是为和宏一起使用设计的。输入 **@@** (两个位于字符)以重复宏。

″*n* **p** 检索完整行或行块的倒数第 *n* 次删除。

操纵文件

用于操纵文件的子命令允许执行在以下各节中概述的任务:

- 将更改保存至文件
- 编辑第二个文件
- 编辑文件列表
- 查找文件信息

将更改保存至文件: 在命令方式中使用以下子命令。如果需要关于 vi 子命令格式的信息，请参阅“vi 常规子命令语法”。

:w 将编辑缓冲区内容写入原始文件中。如果正在 ex 编辑器中使用这条子命令，不必输入 :（冒号）。
:w *File* 将编辑缓冲区内容写入由 *File* 参数指定的文件中。如果正在 ex 编辑器中使用这条子命令，不必输入 :（冒号）。
:w! *File* 用缓冲区内容覆盖由 *File* 参数指定的文件。如果正在 ex 编辑器中使用这条子命令，不必输入 :（冒号）。

编辑另一文件: 在命令方式中输入以下子命令。如果需要关于 vi 子命令格式的信息，请参阅“vi 常规子命令语法”。

:e *File* 编辑指定文件。如果正在从 ex 编辑器使用这条子命令，不必输入 :（冒号）。
:e! 重新编辑当前文件并废弃所有更改。
:e + *File* 从最后开始编辑指定文件。
:e + *Number File* 从指定行号开始编辑指定文件。
:e # 编辑备用文件。在用 **:e** 命令访问另一个文件之前，备用文件通常是以前的文件名。但是，如果当调用一个新文件时更改在当前文件中处于暂挂状态，则该新文件成为备用文件。这条子命令和 **Ctrl-A** 子命令相同。
:r *File* 通过在当前行下面添加新行，将文件读取到编辑缓冲区。如果正在从 ex 编辑器使用这条子命令，不必输入 :（冒号）。
:r !*Command* 通过在当前光标位置下面添加新行，运行指定命令并将其输出放到文件中。
:ta *Tag* 从 *Tag* 标记符位置开始，编辑包含该*标记符*的文件。要使用这条子命令，必须首先使用 **ctags** 命令创建一个函数名及其位置的数据库。如果正在从 ex 编辑器使用这条子命令，不必输入 **:**（冒号）。
Ctrl-A 编辑备用文件。备用文件通常使用前一个当前文件名。但是，如果当调用一个新文件时更改在当前文件中处于暂挂状态，那么新文件成为备用文件。这条子命令和 **:e #** 子命令相同。

编辑文件列表: 在命令方式中输入以下子命令。如果需要关于 vi 子命令格式的信息，请参阅“vi 常规子命令语法”。

:n 编辑命令行上输入的列表中的下一个文件。如果正从 ex 编辑器使用这条子命令，则不需要 :（冒号）。
:n *Files* 指定一个要编辑的新文件列表。如果正从 ex 编辑器使用子命令，则不需要 **:**（冒号）。

查找文件信息: 在命令方式中输入以下子命令。如果需要关于 vi 子命令格式的信息，请参阅“vi 常规子命令语法”。

Ctrl-G 显示当前文件名、当前行号、文件中行数和光标定位于文件整体的百分比。

其它操作

vi 编辑器提供以下各节描述的子命令:

- 调整屏幕
- 输入 shell 命令
- 中断和结束 vi 编辑器

调整屏幕: 在命令方式中输入以下子命令。可以通过按 Esc 键取消不完整的子命令。如果需要关于 vi 子命令格式的信息，请参阅“vi 常规子命令语法”。

Ctrl-L 清除和重绘屏幕。

Ctrl-R 重绘屏幕并消除以 @ (位于符号)标记的空行。
z*Number* 使窗口长度能显示指定行数。

输入 shell 命令: 以下子命令允许在 vi 编辑器内运行命令。在命令方式中输入这些子命令。如果需要关于 vi 子命令格式的信息，请参阅 “vi 常规子命令语法”。

:sh 进入 shell 以便能运行多条命令。通过按 Ctrl-D 按键顺序可以返回到 vi 编辑器。如果正在 ex 编辑器中使用这条子命令，则不需要 :（冒号）。

:!*Command* 运行指定的命令然后返回到 vi 编辑器。如果正在 ex 编辑器中使用这条子命令，则不需要 :（冒号）。

注: 当跟着 **:!** 子命令时，#（备用文件）、**%**（当前文件）和 **!**（前面的命令）特殊字符都会扩展。要避免这些字符中的任意字符被扩展，可使用 ****（反斜杠)。

:!! 重复最近一次的 **:!***Command* 子命令。

Number **!!***Command* 运行指定的命令并用命令输出替代由 *Number* 指定的行。如果没有指定数目，则缺省值是 1。如果命令需要标准输入，则指定的行用作输入。

!*Object Command* 运行指定命令，并用命令输出替代由 *Object* 参数指定的对象。如果命令需要标准输入，则指定的对象用作输入。

中断和结束 vi 编辑器: 在命令方式中输入以下子命令。如果需要关于 vi 子命令格式的信息，请参阅 “vi 常规子命令语法”。

Q 在命令方式中进入 ex 编辑器。
ZZ 退出 vi 编辑器，保存更改。
:q 退出 vi 编辑器。如果更改了编辑缓冲区的内容，vi 编辑器显示一条警告消息并且不退出。如果正从 ex 编辑器使用这条子命令，则不需要 :（冒号）。
:q! 退出 vi 编辑器，废弃编辑缓冲区。如果正从 ex编辑器使用这条子命令，则不需要 :（冒号）。
Esc 结束文本输入或结束一条不完整的子命令。
Ctrl-? 中断一条子命令。

出口状态

返回以下出口值:

0 指示成功完成。
>0 指示发生错误。

输入文件

输入文件必须是文本文件或和文本文件相似的文件，除了长度不超过 LINE_MAX -1 字节并且不包含空字符的不完整的最近行。

.exrc 文件必须是由 **ex** 命令组成的文本文件。

缺省情况下，vi 编辑器从将要编辑的文件读取行，而不将这些行中任意一行解释成任何格式的 vi 编辑器命令。

相关信息

ctags 命令、**ed** 命令、**ex** 命令、**sed** 命令、**tvi** 命令、**view** 命令。

.profile 文件。

view 命令

用途

以只读方式启动 vi 编辑器。

语法

view [**-c***Subcommand*] [**-l**] [**-t** *Tag*] [**-w***Number*] [**-y**] [**-r** [*File*]] [**+** [*Subcommand*]] [*File* ...]

描述

view 命令以只读方式启动 vi 全屏编辑器。仅在防止对文件做意外的更改时建议使用只读方式。要覆盖只读方式，当执行一条命令时，请使用 !（感叹号）。*File* 参数指定想要浏览的文件名。使用 vi 子命令在文件内移动。使用 **:q** 子命令退出 **view** 命令。如果修改了文件，可通过按 Esc 键和 `wq!` 保存修改。

标志

-c *子命令*	在开始用 vi 查看之前，执行 ex 编辑器子命令。当输入一个空操作数时，如 `-c ''`，则编辑器将光标置于文件最后一行上。
-l	输入包含为了用 LISP 语言编写程序设计的特定功能部件的 vi 编辑器版本。在此方式中，vi 编辑器适当地为 LISP 编程进行缩排，**(**，**)**，**{**，**}**，**[[** 和 **]]** 子命令被修改以便适当地为 LISP 进行操作。
-r [*File*]	在编辑器或系统崩溃后恢复文件。如果没有指定 *File* 参数，编辑器显示所有已保存文件的列表。
-t *Tag*	编辑包含由 *Tag* 参数指定的标记的文件，并将编辑器置于它的定义位置。要使用该标志，必须首先使用 **ctags** 命令创建一个函数名及它们位置的数据库。
-w *Number*	将缺省窗口大小设置为 *Number* 参数指定的值。当终端与正在慢速通信线路上运行编辑器的系统通信时，这很有用。
-y	用任何大于 1024 的值覆盖 1,048,560 的最大行设置。
+[*Subcommand*]	在开始用 vi 查看之前，执行 *Subcommand* 参数指定的 ex 编辑器子命令。如果您没有指定子命令，则光标被置于文件的最后一行。

相关信息

vi 命令、**ctags** 命令。

vmh 命令

用途

为和 MH 命令一起使用启动可视界面。

语法

vmh [**-prompt** *String*] [**-vmhproc** *CommandString* | **-novmhproc**]

描述

vmh 命令启动可视界面以便和 MH 命令一起使用。**vmh** 命令实现 MH 窗口管理协议的服务器端，并为任何实现协议客户机端的程序保持分屏界面。

vmh 命令提示输入命令并将它们发送给协议的客户机端。如果命令产生多个输出屏幕的窗口，则 **vmh** 命令提示用户输入一条子命令。使用 **vmh** 子命令可以显示命令输出的特定部分。

vmh 子命令

Ctrl-L	刷新屏幕。
空格	前进到下一个屏幕。
[*Number*] **Enter 键**	前进指定行数。缺省值为一行。
[*Number*] **d**	前进指定行数的 10 倍。*Number* 变量的缺省值为 1，总共 10 行。
[*Number*] **g**	到指定行。
[*Number*] **G**	到窗口的末尾。如果指定 *Number* 变量，该命令的作用和 **g** 标志的作用一样。
[*Number*] **u**	退回指定行数的 10 倍。*Number* 变量的缺省值为 1，总共 10 行。
[*Number*] **y**	退回指定行数。缺省值为一行。
h	显示一条帮助消息。
q	结束输出

标志

-help	列出命令语法、可用的命令开关（切换开关）和版本信息。 **注**: 对于 MH，此标志的名称必须完全拼写。
-novmhproc	运行缺省值 **vmproc** 而不使用窗口管理协议。
-prompt *String*	用指定的字符串作为提示符。
-vmhproc *CommandString*	指定实现窗口管理协议的客户机端的程序。缺省为 **msh** 程序。

概要文件条目

下面是在 *UserMhDirectory***/.mh_profile** 文件中输入的条目:

`Path:`	指定用户的 MH 目录。
`mshproc:`	为 MH shell 指定程序。

文件

$HOME/.mh_profile	包含 MH 用户概要文件。
/usr/bin/vmh	包含 **vmh** 命令。

相关信息

msh 命令。

mh_alias 文件格式、**mh_profile** 文件格式。

*《AIX 5L V5.2 系统用户指南: 通信与网络》*中的『邮件概述』。

---

vmo 命令

用途

管理虚拟内存管理器可调参数。

语法

vmo [**-p** | **-r**] { **-o** *Tunable* [**=** *Newvalue*]}

vmo [**-p** | **-r**] {**-d** *Tunable* }

vmo [**-p** | **-r**] **-D**

vmo [**-p** | **-r**] **-a**

vmo -?

vmo -h [*Tunable*]

vmo -L [*Tunable*]

vmo -x [*Tunable*]

注: 允许多个 **-o**、**-d**、**-x** 和 **-L**。

描述

注: **vmo** 命令只能由 root 用户执行。

使用 **vmo** 命令来配置虚拟内存管理器调整参数。此命令设置或显示所有虚拟内存管理器调整参数的当前值或下一个引导值。还可以用此命令进行永久更改，或将更改推迟到下一次重新引导之后生效。此命令是设置参数还是显示参数，要由所带标志来决定。带 **-o** 标志的话，两个操作都执行。它既可以显示参数的值，也可以为参数设置新值。

虚拟内存管理器（VMM）维护一个空闲实存页面帧的列表。这些页面帧可用于维持满足一个缺页故障所需的虚拟内存页。当空闲列表上的页面数低于 minfree 参数所指定的数值时， VMM 开始取走页面以加入空闲列表。VMM 继续取走页面直到空闲列表至少达到 maxfree 参数所指定的页面数。

如果内存中的文件页面（永久页面）数少于 minperm% 参数所指定的数值，VMM 就会从计算页面或者文件页面取走帧，而不管重页率。如果文件页面数大于 maxperm% 参数所指定的数值，VMM 就只从文件页面中取走帧。如果介于二者之间，正常情况下 VMM 值取走文件页面，但是如果文件页面的重页率大于计算页面的重页率，计算页面也会被取走。

也可以修改用于决定系统何时耗尽页面空间的阈值。npswarn 参数指定调页空间可用页数值，从该数值系统开始警告进程调页空间不足。npskill 参数指定调页空间可用页面数，从该数值系统开始杀死进程以释放页面空间。

警告: 误用此命令会导致性能退化或操作系统失败。在试验操作 **vmo** 前，应该充分熟悉 Performance Overview of the Virtual Memory Manager (VMM) 和 Tuning VMM Page Replacement。

标志

-h [*Tunable*]
如果指定了一个，则显示关于 *Tunable* 参数的帮助。否则，显示 **vmo** 命令使用语句。

-a
为所有可调参数显示当前、重新引导（当与 **-r** 一起使用时）或永久（当与 **-p** 一起使用时）值，每行都是成对的 *Tunable* **=** ***Value***。对于永久选项，如果一个参数的重新引导值和当前值相等，则只显示一个值。否则，显示 NONE 作为其值。

-d *Tunable*
将 *Tunable* 复位为缺省值。如果 *Tunable* 需要更改（也就是说，它当前还没有被设置成缺省值），并且它属于类型 **Bosboot** 或 **Reboot**，或者如果它属于类型 **Incremental** 并且已经更改其缺省值，而且 **-r** 没有组合使用，那么它将不变而只显示警告。

-D 将所有的可调参数复位为缺省值。如果需要更改的可调参数属于类型 **Bosboot** 或 **Reboot**，或属于类型 Incremental 并且已经更改其缺省值，而且 **-r** 没有组合使用，那么它们将不变而只显示警告。

-o *Tunable*[**=***Newvalue*] 显示值或将可调参数设置为新值。如果一个可调参数需要更改（指定值与当前值不同），并且它属于类型 **Bosboot** 或 **Reboot**，或者如果它属于类型 **Incremental** 并且其当前值比指定值大，而且 **-r** 不组合使用，那么它将不变而只显示警告。

当与 **-r** 一起使用而没有指定新值时，则显示可调参数的下一个引导值。当与 **-p** 一起使用而没有指定新值时，则只有在可调参数的当前值和下一个引导值相同时，才显示值。否则，显示 NONE 作为其值。

-p 当与 **-o**、**-d** 或 **-D** 组合使用时，将更改应用到当前值和引导值，也就是说，除了打开当前值的更新，还打开 **/etc/tunables/nextboot** 文件的更新。这些组合不能用于 **Reboot** 和 **Bosboot** 类型参数，因为它们的当前值不能更改。

当与 **-a** 或 **-o** 组合使用而没有指定新值时，则只有在一个参数的当前值与下一个引导值相同时才会显示参数值。否则，显示 NONE 作为其值。

-r 当与 **-o**、**-d** 或 **-D** 组合使用时，将更改应用到引导值，例如，打开 **/etc/tunables/nextboot** 文件的更新。如果对类型 **Bosboot** 的任何参数做了更改，则提醒用户要运行 bosboot。

当与 **-a** 或 **-o** 一起使用而不指定新值时，则显示可调参数的下一个引导值而不是当前值。

-L [*Tunable*] 列出一个或所有可调参数的特征，每个一行，使用以下格式:

```
NAME                      CUR    DEF    BOOT   MIN    MAX    UNIT           TYPE
     DEPENDENCIES
--------------------------------------------------------------------------------
memory_frames             128K          128K                 4KB pages         S
--------------------------------------------------------------------------------
maxfree                   128    128    130    16     200K   4KB pages         D
     minfree     memory_frames
--------------------------------------------------------------------------------
minfree                   120    120    122    8      200K   4KB pages         D
     maxfree     memory_frames
--------------------------------------------------------------------------------
...
其中:     CUR = 当前值
    DEF = 缺省值
    BOOT = 重新引导值
    MIN = 最小值
    MAX = 最大值
    UNIT = 可调计量单位
    TYPE = 参数类型: D（指 Dynamic）、S（指 Static）、R（指 Reboot）、
           B（指 Bosboot）、M（指 Mount）、I（指 Incremental）和 C（指 Connect）
    DEPENDENCIES = 从属可调参数的列表，每个一行
```

-x [*Tunable*] 列出一个或所有可调参数的特征，每个一行，使用以下（电子表格）格式:

```
tunable,current,default,reboot,min,max,unit,type,{dtunable }

其中:     current = 当前值
    default = 缺省值
    reboot = 重新引导值
    min = 最小值
    max = 最大值
    unit = 可调计量单位
    type = 参数类型: D（指 Dynamic）、S（指 Static）、R（指 Reboot）、B（指 Bosboot）
               M（指 Mount）、I（指 Incremental）和 C（指 Connect）
    dtunable = 从属可调参数的列表
```

-? 显示 **vmo** 命令用法语句。

任何对 Mount 类型的参数的更改（带 **-o**、**-d** 或 **-D**）将产生一个显示信息，警告用户该更改仅对将来的安装有效。

任何对 Connect 类型的参数的更改（带 **-o**、**-d** 或 **-D** 标识）将导致重新启动 **inetd**，并显示一个信息，警告用户更改仅对将来的套接字连接有效。

任何试图在不使用 **-r** 的情况下对 **Bosboot** 或 **Reboot** 类型的参数的更改（带 **-o**、**-d** 或 **-D**），都会导致错误消息。

任何试图将 Incremental 类型参数的当前值更改为比它小的新值（带 **-o**、 **-d** 或 **-D**，但不带 **-r**）都会导致错误消息。

可调参数类型

所有通过调整命令（**no**、**nfso**、**vmo**、**ioo** 和 **schedo**）操纵的可调参数已经被分为这些类别:

Dynamic	如果该参数可在任何时间更改
Static	如果该参数决不可更改
Reboot	如果该参数仅可在重新引导期间更改
Bosboot	如果该参数仅可通过运行 bosboot 并重新引导机器来更改
Mount	如果参数的更改仅对将来文件系统或目录安装有效
Incremental	如果该参数仅可增量（除了引导时间之外）
Connect	如果参数的更改仅对将来的套接字连接有效

对于 Bosboot 类型的参数，无论何时执行更改，调整命令会自动提示用户，问他们是否想执行 **bosboot** 命令。对于 Connect 类型的参数，调整命令会自动重新启动 **inetd** 守护程序。

注意由 **vmo** 命令管理的当前参数集合仅包含 Static、Dynamic 和 Bosboot 类型。

兼容性方式

当在 AIX 5.2 以前版本的兼容性方式（由 **sys**0 的 **pre520tune** 属性控制）中运行时，重新引导参数值（除了那些 Bosboot 类型）并不真正完全有意义，因为在该方式下不会在引导时间应用这些参数。关于更多信息，请参阅*《AIX 5L V5.2 性能管理指南》*中的『AIX 5.2 调整增强』。

在 AIX 5.2 以前版本的兼容性方式中，通过在引导序列期间在所调用的脚本中嵌入调用调整命令，来继续完成对调整参数的重新引导值的设置。所以 **Reboot** 类型的参数可以不使用 **-r** 标志来设置，这样现有的脚本也可以继续工作。

当机器迁移到 AIX 5L V5.2 时，该方式自动打开。对于完全安装，该方式关闭且参数的重新引导值通过在重新引导序列过程中应用 **/etc/tunables/nextboot** 文件的内容设置。只有在该方式下，**-r** 和 **-p** 标志才能完全发挥作用。有关更多信息，请参阅 *AIX 5L Version 5.2 Performance Tools Guide and Reference* 中的 Kernel Tuning。

可调参数

defps

用途: 打开／关闭延迟页面空间分配（DPSA）策略。

值: 缺省值: 1

范围: 0 或 1（DPSA 已打开）

类型: Dynamic

诊断: 不适用

调整 如果担心页面空间过量使用，那么关闭 DPSA 策略可能有用。获取关于减少调页空间需求的值。

请参阅:

用 vmtune 命令在 LPAS 和 DPSA 之间选择

framesets

用途: 指定每个内存池的实内存页集合数。在 UP 内核中该参数不存在。

值:

缺省值: 2

范围: 1 到 10

类型: Bosboot

诊断: 不适用

调整 不适用

lgpg_regions

用途: 指定保留用以执行带有 SHM_LGPAGE 标志的 **shmget( )** 系统调用的大型页面的数目。在运行在基于非 Power4 机器上的 64 位内核中该参数不存在。

值:

缺省值: 0

范围: 0 - 页数。

类型: Bosboot

诊断: 不适用

调整 除了此选项外还必须使用 lpgpg_size。必须修改应用程序以便调用 **shmget()** 时指定 SHM_LGPAGE 标志。在有许多 TLB 不命中和大量内存被访问的情况下，这将会提高性能。

参考: 其它 I/O 调整参数

lgpg_size

用途: 指定用于实现带 SHM_LGPAGE 标志的 *shmget()* 系统调用的、硬件支持的大型页面的大小（用字节表示）。在运行在基于非 Power4 机器上的 64 位内核中该参数不存在。

值:

缺省值: 0

范围: 0 或 268435456（在非 POWER4 上），或者是 0 或 16777216（在 POWER4 上）。

类型: Bosboot

诊断: 不适用

调整 除了本参数之外，lgpg_region 必须设置成非零值。必须修改应用程序以便调用 **shmget()** 时指定 SHM_LGPAGE 标志。在有许多 TLB 不命中和大量内存被访问的情况下，这将会提高性能。

请参阅:

其它 I/O 调整参数

lrubucket

用途: 指定每个存储区的内存帧数。页替换算法将实存划分为帧存储区。在有多个内存池的系统上，lrubucket 参数是每个内存池帧数。

值:

缺省值: 131072 帧

范围: 65536 到内存帧的总数目

类型: Dynamic

诊断: 不适用

调整 大多数的系统上不推荐调整此参数。页替换算法不是扫描系统中的每一个帧搜索自由帧，而是扫描整个存储区内容，并且在继续进行到第二个存储区之前对同一存储区扫描第二遍。

请参阅:

调整 lrubucket 以减少内存扫描开销

maxclient%

用途: 指定可以用于对客户机页面进行高速缓存的 RAM 最大百分比。与 maxperm% 类似，但是不能大于 maxperm%。

值:

缺省值: 80

范围: 1% 到 100%。

类型: Dynamic

诊断: 如果 J2 文件页面或 NFS 页面正在促使工作的存储页面被调出，那么可以减少 maxclient。

调整 如果由于太多的 J2 客户机页面或 NFS 客户机页面在内存中，产生调出到调页空间的情况，则减少 maxclient 的值。增加值会允许更多的 J2 或 NFS 客户机页面在页替换启动之前放入内存。

参考: 其它 I/O 调整参数

maxfree

用途: 指定页面取走停止时的空闲列表上的帧数。

值:

缺省值: 128

范围: 16 到 204800

类型: Dynamic

诊断: 用 **vmstat n** 观察空闲列表大小的更改。

调整 如果 **vmstat n** 显示应用程序的要求频繁驱使空闲列表大小低于 minfree，则增加 maxfree 以减少补充空闲列表的调用。通常，使 maxfree - minfree 等于或小于 100。将值设置太高会使页替换运行更长时间。值必须比 minfree 至少大 8。

请参阅:

用 vmtune 命令调整 VMM 页替换

maxperm%

用途: 指定页面取走算法只取走文件页面的临界点。

值:

缺省值: 内存帧的总数目 * 0.8

范围: 1 到 100

类型: Dynamic

诊断: 用 **iostat n** 监视磁盘 I/O。

调整 此值表示为系统中全部实存页面帧的一个百分比。减小此值可以减少或消除因大量文件页面访问所造成的工作存储页的页替换。增加此值可以帮助 NFS 服务器（大部分是只读的）。例如，如果有些文件被认为是反复读取的，从启动开始 I/O 速率不随时间减少，则 maxperm 可能太低。

请参阅:

用 vmtune 命令调整 VMM 页替换

maxpin%

用途: 指定能被固定的实存的最大百分比。

值:

缺省值: 80%

范围: 1 到 99

类型: Dynamic

诊断: 不能固定内存，尽管自由内存可用。

调整 如果更改该值，新值应该确保至少 4 MB 实存没有被固定，用来供内核使用。maxpin 值必须大于 1 小于 100。仅在极端情况下更改此参数，比如标准程序达到最大负载时。

memory_affinity

用途: 该参数可用来指示 VMM（如果可能的话）在正在运行执行线程的同一个 MCM 中分配内存帧。此参数只能启用内存亲缘关系性，然后可通过将进程的 **MEMORY_AFFINITY** 环境变量设置为 MCM，可以为一个给定的进程打开这种内存亲缘性。仅在基于 Power4 的机器上支持该参数。

值:

- 缺省值: 1（已启用）
- 范围: 0 或 1（已禁用或已启用）
- 类型: Bosboot

诊断: 不适用

调整 如果受控，可重复的标准程序在执行时将显示出意想不到的可变性，这使 memory_affinity 有益于性能。

请参阅:

使用 AIX 内存亲缘性支持

mempools

用途: 更改内存池的数目（将在系统引导时配置）。在 UP 内核中该参数不存在。

值:

缺省值: CPU／8 的数目

范围: 1 到 CPU 的数目

类型: Bosboot

诊断: 不适用

调整 值 0 恢复内存池的缺省数目。不允许对 UP 内核进行更改，并且不推荐用于其它的内核。

请参阅:

调整内存池

minfree

用途: 指定空闲列表中 VMM 开始取走页面来补充空闲列表的最小帧数目。

值:

缺省值: maxfree - 8

范围: 8 到 204800

类型: Dynamic

诊断: vmstat n

调整 当自由帧的数目达到 minfree 时，发生页替换。如果进程由于页面被取走而被延迟，则增加 minfree 来改进响应时间。minfree 和 maxfree 之差应该始终等于或大于 maxpgahead。

请参阅:

用 vmtune 命令调整 VMM 页替换

minperm%

用途: 按内存帧总数的百分比指定一个点，在该点之下，页面调度程序将会取走文件或者计算页面而不管重调页率为多少。

值:

缺省值: 20%

范围: 1 到 100。

类型: Dynamic

诊断: 用 **iostat n** 监视磁盘 I/O。

调整 如果内存中大量文件页面使工作存储页面被替换，则减少此参数可能有用。如果有些文件被认为是反复读取的，从启动开始 I/O 速率不随时间减少，则 minperm 可能太低。

请参阅:

用 vmtune 命令调整 VMM 页替换

nokilluid

用途: 在缺乏调页空间的情况下不杀死低于此值的用户标识。

值:

缺省值: 0（关闭）

范围: 任何正整数。

类型: Dynamic

诊断: 不适用

调整 缺乏调页空间的系统和系统管理员进程将会被杀死。设置为 1 以防止特殊用户标识进程由于缺乏页面空间或为确保有足够的调页空间可用而被杀死。

请参阅:

选择 npswarn 和 npskill 设置

npskill

用途: 指定自由调页空间页数，据此操作系统开始杀死进程。

值:

缺省值: MAX（64，调页空间页面的数目／128）

范围: 0 到系统上调页空间页面总数

类型: Dynamic

诊断: 不适用

调整 如果感到进程由于缺乏调页空间正在被杀死，可以增大这一值。

请参阅:

选择 npswarn 和 npskill 设置

npswarn

用途: 指定自由调页空间页数，据此操作系统开始发送 SIGDANGER 信号到进程。

值:

缺省值: MAX（512，4*npskill）

范围: 0 到系统上调页空间页面总数。

类型: Dynamic

诊断: 不适用

调整 如果感到进程由于缺乏调页空间正在被杀死，可以增大此值。

请参阅:

选择 npswarn 和 npskill 设置

num_spec_dataseg

用途: 保留特殊数据段标识供与环境变量 DATA_SEG_SPECIAL=Y 一起的进程执行时使用。分配这些数据段以便这些段中的硬件页表条目更好地在高速缓存中分布以减少高速缓存冲突。在满足请求数目条件下，尽可能多地保留。重新引导后运行 **vmo -a** 显示保留的实际数目。仅在运行在基于 Power4 机器上的 64 位内核中支持该参数。

值:

缺省值: 0

范围: 0 或正数

类型: Bosboot

诊断: 不适用

调整 要保留的正确数目取决于与 DATA_SEG_SPECIAL=Y 同时运行的进程数目和每个进程所使用的数据段数。

pagecoloring

用途: 打开或关闭 VMM 中的页面着色。在 64 位内核中不支持该参数。

值:

缺省值: 0（关闭）

范围: 0 或 1。

类型: Bosboot

诊断: 不适用

调整 此参数对于在有直接映射高速缓存的机器上运行的一些应用程序有用。

pta_balance_threshold

用途: 指定一个点，新的 pta 段将分配在该点。在 64 位内核中该参数不存在。

值:

缺省值: pta 段大小 * 0.5

范围: 1 到 99。

类型: Dynamic

诊断: 由于 **dsi** 系统将会崩溃（异常结束代码 300），其堆栈类似于以下:

```
findsrval64()
shmforkws64()
shmforkws()
procdup()
kforkx()
kfork()
```

转储调查将会显示，pta 段对于生成缺页故障的页面是满的。

调整 将 pat 平衡阈值调低将促使新的 pta 段提前分配，因此减少了 pta 段填满并使得系统崩溃的机会。如果可能，更好的解决方案是移至没有此潜在问题的 64 位内核。

spec_dataseg_int

用途: 修改使用 num_spec_dataseg 保留的特殊数据段标识之间的间隔。仅在运行基于 Power4 机器上的 64 位内核中支持该参数。

值:

缺省值: 512

范围: 1 到任何正整数

类型: Bosboot

诊断: 不适用

调整 通常，对于带 DATA_SEG_SPECIAL=Y 执行的进程，要优化性能，所有进程都访问的数据段页面越多，则该值应当越高。然而，过高的值会限制可以保留的特殊段标识的数目。性能影响高度依赖于硬件体系结构、应用程序行为，而且对于不同体系结构和不同应用程序最佳值可能不同。

strict_maxperm

用途: 如果设置为 1，maxperm 值对于可用作持久文件高速缓存的 RAM 的数量是一个硬性限制。

值:

缺省值: 0（关闭）

范围: 0 或 1。

类型: Dynamic

诊断: 由于 RAM 中文件页面太多会导致过多的页面调出至页面空间。

调整 设置为 1 以使 maxperm 值成为一个硬性限制（和 masperm 参数的调整联合使用）。

参考: 用 strict_maxperm 对持久文件高速缓存设置硬性限制

v_pinshm

用途: 如果设置为 1，将允许对共享内存段的固定。

值:

缺省值: 0（关闭）

范围: 0 或 1。

类型: Dynamic

诊断: 从共享内存段对 AIO 缓冲区固定或解固定的开销太大的情况下更改此值。

调整 仅当进行 **shmget** 调用时应用程序也设置 SHM_PIN 标志，或当从共享内存段进行 I/O 同步时，此选项才有用。

参考: 其它 I/O 调整参数

示例

1. 要列出所有由 **vmo** 命令管理的所有可调参数的当前和重新引导值、范围、单元、类型和相关性，请输入:

   ```
   vmo -L
   ```

2. 要在 POWER4 系统中打开或保留 16MB 大型页，请输入:

   ```
   vmo -r -o lgpg_regions=10 -o lgpg_size=16777216
   ```

   该命令将向用户建议 **bosboot**， 并警告在更改生效之前必须重新引导。

3. 要显示 **nokilluid** 的帮助，请输入:

   ```
   vmo -h nokilluid
   ```

4. 要在下一次重新引导后打开 **v_pinshm**，请输入:

   ```
   vmo -r -o v_pinshm=1
   ```

5. 要将所有 **vmo** 可调参数永久复位为缺省值，请输入:

   ```
   vmo -p -D
   ```

6. 要列出所有虚拟内存管理器可调参数的重新引导值，请输入:

   ```
   vmo -r -a
   ```

7. 要列出（电子表格格式）由 **vmo** 命令管理的所有可调参数的当前和重新引导值、范围、单元、类型和相关性，请输入:

   ```
   vmo -x
   ```

相关信息

ioo 命令、**schedo** 命令、**no** 命令、**nfso** 命令、**tunchange** 命令、**tunsave** 命令、**tunrestore** 命令、**tuncheck** 命令和 **tundefault** 命令。

『虚拟内存管理器（VMM）性能概述』 *《AIX 5L V5.2 性能管理指南》*

AIX 5L Version 5.2 Performance Tools Guide and Reference 中的 Kernel Tuning。

*《AIX 5L V5.2 性能管理指南》*中的『AIX 5.2 调整增强』。

---

vmstat 命令

用途

报告虚拟内存统计信息。

语法

vmstat [**-f**] [**-i**] [**-s**] [**-l**] [**-t**] [**-v**] [*PhysicalVolume* ...] [*Interval* [*Count*]]

描述

vmstat 命令报告关于内核线程、虚拟内存、磁盘、陷阱和 CPU 活动的统计信息。由 **vmstat** 命令生成的报告可以用于平衡系统负载活动。系统范围内的这些统计信息（所有的处理器中）求和平均，以百分比表达，或者计算总和。

如果调用 **vmstat** 命令时不带标志，则报告包含系统启动后虚拟内存活动的摘要。如果指定 **-f** 标志，则 **vmstat** 命令报告自从系统启动后派生的数量。*PhysicalVolume* 参数指定物理卷的名称。

Interval 参数指定每个报告之间的时间量（以秒计）。第一个报告包含系统启动后时间的统计信息。后续报告包含自从前一个报告起的时间间隔过程中所收集的统计信息。如果没有指定 *Interval* 参数，**vmstat** 命令生成单个报告然后退出。*Count* 参数只能和 *Interval* 参数一起指定。如果指定了 *Count* 参数，其值决定生成的报告数目和各自的秒数。如果 *Interval* 参数被指定而没有 *Count* 参数，则连续生成报告。一个为 0 的 *Count*参数是不容许的。

AIX 4.3.3 及其更新版本包含方法的增强，该方法是用于计算 CPU 等待磁盘 I/O 所花时间（*wio* 时间）的百分数。在某些的环境下，用于 AIX 4.3.2 和该操作系统的较早版本中的方法会给出 SMP 上 wio 时间的膨胀了的视图。

AIX 4.3.2 和较早版本中使用的方法如下：在每个处理器的每一次时钟中断（每个处理器一秒钟 100 次），确定四种类别（usr/sys/wio/idle）中的哪一种可以用于最后 10 毫秒时间。如果在时钟中断的时候，CPU 正忙于用户方式，那么用户可以获取被添加到其类别中的时钟滴答。如果在时钟中断的时候，CPU 正忙于内核方式，那

么系统类别获取滴答。如果 CPU 不忙的话，则检测是否在进行磁盘 I/O。如果在进行任何的磁盘 I/ O，则增大 wio 类别。如果没有进行磁盘 I/O 而且 CPU 不忙，则空闲类别获取滴答。wio 时间的膨张视图产生于所有被归类为 wio 的空闲 CPU，而不管等待 I/O 的线程数目。例如，只有一个 I/O 线程的系统可以报告 90% 以上的 wio 时间，而不管它所有的 CPU 数目。wio 时间由命令 **sar**（*%wio*）、**vmstat**（*wa*）和 **iostat**（*%iowait*）报告。

用于操作系统 AIX 4.3.3 及其更新版本的方法如下：如果一个未完成的 I/O 在空闲的 CPU 上启动，则操作系统 AIX 4.3.3 中的更改将只把该 CPU 标记成 wio。当只有少量线程在进行 I/O 而系统其它部分是空闲的，此方法能报告低得多的 wio 时间。例如，一个具有四个 CPU 和一个正在进行 I/O 的线程的系统将最多报告 25% 的 wio 时间。具有 12 个 CPU 和一个正在进行 I/O 的线程的系统只报告最大为 8% 的 wio 时间。NFS 客户机通过 VMM 读取／写入，biods 在 VMM 中花费的等待 I/O 完成的时间现在被报告为 I/O 等待时间。

内核为内核线程、调页和中断活动维护统计信息，**vmstat** 命令通过使用**perfstat** 内核扩展来对其进行访问。磁盘输入／输出统计信息由设备驱动器维护。对于磁盘，平均传递速率由使用活动时间和传替信息数目来决定。活动时间的百分数根据报告期间驱动器忙的时间量来计算。

由 **vmstat** 命令生成的以下报告示例包含栏标题及其描述：

`kthr`：内核线程状态在采样间隔期间每秒钟更改一次。

`r`　置于运行队列中的内核线程数目。
`b`　置于等待队列（等待资源、等待输入／输出）的内核线程数目。

内存：关于使用虚拟内存和实内存的信息。如果虚拟页已经被访问的话，虚拟页可以被认为是活动的。一页为 4096 个字节。

`avm`　活动虚拟页。
`fre`　空闲列表的大小。

注：大部分实内存用作文件系统数据的高速缓存。对于保持较小的空闲列表，这是很正常的。

页：关于缺页故障和调页活动的信息。这些是间隔的平均值，以秒为单位给出。

`re`　页面调度程序输入／输出列表。
`pi`　从调页空间调度进的页面。
`po`　调出到调页空间的页面。
`fr`　释放的页（页面替换）。
`sr`　通过页替换算法扫描的页面。
`cy`　按页替换算法的时钟周期。

故障：采样间隔平均每秒的捕获和中断率。

`in`　设备中断
`sy`　系统调用。
`cs`　内核线程上下文切换。

`Cpu`：CPU 使用时间故障百分比。

`us`　用户时间。
`sy`　系统时间。
`id`　CPU 空闲时间。
`wa`　CPU 空闲时间，在此期间系统有未完成的磁盘/NFS I/O 请求。请参阅上面的详细描述。

磁盘：每秒向指定物理卷提供的传送数目，该过程在采样间隔中发生。*PhysicalVolume* 参数可以用于指定一到四个名称。每个指定驱动器的传送统计信息按指定顺序给出。该计数代表向物理设备的请求数。它并不暗示读取或写入的数据量。几个逻辑请求可以组合成为一个物理请求。

如果指定一个 **-l** 标志，I/O 定向视图用以下栏目变化表示。

kthr　　除了栏 **r** 和 **b**之外，栏 **p** 也将显示。

　　p　　每秒等待实际物理 I/O 的线程数。

页　　将显示新栏 **fi** 和 **fo**，代替 **re** 和 **cy** 栏。

　　fi　　每秒调入的文件。

　　fo　　每秒调出的文件。

标志

注： **-f** 和 **-s** 标志都可以在命令行上输入，但是系统只接收第一个指定的标志并覆盖第二个标志。

-f　报告从系统启动后的派生数目。

-i　显示从系统启动后每个设备造成的中断数目。

-l　用新的输出栏显示 I/O 定向视图，**p** 在标题 kthr 下；栏 **fi** 和 **fo** 在标题页面下，而不是栏下；**re** 和 **cy** 在页标题中。

-s 将总数结构中的内容写入到标准输出，该结构包含从系统初始化后调页事件的绝对计数。**-s** 标志只能与 **-v** 标志一起使用。如下描述了这些事件:

地址翻译错误
随地址翻译页面错误每发生一次而增加。可以请求 I/O 来解决页面错误，也可以不这样做。存储保护页面错误（固定失误）不包含在此计数之内。

页面输入
随着虚拟内存管理器每次读入页时而增加。计数随着页面空间和文件空间的页面输入而增加。它和页面输出统计信息一起表示实际 I/O（由虚拟内存管理器启动）的总量。

页面输出
随着虚拟内存管理器每次写出页时而增加。计数随着页面空间和文件空间的页面输出而增加。它和页面输入统计信息一起表示实际 I/O（由虚拟内存管理器启动）的总量。

调页空间页面输入
只随着 VMM 启动的来自调页空间的页面输入而增加。

调页空间页面输出
只随着 VMM 启动的来自调页空间的页面输出而增加。

总回收 当不启用一个新的 I/O 请求也可以满足地址翻译错误时，则增加它。如果页面以前已经被 VMM 请求过的，但是 I/O 还没有完成；或者页面被预读算法提前提取，但是被故障段隐藏了；或者如果页面已经被放入空闲列表中，但还没有重新使用，则会发生此情况。

零填充页面错误
如果缺页故障针对的是工作存储器，且可以通过指定一个帧并以零填充帧来满足它的话，则该值增加。

可执行填充页面错误
随着每个指令页面错误而增加。

用时钟检查页面
VMM 利用时钟算法实施伪最近最少使用（1ru）的页面替换模式。时钟检查过的页面是 *aged*。为每个时钟检查过的页面增加此计数值。

时钟指针的转动
随着每次 VMM 时钟旋转而增加（即在每一次完整的内存扫描后）。

用时钟释放的页面
随着时钟算法从实内存中选择释放的每一个页面而增加。

回溯　随着解决前一个页面错误时出现的每一个页面错误而增加。（新的页面错误必须首先解决，然后最初的页面错误也会*回溯*。）

固定丢失
VMM 通过除去对页面的寻址能力来增强对并发性的固定。由于一个固定丢失，一个页面错误也随之产生，每当此情况发生时，此计数增加。

空闲帧等待
在收集可用帧时，每次 VMM 等待一个进程时增加。

扩展 XPT 等待
每次正在进行提交而使得 VMM 等待一个进程时，随着正在被访问的段而增加。

暂挂 I/O 等待
每次 VMM 等待一个进程时随着要完成的页面输入 I/O 而增加。

启动 I/O
随着每个被 VMM 启动的读取或写入 I/O 请求而增加。此计数应该与页面输入和页面输出的总数相等。

iodones
在每次完成 VMM I/O 请求时增加。

CPU 上下文交换
随着每次 CPU 上下文交换而增加（新进程的分派）。

设备中断
在每次硬件中断上增加。

软件中断
在每次软件中断上增加。一次软件中断是一个类似于硬件中断（保存一些状态和服务器例程分支）的机器指令。系统调用用软件中断指令来完成，该指令转换控制到系统调用处理程序例程。

陷阱　不通过操作系统来维护。

syscalls
随着每次系统调用而增加。

-t　打印 **vmstat** 的每一输出行旁边的时间戳记。时间戳记按照 `HH: MM: SS` 格式显示。

注: 如果 **-f**、**-s** 或者 **-i** 标志被指定，则时间戳记将不打印。

-v 将虚拟内存管理器维护的不同统计信息写入标准输出。**-v** 标志只能与 **-s** 标志一起使用。

内存页 在 4 KB 页面数目中的实内存的多少。

lruable 页
要用于替换的 4 KB 页面的数目。此数目不包含被用于 VMM 内部页和用于内核文本的固定部分的页面。

空闲页面
空闲 4 KB 页面的数目。

内存池 指定内存池数目的调整参数（使用 **vmo** 管理）。

固定页面
固定的 4 KB 页面的数目。

maxpin 百分比
指定能被固定的实内存百分数的调整参数（使用 **vmo** 管理）。

minperm 百分比
实内存百分比的调整参数（使用 **vmo** 管理）。它指定一临界点，低于此临界点时阻止页面重新调度算法使用文件页面。

maxperm 百分比
实内存百分比的调整参数（使用 **vmo** 管理）。指定一临界点，高于此临界点时页面取走算法只取走文件页面。

numperm 百分比
当前由文件高速缓存使用的内存百分数。

文件页面
当前由文件高速缓存使用的 4 KB 页面的数目。

压缩百分比
由压缩页面使用的内存百分数。

压缩页面
压缩内存页面的数目。

numclient 百分数
被客户机页面占用的内存百分数。

maxclient 百分数
指定能用于客户机页面的最大内存百分数的调整参数（使用 vmo 管理）。

客户机页面
客户机页面的数目。

已调度的远程页面输出
调度用于客户机文件系统的页面输出的数目。

无 pbuf 而阻塞的暂挂磁盘 I/O
没有可用 pguf 而阻塞的暂挂磁盘 I/O 请求的数目。Pbufs 是用于保存逻辑卷管理器层上的 I/O 请求的固定的内存缓冲区。

无 psbuf 而阻塞的调页空间 I/O
没有可用 pguf 而阻塞的调页空间 I/O 请求的数目。Psbuf 是用于保存虚拟内存管理器层上的 I/O 请求的固定内存缓冲区。

-v *（由 **-v**显示的统计信息，接上页）:*

无 fsbuf 而阻塞的文件系统 I/O
没有可用 pguf 而阻塞的文件系统 I/O 请求的数目。Fsbuf 是用于保存文件系统层上的 I/O 请求的固定内存缓冲区。

无 fsbuf 而阻塞的客户机文件系统
没有可用 fguf 而阻塞的文件系统 I/O 请求的数目。NFS（网络文件系统）和 VxFS（Veritas）是客户机文件系统。Fsbuf 是用于保存文件系统层上的 I/O 请求的固定内存缓冲区。

无 fsbuf 而阻塞的外部页面调度程序文件系统 I/O
没有可用 fguf 而被阻塞的外部页面调度程序客户机文件系统 I/O 请求的数目。JFS2 是一个外部页面调度程序客户机文件系统。Fsbuf 是用于保存文件系统层上的 I/O 请求的固定内存缓冲区。

示例

1. 要显示引导后的统计信息摘要，请输入:
   ```
   vmstat
   ```
2. 要显示以 2 秒时间间隔的 5 次摘要，请输入:
   ```
   vmstat 2 5
   ```
   第一次摘要包含引导后的时间统计信息。
3. 要显示引导后的统计信息摘要（包含逻辑磁盘 scdisk13 和 scdisk14的统计信息），请输入:
   ```
   vmstat scdisk13 scdisk14
   ```
4. 要显示派生统计信息，请输入:
   ```
   vmstat  -f
   ```
5. 要显示不同事件的计数，请输入:
   ```
   vmstat -s
   ```
6. 要显示 **vmstat**的每一输出栏旁边的时间戳记，请输入:
   ```
   vmstat -t
   ```
7. 要显示有一套可替换栏的新 I/O 定向视图，请输入:
   ```
   vmstat -I
   ```
8. 要显示所有可用的 VMM 统计信息，请输入:
   ```
   vmstat -vs
   ```

文件

/usr/bin/vmstat 包含 **vmstat** 命令。

相关信息

iostat 和 **vmo** 命令。

*《AIX 5L V5.2 性能管理指南》*中的『监视和调整内存的使用』。

vmtune 命令

用途

用于设置虚拟内存管理器操作参数和其它 AIX 组件的兼容性脚本。

语法

vmtune [**-a**]

vmtune [**-M** *Number*] [**-p** *Number*] [**-P** *Number*] [**-t** *Number*] [**-f** *Number*] [**-F** *Number*] [**-l** *Number*] [**-r** *Number*] [**-R** *Number*] [**-N** *Number*] [**-w** *Number*] [**-W** *Number*] [**-k** *Number*] [**-c** *Number*] [**-b** *Number*] [**-B** *Number*] [**-u** *Number*] [**-d 0|1**] [**-s 0|1**] [**-S 0|1**] [**-L** *Number*] [**-g** *Number*] [**-h 0|1**] [**-m** *Number*] [**-C 0|1**] [**-v** *Number*] [**-n** *Number*] [**-j** *Number*] [**-J** *Number*] [**-z** *Number*] [**-Z** *Number*] [**-q** *Number*] [**-Q** *Number*] [**-V** *Number*] [**-i** *Number*] [**-y 0|1**] [**-T** *Number*]

描述

vmtune 样本程序正在逐渐淘汰，将来的发行版将不支持该样本程序。它会被 **vmo** 命令（对于所有的纯 VMM 参数）和 **ioo** 命令（对于所有与 I/O 有关的参数）所替代，这两个命令能用于设置先前由 **vmtune** 设置的大部分参数。**-v** 标志已经添加到 **vmstat** 以替换 **-A**标志，-A 标志显示计数器值而不是调整参数。对于AIX 5.2，为了帮助过渡提供了一个调用 **vmo** 和 **ioo** 的兼容性脚本。

vmtune命令仅能由 root 用户执行。由 **vmtune** 命令引起的更改直到系统下次重新引导后才会生效。在 AIX 5.2 中，**vmo** 和 **ioo** 都能用于使更改永久化和更改需要执行 **bosboot** 和／或重新引导系统才会生效的参数。

警告: 误用此命令会导致性能退化或操作系统失败。在使用 **vmtune** 更改系统参数之前，确保您已经阅读并理解*《性能管理指南》*中相应的调整部分。

标志

注: 如果没有指定标志，则调用 **vmo -a** 和 **ioo -a** 显示当前值。

-a	调用 **vmo -a** 和 **ioo -a** 来显示所有参数的当前值。
-A	调用 **vmstat -v** 来显示当前统计计数器。
-b *Number*	调用 **ioo -o numfsbuf=***Number* 来设置文件系统缓冲结构数。
-B *Number*	调用 **ioo -o hd_pbuf_cnt=***Number* 来设置 LVM 使用的 pbuf 数。
-c *Number*	调用 **ioo -o numclust=***Number* 来设置被后写处理的 16 KB 群集数。
-C 0|1	已接受，但不直接支持。使用 **vmo -r -o pagecoloring= 0|1** 来禁用／启动用于特定硬件平台的页面着色。
-d 0|1	调用 **vmo -o deffps=0|1** 来打开和关闭调页空间分配延时。
-f *Number*	调用 **vmo -o minfree=***Number* 来设置在空闲列表上的帧数。
-F *Number*	调用 **vmo -o maxfree=***Number* 来设置取走停止时的空闲列表上的帧数。
-g *Number*	已接受，但不直接支持。使用 **vmo -r -o lgpg_size** *Number***-o lgpg_regions=** *NumberRegions* 来设置硬件支持的大型页面的大小，以字节表示。
-h 0|1	调用 **vmo -o strict_maxperm= 0|1** 来指定 maxperm% 是否应该被强制限定。
-i *Number*	已接受，但不直接支持。使用 **vmo -r -o spec_dataseg_int=** *Number* 来设置当存储专用数据段标识符时使用的间隔时间。
-j *Number*	调用 **ioo -o j2_nPagesPerWriteBehindCluster=***Number* 来设置每个后写群集的页数。
-J *Number*	调用 **ioo -o j2_maxRandomWrite =***Number* 来设置随机写阈值计数。
-k *Number*	调用 **vmo -o npskill=***Number* 来设置进程开始被杀死的调页空间页数。
-l *Number*	调用 **vmo -o lrubucket=***Number* 来设置最近最少使用的页大小替换存储区大小。

-L *Number*	已接受，但不直接支持。使用 **vmo -r -o lgpg_regions**= *Number***-o lgpg_size** =*Size* 来设置保留的大型页数。
-m *Number*	已接受，但不直接支持。使用 **vmo -r -o mempools=***Number* 来设置内存池数。
-M *Number*	调用 **vmo -omaxpin=***Number* 来设置能被固定的实内存最大百分比。
-n *Number*	调用 **vmo -o nokilluid=***Number* 来指定当调页空间不足时不能被杀死的进程 uid 范围。
-N *Number*	调用 **ioo -o pd_npages=***Number* 来设置当一个文件被删除时从 RAM 的一个块里应该删除的页面数。
-p *Number*	调用 **vmo -o minperm%=***Number* 来设置一临界点，防止页面重新调度算法调度低于此临界点的文件页面。
-P *Number*	调用 **vmo -o maxperm%=***Number* 来设置一临界点，高于此临界时页面取走算法只取走文件页面。
-q *Number*	调用 **ioo -o j2_minPageReadAhead=***Number* 来设置预读页面的最小数。
-Q *Number*	调用 **ioo -o j2_maxPageReadAhead=***Number* 来设置预读页面的最大数。
-r *Number*	调用 **ioo -o minpgahead=***Number* 来设置带启动顺序预读的页数。
-R *Number*	调用 **ioo -o maxpgahead=***Number* 来设置预读页面的最小数。
-s 0|1	调用 **ioo -o sync_release_illock=0|1** 来启用能够在同步过程中使保持索引节点固定所需时间最小化的代码。
-S 0|1	调用 **vmo -o v_pinshm=0|1** 来启用在 **shmget** 系统调用中的**SHM_PIN** 标志。
-t *Number*	调用 **vmo -o maxclient%=***Number* 来设置一临界点，高于此临界点时页面取走算法只取走客户机文件页面。
-T *Number*	调用 **vmo -o pta_balance_threshold=***Number* 来设置新 pta 段的定位点。
-u *Number*	调用 **vmo -o lvm_bufcnt=***Number* 来设置用于原始物理 I/O 的 LVM 缓冲区数。
-v *Number*	已接受，但不直接支持。使用 **vmo -r -o framesets**= *Number* 来设置每内存池的帧集合数目。
-V *Number*	已接受，但不直接支持。使用 **vmo -r -o num_spec_dataseg**= *Number* 来设置保留的指定数据段标识数。
-w *Number*	调用 **vmo -o npswarn=***Number* 来设置在页面用后写算法被同步到磁盘之前 RAM 中累积的随机写的阈值。
-W *Number*	调用 **ioo -o maxrandwrt=***Number* 来设置调页空间空闲页面数，到达此数目时 **SIGDANGER** 被发送到进程。
-y 0|1	已接受，但不直接支持。使用 **vmo -r -o memory_affinity**=**0|1** 以在某个硬件上启用内存亲合关系。
-z *Number*	调用 **ioo -o j2_nRandonCluster=***Number* 来设置随机写阈值距离。
-Z *Number*	调用 **ioo -o j2_nBufferPerPagerDevice=***Number* 来设置每个页故障处理设备的缓冲区数。
-?	显示命令及其标志的描述。

关于可调参数的更多详细信息，请参阅 **vmo** 命令和 **ioo** 命令的『可调参数』章节。

相关信息

vmo 命令、**ioo** 命令。

*《AIX 5L V5.2 性能管理指南》*中的『对 **vmtune** 和 **schedtune** 的修改』。

AIX 5L Version 5.2 Performance Tools Guide and Reference 中的 Kernel Tuning 一节。

虚拟内存管理器（VMM）性能概述

vpdadd 命令

用途

将项添加到产品、lpp、历史和供应商数据库中。

语法

vpdadd { **-c** *Component* | **-p** *Product* | **-f** *Feature* } **-v** *v.r.m.f* [**-D** *Destdir*] [**-U** *Command*] [**-R** *Prereq*] [**-S** *Msg_Set*] [**-M** *Msg_Number*] [**-C** *Msg_Catalog*] [**-P** *Parent*] [**-l** *Description*]

描述

vpdadd 命令和安装程序一起使用或被安装程序使用，希望将这些安装程序列入重要产品数据库（VPD）。VPD 由产品、lpp 和历史数据库构成。库存清单数据库项必须由 **sysck** 命令添加。现在包含一种新的供应商数据库来跟踪使用目标目录和非 **installp** 卸载程序的产品。

vpdadd 命令使用最高级别为 *Product*，其次为 *Fveature*，再次为 *Component* 的树结构。

Component 是最底层的可安装单元，但是在这个层次结构里，不能选择一个 *Component* 用于安装或卸载。因此，如果安装程序正在使用 **vpdadd** 命令来更新安装数据库，它们应该察看自己的树形表示并且基于它们的结构添加项。如果每个安装只添加一个项，那么添加 *Product* 类型而不是 *Component* 类型将会允许该项列入卸载基于 Web 的系统管理器和 SMIT 界面。所有的项都是在 VPD 里进行的，但是在缺省 **lslpp** 列表（**-Lc**）中滤除了 *Component* 和 *Feature*。

标志

-C *Msg_Catalog* 指定用以搜索 *Component* 的一个翻译描述的消息编目。缺省（英文）描述是用 **-l** 标志指定的。如果消息编目不在标准 NLSPATH 中，那么应该给出全路径名。

-c *Component* 指定添加到 VPD 的 *Component* 名称。仅当一个项是唯一的时才能被添加。唯一性描述为有一个不同的目标目录。如果 *Component* 的相同实例已经在数据库里，则不添加任何项，并且会返回错误。这允许强制安装（即重新安装）。

-D *Destdir* 指定根（前缀）路径，当安装时（和用 **sysck** 命令添加到库存数据库时）这个路径被添加到 *Component* 中的全部文件中。*Component* 中的文件都按相对路径名列出，所以根路径是允许更改的。缺省目标目录是 **/opt**。

-f *Feature* 指定添加到 VPD 的 *Feature* 名称。仅当一个项是唯一的时才能被添加。唯一性描述为有一个不同的 VRMF 或目标目录。如果 *Feature* 的相同实例已经在数据库里，则不添加任何项，并且不返回错误。这允许强制安装（即重新安装）。

-l *Description* 指定 *Component*、*Feature* 或 *Product* 的缺省值描述。描述必须在双引号中指定。在描述内部允许使用单引号，双引号必须以一个 **** 开头。

-M *Msg_Number* 指定描述的消息号。

-P *Parent* 指定父代软件单元。一个 *Component* 根据它在树中的位置，指定一个 *Feature* 或 *Product* 做为其父代单元。这个标志用来在基于 Web 的系统管理器中提供树列表。

-p *Product* 指定添加到 VPD 的 *Product* 名称。仅当一个项是唯一的时才能被添加。唯一性是指有一个不同的 VRMF 或目标目录。如果 *Product* 的相同实例已经在数据库里，则不添加任何项，并且不返回错误。这允许强制安装（即重新安装）。

-R *Prereq* 指定一个安装 *Component* 所必备的 *Component*（文件集）。参数必须在引号中指定。可多次使用该标志来指定多项先决条件。尽管在安装时这些会被（安装程序）认为是先决条件，为了避免创建循环的先决条件链，在 *Product* 数据库里它们都被列为并存条件。

-S *Msg_Set* 指定消息集（如果目录中的消息超过一个）。

-U *Command* 为该 *Component* 指定用来启动卸载程序的 *Command*。这可能仅仅是一个命令路径名；如果存在一个全局卸载程序，它也可能包含参数。**geninstall** 命令调用该卸载程序，并且如果在 VPD 中设置了该值，**installp** 也不会卸载一个文件集。

-v *v.r.m.f* 正在添加的 *Component*、*Feature* 或 *Product* 的 VRMF。

示例

1. 以下示例显示了注册表服务如何调用 **vpdadd** 为 *Foo 产品* 添加一个 *Component*。该 *Component* 有两个先决条件，一个特定于操作系统，另一个列为 GUID。

   ```
   vpdadd -c EPL2890198489F -v 1.2.3.0 -R "bos.rte.odm 4.3.3.0" -R "8KDEOKY90245686 1.1.0.0"  \
   -U /usr/opt/foo/uninstaller.class -p KID892KYLIE25 -I "Foo Database Component"
   ```

2. 要向 VPD 添加一个新产品 `devices.pci.cool.rte`，请输入:

   ```
   vpdadd -p devices.pci.cool.rte -v 5.1.0.0 -U /usr/sbin/udisetup
   ```

文件

/usr/sbin/vpdadd

相关信息

installp 命令、**lslpp** 命令、**vpddel** 命令和 **geninstall** 命令。

---

vpddel 命令

用途

从产品、lpp、历史和供应商数据库中除去项。

语法

vpddel { **-c** *Component* | **-p** *Product* | **-f** *Feature* } **-v** *v.r.m.f* **-D** *Dest_dir*

描述

vpddel 命令从产品、lpp、历史和供应商数据库中除去项。必须指定 vrmf 和目标目录才能除去正确的项。

标志

-c *Component*	除去指定的 *Component*。在除去 *Component* 时，必须包含 VRMF。
-D *Dest_dir*	指定要除去的 *Component* 的目标目录。如果没有包含目标目录，则使用缺省目录 **/opt**。
-f *Feature*	指定要从供应商数据库中除去的 *Feature*。
-p *Product*	要从供应商数据库中除去的 *Product*。
-v *V.R.M.F*	指定要从 VPD 和供应商数据库中删除的 *Component* 版本、发行版、修订版以及修正级别。

示例

要从产品、历史、lpp 和供应商数据库中除去 *Component* `EPL2890198489F`，请输入:

```
vpddel -c EPL2890198489F -v 1.2.3.0 -D /usr/lpp/Foo
```

文件

/usr/sbin/vpddel

相关信息

vpdadd 命令和 lslpp 命令。

w 命令

用途

打印当前系统活动摘要。

语法

w [**-h**] [**-u**] [**-w**] [**-l** | **-s**] [*User*]

描述

w 命令打印系统中当前活动的摘要。该摘要包含以下内容:

用户	登录的用户。
tty	用户使用的 tty 名称。
注册 @	用户登录的日期和时间。
空闲	自一个程序上次试图从终端读取开始所持续的分钟数。
JCPU	该终端上的所有进程及其子进程使用的系统部件时间。
PCPU	当前活动进程所使用的系统部件时间。
什么	当前进程的名称和参数。

摘要标题行显示当前日期和时间、系统已经启动的时间、登录到系统的用户数和平均负载。平均负载是指在先前 1、5、15 分钟时间间隔内的可运行进程数。

下面的示例显示了登录时间字段所用的不同格式:

`10:25am`	用户在最近 24 小时内登录。
`Tue10am`	用户在 24 小时和 7 天之间登录。
`12Mar91`	用户在 7 天以前登录。

如果用 *User* 参数指定了用户名，则输出仅限于该用户。

标志

-h 禁用标题。

-l 用长格式打印摘要。这是缺省值。

-s 用短格式打印摘要。在短格式中，tty 是缩写，并且登录时间、系统部件时间和命令参数都被省略。

-u 打印日期和时间、自上次系统启动以来的时间总计、登录的用户数和正在运行的进程数。这是缺省值。指定 **-u** 标志而不指定 **-w** 或 **-h** 标志等效于 **uptime** 命令。

-w 等效于同时指定 **-u** 和 **-l** 标志，这是缺省值。

文件

/etc/utmp
包含用户列表。

相关信息

who 命令、**finger** 命令、**ps** 命令、**uptime** 命令。

wait 命令

用途

等待直到一个进程标识终止。

语法

wait [*ProcessID* ...]

描述

wait 命令等待（暂停执行）直到由 *ProcessID* 变量指定的进程标识终止。如果没有指定 *ProcessID* 变量，**wait** 命令会等待直到调用 shell 的所有已知进程标识终止并且以 0 为出口状态退出。如果 *ProcessID* 变量代表一个未知的进程标识，**wait** 命令会把它们视为以 127 为出口状态退出的已知进程标识。**wait** 命令以 *ProcessID* 变量指定的上个进程标识的出口状态退出。

标志

ProcessID 指定一个命令的无符号十进制整数进程标识，**wait** 命令会等待直到该进程标识终止。

出口状态

如果指定一个或多个操作数，所有的操作数终止或不为调用 shell 所知，并且上个操作数指定的状态是已知的，那么 **wait** 命令的出口状态与上个操作数指定的命令出口状态信息相同。如果进程由于收到信号而异常终止，那么出口状态会大于 128 并且与其它信号生成的出口状态信息不同，尽管未指定确切的状态值（请参阅 **kill -l** 命令选项）。否则，**wait** 命令会以以下值中的一个退出:

0 以无操作数方式调用 **wait** 命令，并且调用的 shell 所已知的所有进程标识已经终止。
1-126 **wait** 命令检测到一个错误。
127 由上一个指定的 *ProcessID* 操作数所标识的命令是未知的。

文件

/usr/bin/wait 包含 **wait** 命令。

相关信息

shutdown 命令、**sleep** 命令、**wall** 命令。

alarm 子例程、**pause** 子例程、**sigaction** 子例程。

*《AIX 5L V5.2 系统用户指南: 操作系统与设备》*中的『shell 概述』。

wall 命令

用途

将一条消息写给所有登录的用户。

语法

wall [**-a**] [**-g** *Group*][*Message*]

描述

wall 命令将一条消息写给所有登录的用户。如果没有指定 *Message* 参数， **wall** 命令从标准输入读取消息直到遇到文件结束符。然后，消息被发送给所有已登录用户。消息以以下标题开始:

```
hh:mm:ss 来自于
 user@node

 (tty)的广播消息...
```

`hh:mm:ss` 代表消息被发送时的小时、分钟和秒。

要覆盖其它用户设置的任何保护，必须以 root 用户权限进行操作。典型情况下，root 用户使用 **wall** 命令来警告所有其它用户系统即将关闭。

注:

1. **wall** 命令仅对本地节点发送消息。
2. 消息可以包含多字节字符。

标志

表 1.

-a	执行缺省操作。这个标志是为兼容 System V 而提供的。它对控制台和伪终端广播消息。
-g *Group*	仅对指定的组广播。

文件

/dev/tty 指定一个设备。

相关信息

mesg 命令、**su** 命令、**write** 命令。

---

ewallevent、wallevent 命令

用途

向所有登录用户广播一个事件或一个重整事件。

语法

ewallevent [**-h**]

wallevent [**-h**]

描述

ewallevent 脚本总是用英文返回消息。返回 **wallevent** 脚本消息所用的语言取决于语言环境设置。

当事件或重整事件发生时，这些脚本对所有当前登录到主机的用户广播关于这个事件或重整事件的消息。事件或重整事件信息在环境变量中被事件响应资源管理器捕获并公布，这些环境变量是在事件或重整事件发生时由事件响应资源管理器生成的。这些脚本能用作事件响应资源运行的操作。它们还能用作创建其它用户定义操作的模板。

请参阅 *RSCT Administration Guide* 以理解事件响应资源如何运行操作命令。

当事件或重整事件发生时，消息会以这种格式显示在所有登录用户的控制台上，对事件或重整事件来说这些脚本是一个响应操作:

hh:mm:ss 来自于 *user@host*（*tty*）的广播消息...

yy/mm/dd hh:mm:ss 在 *resource_class_name* 的资源 *resource_name* 上的因条件 *condition_name* 发生了 *severity event_type*。
该资源在 *node_name* 受到监视，并驻留于 {*node_names*}。

返回有关在 *RSCT Administration Guide* 中列出并描述的 ERRM 环境变量的事件信息，而且也包含以下信息:

本地时间
: 观察事件和重整事件的时间。ERRM 提供的实际环境变量是 RRM_TIME。这个值已经本地化且在显示之前已转换成可阅读的格式。

wallevent 脚本捕获环境变量值并用 **wall** 命令将一条消息写到当前登录用户控制台。

标志

–h 将脚本的使用语句写到标准输出。

参数

log_file
: 指定记录事件信息的文件名。应当指定 *log_file* 参数的绝对路径。

  log_file 视为一个循环日志并大小固定为 64KB。当 *log_file* 写满时，新项覆盖现有的最旧的项。

  如果 *log_file* 已经存在，事件信息被追加到它的后面。如果 *log_file* 不存在，会创建它以使事件信息能写入其中。

出口状态

0 脚本已经成功运行。

1 当脚本运行时发生错误。

限制

1. 这些脚本必须运行在 ERRM 正在运行的节点上。
2. **wall** 命令用来将一条消息写到当前登录用户控制台。请参考 **wall** 的联机帮助页以获取关于 **wall** 命令的更多信息。

标准输出

当指定 **-h** 标志时，将脚本使用语句写到标准输出。

示例

1. 假定在关键通知响应里 **wallevent** 脚本是一个预定义操作，它与资源 **/var** 上**已使用的 /var 空间**条件相关联。已符合为该情况定义的事件表达式的阈值，同时发生一个事件。关键通知响应发生，且 **wallevent** 运行。在所有登录用户的控制台上显示以下消息:

   ```
   18:42:03 来自于 joe@neverland.com（pts/6）的广播消息...

   02/03/28 18:41:50 时在 IBM.FileSystem 的资源 /var 上因“条件”已使用的 /var 空间而发生了关键事件
   该资源在 c174n05 受到监视，并驻留于 {c174n05}。
   ```

2. 当为资源 **/var** 上的**已使用的 /var 空间**情况发生一个重整事件，下列信息显示在所有登录用户的控制台上:

   ```
   18:42:03 来自于 joe@neverland.com（pts/6）的广播消息...

   02/03/28 18:41:50 时在 IBM.FileSystem 的资源 /var 上因“条件”已使用的 /var 空间而发生了关键重整事件
   该资源在 c174n05 受到监视，并驻留于 {c174n05}。
   ```

位置

/usr/sbin/rsct/bin/ewallevent 包含 **ewallevent** 脚本

/usr/sbin/rsct/bin/wallevent 包含 **wallevent** 脚本

相关信息

命令: **wall**

书籍: 请参阅 *RSCT Administration Guide* 以了解有关事件响应资源管理器（ERRM）和有关如何使用 ERRM 环境变量的信息

---

watch 命令

用途

观察一个可能不可信赖的程序。

语法

watch [**-e** *Events*] [**-o** *File*] *Command* [*Parameter* ...]

描述

watch 命令允许 root 用户或一个审计组成员观察一个认为是不可信赖的程序操作。**watch** 命令执行用 *Command* 参数指定的程序，可以带任何 *Parameter* 字段也可以没有，记录所有审计事件或用 **-e** 标志指定的审计事件。

watch 命令观察所有在程序运行时创建的进程，包括任何子进程。**watch** 命令会继续直到所有的进程（包括它创建的进程）退出，以观察所有发生的事件。

watch 命令格式化审计记录并把它们写到标准输出或用 **-o** 标志指定的一个文件。

为了使 **watch** 命令起作用，一定不要配置和启用审计子系统。

标志

-e*Events* 指定要审计的事件。*Events* 参数是在 **/etc/security/audit/events** 文件中定义的以逗号分隔的审计事件列表。缺省值是所有事件。

-o *File* 指定输出文件的路径名。如果没有使用 **-o** 标志，则输出写到标准输出。

安全性

访问控制：这个命令应授予 root 用户和审计组成员执行（x）访问权。该命令应该为 root 用户的 **setuid**，以使其能访问其它审计子系统命令和文件，并有**可信计算库**属性。

访问的文件：

方式	文件
r	**/dev/audit**
x	**/usr/sbin/auditstream**
x	**/usr/sbin/auditselect**
x	**/usr/sbin/auditpr**

示例

1. 要观察 **bar** 命令打开的所有文件，请输入：

   ```
   watch -e FILE_Open /usr/lpp/foo/bar -x
   ```

   这个命令打开审计设备并执行 **/usr/lpp/foo/bar** 命令。然后它读取所有记录并选择和格式化那些 `FILE_Open` 事件类型的记录。
2. 要观察 `xyzproduct` 程序的安装，该程序可能是不可信赖的，请输入：

   ```
   watch /usr/sbin/installp xyzproduct
   ```

   这个命令打开审计设备并执行 **/usr/sbin/installp** 命令。然后读取所有记录并将它们格式化。

文件

/usr/sbin/watch	包含 **watch** 命令。
/dev/audit	指定从中读取审计记录的审计设备。

相关信息

audit 命令、**auditbin** 守护程序、**auditcat** 命令、**auditpr** 命令、**auditselect** 命令、**auditstream** 命令、**login** 命令、**logout** 命令、**su** 命令。

auditread 子例程。

关于用户的标识和认证、自主访问控制、可信计算库和审计的更多信息，请参阅《*AIX 5L V5.2 安全指南*》。

关于审计的更多信息，请参阅《*AIX 5L V5.2 安全指南*》中的『审计综述』。

---

wc 命令

用途

统计一个文件中的行数、字数、字节数或字符数。

语法

wc [**-c** | **-m**] [**-l**] [**-w**] [*File* ...]

wc **-k** [**-c**] [**-l**] [**-w**] [*File* ...]

描述

缺省情况下，**wc** 命令对 *File* 参数指定的文件中的行数、字数和字节数进行计数。这个命令将换行符数、字数和字节数写到标准输出并为所有指定的文件保留一个总数。

当使用 *File* 参数时， **wc** 命令显示文件名以及请求的计数。如果没有给 *File* 参数指定一个文件名，**wc** 命令使用标准输入。

wc 命令受 **LANG**、**LC_ALL**、**LC_CTYPE** 和 **LC_MESSAGES** 环境变量影响。

wc 命令把一个字看作是被一个空格（如空白和跳格）分隔的非零长度字符串。

标志

-c 统计字节数，除非指定 **-k** 标志。如果指定 **-k** 标志，**wc** 命令统计字符数。

-k 统计字符数。指定 **-k** 标志等同于指定 **-klwc** 标志。如果把 **-k** 标志同其它标志一起使用，那么必须包含 **-c** 标志。否则，将会忽略 **-k** 标志。请参阅示例 4 和5 以获取更多信息。

注: 这个标志在将来的发行版中将会撤销。

-l 统计行数。

-m 统计字符数。这个标志不能与 **-c** 标志一起使用。

-w 统计字数。一个字被定义为由空白、跳格或换行字符分隔的字符串。

注: 如果没有指定标志，缺省情况下，**wc** 将统计文件中的或来自标准输入的行、字和字节数。

出口状态

这个命令返回以下出口值:

0 命令成功运行。

>0 发生一个错误。

示例

1. 要显示一个文件的行数、字数和字节数统计，请输入:

   ```
   wc chap1
   ```

   wc 命令显示 `chap1` 文件中的行数、字数和字节数。

2. 要仅显示字节数和字数统计，请输入:

   ```
   wc -cw chap*
   ```

   wc 命令显示以 `chap` 开头的每个文件中的字节数和字数。这个命令也显示这些文件中的总字节数和字数。

3. 要显示一个文件的行数、字数和字符数统计，请输入:

   ```
   wc -k chap1
   ```

   wc 命令显示 `chap1` 文件中的行数、字数和字符数。

4. 要显示一个文件的字数和字符数统计，请输入:

```
wc -kcw chap1
```

wc 命令显示 chap1 文件中的字符数和字数。

5. 要在标准输入上使用 **wc** 命令，请输入:

```
wc -klw
```

wc 命令显示在标准输入中的行数和字数。**-k** 标志被忽略。

6. 要显示一个文件的字符数统计，请输入:

```
wc -m chap1
```

wc 命令显示 chap1 文件中的字符数。

7. 要在标准输入上使用 **wc** 命令，请输入:

```
wc -mlw
```

wc 命令显示标准输入中的行数、字数和字符数。

文件

/usr/bin/wc、**/bin/wc**	包含 **wc** 命令。
/usr/ucb/wc	包含到 **wc** 命令的符号链路。

相关信息

*《AIX 5L V5.2 系统用户指南: 操作系统与设备》*中的『文件概述』和『输入输出重定向概述』。

AIX 5L Version 5.2 National Language Support Guide and Reference 中的 Understanding Locale Environment Variables。

---

what 命令

用途

显示文件中的标识信息。

语法

what [**-s**] *Pathname/File*.

描述

what 命令搜索指定的文件，查找所有发生用 **get** 命令替换 **@(#)** 关键字母的模式（参阅 **get** 或 **prs** 命令获取关于标识关键字的描述）。根据约定，被替代的值是 ″@(#)″ （双引号、符号″@″、左括号、井号、右括号、双引号）。如果没有指定文件， **what** 命令从标准输入读取。

无论模式后跟是什么字符，**what** 命令都写到标准输出，一直到（但是不包含）第一个双引号 （″）、大于符号（>）、换行符、反斜杠 (\) 或空字符。

what 命令应该和 **get** 命令一起使用，这样会自动插入标识信息。也可以对手工插入信息的文件使用 **what** 命令。

注: **what** 命令可能在可执行文件中找不到 SCCS 标识字符串。

标志

-s 仅搜索第一个出现的 **@(#)** 模式。

出口状态

该命令返回以下出口值:

0 找到所有匹配内容。
1 没有找到任何匹配内容。

示例

假设文件 `test.c` 包含一个 C 程序，这个程序含有以下行:

```
char ident[ ] = "@(#)Test Program";
```

如果编译 `test.c` 来生成 `test.o`，那么命令:

```
what test.c test.o
```

显示:

```
test.c:
Test Program
test.o:
Test Program
```

注: 如果文件不在当前目录里，则需要完整的文件路径名 `usr/bin/test.c` 和 `user/bin/test.o`。

文件

/usr/bin/what 包含 **what** 命令。

相关信息

get 命令、**sccshelp** 命令。

sccsfile 文件格式。

AIX 5L Version 5.2 General Programming Concepts: Writing and Debugging Programs 中的 List of SCCS Commands。

AIX 5L Version 5.2 General Programming Concepts: Writing and Debugging Programs 中的 Source Code Control System (SCCS) Overview。

---

whatis 命令

用途

描述一个命令执行什么功能。

语法

whatis [**-M** *PathName*] *Command* ...

描述

whatis 命令在用 **catman -w** 命令创建的数据库中查找 *Command* 参数指定的命令、系统调用、库函数或特殊文件名。**whatis** 命令显示手册部分的页眉行。然后可以发出 **man** 命令以获取附加的信息。

whatis 命令等同于使用 **man -f** 命令。

> **注**: 当 **/usr/share/man/whatis** 数据库是使用 **catman -w** 命令从 HTML 库构建时，部分 3 等同于部分 2 或 3。要获取这些部分的进一步解释，请参阅 **man** 命令。

标志

-M *PathName* 指定一个可替代的搜索路径。搜索路径由 *PathName* 参数指定，并且是一个 **whatis** 命令期望找到标准手册子目录的以冒号分隔的目录列表。

示例

要找出 **ls** 命令做了哪些操作，请输入:

```
whatis ls
```

这个命令产生以下输出:

```
ls(1)  — 显示一个目录的内容。
```

文件

/usr/share/man/whatis 包含 **whatis** 数据库。

相关信息

apropos 命令、**catman** 命令、**ls** 命令、**man** 命令。

---

whatnow 命令

用途

为草稿处置启动一个提示界面。

语法

whatnow [{ **-draftfolder +***Folder* | **-nodraftfolder** | *File* } { **-draftmessage** *Message* | *File* }] [**-editor** *Editor* | **-noedit**] [**-prompt** *String*]

描述

whatnow 命令为消息处置提供一个界面。缺省情况下，界面在当前草稿消息上操作。当输入 **whatnow** 命令时，系统载入界面并返回以下提示:

```
现在怎样?
```

在该界面中能用 **whatnow** 子命令操作消息草稿。要查看子命令列表，在现在怎样？提示时按 Enter 键。要退出界面，按 q。

如果没有指定 **-draftfolder** 标志或如果没有定义 **$HOME/.mh_profile** 文件中的 `Draft-Folder:` 项，**whatnow** 命令会在 MH 目录中搜索一个**草稿**文件。在 **-draftfolder +***Folder* 标志之后指定消息，与指定 **-draftmessage** 标志的一样。

要更改 **whatnow** 命令的缺省编辑器，请使用 **-editor** 标志或在 *UserMhDirectory***/.mh_profile** 文件中定义 `Editor:` 项。

注: **comp**、**dist**、 **forw** 或 **repl** 命令使用与 **whatnow** 命令相同的界面。

标志

-draftfolder +*Folder*
指定包含消息的文件夹。缺省情况下，系统使用 *UserMhdirectory***/draft** 文件。在**-draftfolder +***Folder* 之后指定一条消息，与使用**-draftmessage** 标志时一样。

-draftmessage *Message*
指定草稿消息。

-editor *Editor*
指定 *Editor* 变量的值是用于撰写或修改消息的初始编辑器。

-help
列出命令语法，可用的开关（切换）和版本信息。

注: 对于 MH，这个标志的名字必须完全拼写出来。

File
用户选择的草稿文件。

Message
指定消息。用以下参考资料来指定消息:

Number 消息编号。

cur 或 .（句点）
当前消息。这是缺省值。

first 在文件夹里的第一条消息。

last 在文件夹里的最后一条消息。

next 当前消息以后的消息。

prev 当前消息以前的消息。

-nodraftfolder
将草稿放到 *UserMhDirectory***/draft** 文件中。

-noedit 禁止初始编辑。

-prompt *String*
将指定的字符串作为提示符使用。缺省的字符串是 现在怎样？。

whatnow 子命令

whatnow 子命令使您能够编辑消息，指导消息的处置，或结束 **whatnow** 命令的处理。

display [*Flags*]
显示重新分发或答复的消息。可以指定任何对列表程序有效的 *Flags* 参数。（使用在 **$HOME/.mh_profile** 文件中的 `lproc:` 项来设置一个缺省列表程序。）如果指定了对列表程序无效的标志， **whatnow** 命令不会传递草稿的路径名。

edit [*CommandString*]
用 *CommandString* 参数为消息指定一个编辑器。可以指定编辑器和对这个编辑器有效的任何标志。如果不指定一个编辑器，**whatnow** 命令会使用在 *UserMhDirectory***/.mh_profile** 文件中的 `Editor:` 项指定的编辑器。如果没有定义 `Editor:` 项，**whatnow** 命令会启动在先前的编辑会话中使用的编辑器。

list [*Flags*]
显示草稿。可以指定对列表程序有效的任何 *Flags* 参数。（要指定一个缺省列表程序，请在 **$HOME/.mh_profile** 文件中设置 `lproc:` 项。）如果指定了对列表程序无效的任何标志，**whatnow** 命令不会传递草稿路径名。

push [*Flags*]
在后台发送消息。可以指定对 **send** 命令有效的任何标志。

quit [**-delete**]
结束 **whatnow** 会话。如果指定了 **-delete** 标志，**whatnow** 命令会删除草稿。否则，**whatnow** 命令会存储草稿。

refile [*Flags*] **+***Folder*
将指定文件夹中的草稿归档并提供一个新的指定了以前格式的草稿。可以指定对作为 **fileproc** 运行的命令有效的任何 *Flags*参数。（可以在 **$HOME/.mh_profile** 文件中设置一个缺省 `fileproc:` 项。）

send [*Flags*]
发送消息。可以指定对 **send** 命令有效的任何标志。

whom [*Flags*]
显示消息将会被发送到的地址。可以指定对 **whom** 命令有效的任何标志。

概要文件项

以下项是在 *UserMhDirectory***/.mh_profile** 文件中的项:

`Draft-Folder:`	为草稿设置缺省文件夹。
`Editor:`	设置缺省编辑器。
`fileproc:`	指定用于接力传送消息的程序。
`LastEditor-next:`	指定在退出 *LastEditor* 变量指定的编辑器之后使用的编辑器。
`lproc:`	指定用于列出消息内容的程序。
`Path:`	指定 *UserMhDirectory*。
`sendproc:`	指定用于发送消息的程序。
`whomproc:`	指定用于确定消息将会被发送到哪个用户的程序。

示例

1. 当答复一条消息时为了显示原始消息，在现在怎样？提示下输入以下内容:

   ```
   display
   ```

   系统显示原始消息。如果从另外一个命令而不是从 **dist** 或 **repl** 命令中输入 **display** 子命令，则会收到一条系统消息表明没有可显示的备用消息。

2. 要用 `vi` 编辑器编辑草稿消息，请在现在怎样？提示下输入以下内容:

   ```
   edit vi
   ```

3. 要用在 **.mh_profile** 文件中指定的缺省编辑器编辑草稿消息，请在现在怎样？提示下输入以下内容:

   ```
   edit
   ```

4. 要列出已经编辑好的草稿消息的内容，请在现在怎样？提示下输入以下内容:

   ```
   list
   ```

   将显示正在撰写的草稿消息。

5. 要在后台发送草稿消息并立即获取一个 shell 提示，请在现在怎样？提示下输入以下内容:

   ```
   push
   ```

   草稿消息已发送，并且会立即收到 shell 提示。

6. 要退出正在撰写的草稿消息并将其保存到一个文件以使您能在以后完成撰写，请在现在怎样？提示下输入以下内容:

   ```
   quit
   ```

   系统会用一条类似于以下内容的消息来响应。

   ```
   whatnow: draft left on /home/dale/Mail/draft
   ```

   在该示例中，用户 dale 的草稿消息保存到 **/home/dale/Mail/draft** 文件中。
7. 要退出正在撰写的草稿消息并删除该消息，请在现在怎样？提示下输入以下内容:

   ```
   quit -delete
   ```

   当删除草稿消息时显示 shell 提示。
8. 要在发送之前将正在编辑的草稿消息归档，请在现在怎样？提示下输入以下内容:

   ```
   refile +tmp
   ```

   系统会用一条类似于以下内容的消息来响应:

   ```
   Create folder "home/dale/Mail/tmp"?
   ```

   在该示例中，如果回答 yes，草稿消息会归档在用户 dale 的文件夹 tmp 中。
9. 要发送已经撰写完的草稿消息，请在现在怎样？提示下输入以下内容:

   ```
   send
   ```

   当消息被发送后会显示 shell 提示。
10. 要验证在草稿消息中的所有邮件投递系统能识别的地址，请在现在怎样？提示下输入以下内容:

    ```
    whom
    ```

    系统会用一条类似于以下内容的消息来响应:

    ```
    jeanne... 用户 未知
    dale@venus... 无法投递
    ```

    在该示例中，邮件投递系统能将 dale@venus 作为一个正确的地址识别，但是不能将 jeanne 作为一个正确的地址识别。

文件

$HOME/.mh_profile	指定 MH 用户概要表。
UserMhDirectory/**draft**	包含当前消息草稿。
/usr/bin/whatnow	包含 **whatnow** 命令。

相关信息

comp 命令、**dist** 命令、**forw** 命令、**prompter** 命令、**refile** 命令、**repl** 命令、**rmm** 命令、**scan** 命令、**send** 命令、**whom** 命令。

mh_alias 文件格式、**mh_profile** 文件格式。

*《AIX 5L V5.2 系统用户指南: 通信与网络》*中的『邮件概述』。

whereis 命令

用途

找到程序的源、二进制文件或手册。

语法

whereis [**-s**] [**-b**] [**-m**] [**-u**] [{ { **-S** | **-B** | **-M** } *Directory* ... }... **-f**] *File* ...

描述

whereis 命令找到指定文件的源、二进制文件和手册等各部分。提供的名称首先去除前导路径名组件和任何（单个的）格式为 *.ext* 的尾随扩展名（例如，**.c**）。由于使用源代码控制系统（请参阅 **SCCS**）而引起的 **s.** 前缀也会被处理。然后命令试图从一个标准位置列表中查找期望的程序。

如果输入错误的选项，则返回一条用法消息。在其它情况下，不提供诊断。

标志

如果给出 **-b**、**-s**、**-m** 或 **-u** 标志中的任一个，则 **whereis** 命令仅分别搜索二进制文件、源、手册或不寻常部分（或其中的两个）。

-b 搜索文件的二进制部分。
-m 搜索文件的手册部分。
-s 搜索文件的源部分。
-u 搜索不寻常文件。如果一个文件没有每个请求类型的一个项则认为该文件是不寻常文件。输入 `whereis -m -u *`，则寻找当前目录中那些没有文档的文件。

-B、**-M**、 和 **-S** 标志可以用于更改或限制 **whereis** 命令搜索的位置。由于该程序为了运行得更快而使用了 **chdir** 子例程，所以用 **-M**、**-S** 和 **-B** 标志目录列表给出的路径名必须是完整的；例如，它们必须以一个 **/**（斜杠）开始。

-B 与 **-b** 一样，但是添加了要搜索的目录。更改或限制 **whereis** 命令搜索二进制文件的位置。
-M 与 **-m** 一样，但是添加了要搜索的目录。更改或限定 **whereis** 命令搜索手册部分的位置。
-S 与 **-s** 一样，但是添加了要搜索的目录。更改或限定 **whereis** 命令搜索源的位置。
-f 终止最后的 **-M**、**-S** 或 **-B** 目录列表并发文件名起始位置信号。

示例

要查找 **/usr/ucb** 目录中的所有文件，这些文件或者在 **/usr/man/man1** 目录里没有归档或者在 **/usr/src/cmd** 目录里没有源，请输入:

```
cd /usr/ucb
whereis -u -M /usr/man/man1 -S /usr/src/cmd -f *
```

文件

/usr/share/man/* 包含手册文件的目录。
/sbin，**/etc**，**/usr/**{**lib**，**bin**，**ucb**，**lpp**} 包含二进制文件的目录。
/usr/src/* 包含源代码文件的目录。

相关信息

chdir 子例程。

which 命令

用途

定位一个程序文件，包含别名和路径。

语法

which [*Name* ...]

描述

which 命令取得程序名列表并寻找当这些文件名作为命令给定时所运行的文件。**which** 命令展开每个参数（如果它有别名），并沿着用户的路径搜索它。别名和路径从用户主目录中的 **.cshrc** 文件取得。如果 **.cshrc** 文件不存在，或者如果路径没有在 **.cshrc** 文件中定义，则 **which** 命令使用在用户环境中定义的路径。

如果名称的别名多于一个单字或如果在路径里没有找到一个有参数名称的可执行文件，那么将会给出诊断信息。

在 Korn shell 中，能使用 **whence** 命令来生成一个更详细的报告。请参阅《*AIX 5L V5.2 系统用户指南：操作系统与设备*》中的『Korn Shell 专用命令』以获取关于 **whence** 命令的更多信息。

示例

要查找与 `lookup` 命令名有关的可执行文件:

```
which lookup
```

文件

$HOME/.cshrc 包含别名和路径值的源。

相关信息

csh 命令、**find** 命令、**file** 命令、**ksh** 命令、**sh** 命令、**whereis** 命令。

《*AIX 5L V5.2 系统用户指南：操作系统与设备*》中的『shell 概述』描述了 shell、不同类型以及它们如何影响命令的解释方式。

《*AIX 5L V5.2 系统用户指南：操作系统与设备*》中的『命令概述』。

which_fileset 命令

用途

搜索 **/usr/lpp/bos/AIX_file_list** 文件，查找指定文件名或命令。这个命令仅适用于 AIX 4.2.1 或后续版本。

语法

which_fileset [*File*]

描述

which_fileset 命令搜索 **/usr/lpp/bos/AIX_file_list**，查找指定文件名或命令名，并打印出提供该文件或命令的文件集名。

/usr/lpp/bos/AIX_file_list 文件很大并且不会自动安装。必须安装 **bos.content_list** 文件集以接收这个文件。

File 参数可以是命令名、全路径名或一个搜索模式的正则表达式。

示例

1. 要显示 dbx 命令在哪个文件集中提供，请输入:

   ```
   which_fileset dbx
   ```

   屏幕显示以下内容:

   ```
   /usr/bin/dbx > /usr/ccs/bin/dbx             bos.adt.debug 4.2.1.0
   /usr/ccs/bin/dbx                            bos.adt.debug 4.2.1.0
   ```

2. 要显示所有包含 *sendmail* 字符串的命令和路径，请输入:

   ```
   which_fileset sendmail.*
   ```

   屏幕显示以下内容:

   ```
   /usr/ucb/mailq > /usr/sbin/sendmail    bos.compat.links 4.2.0.0
   /usr/ucb/newaliases > /usr/sbin/sendmail bos.compat.links 4.2.0.0
   /usr/lib/nls/msg/Ca_ES/sendmail87.cat  bos.msg.Ca_Es.net.tcp.client 4.2.0.0
   /usr/lib/nls/msg/ca_ES/sendmail87.cat  bos.msg.ca_Es.net.tcp.client 4.2.0.0
   /usr/lib/nls/msg/cs_CZ/sendmail87.cat  bos.msg.cs_CZ.net.tcp.client 4.2.0.0
   /usr/lib/nls/msg/De_DE/sendmail87.cat  bos.msg.De_DE.net.tcp.client 4.2.0.0
   /usr/lib/nls/msg/de_DE/sendmail87.cat  bos.msg.de_DE.net.tcp.client 4.2.0.0
   /usr/lib/nls/msg/En_US/sendmail87.cat  bos.msg.En_US.net.tcp.client 4.2.0.0
   /usr/lib/nls/msg/en_US/sendmail87.cat  bos.msg.en_US.net.tcp.client 4.2.0.0
   /usr/lib/nls/msg/Es_ES/sendmail87.cat  bos.msg.Es_ES.net.tcp.client 4.2.0.0
   /usr/lib/nls/msg/es_ES/sendmail87.cat  bos.msg.es_ES.net.tcp.client 4.2.0.0
   /usr/lib/nls/msg/Fr_FR/sendmail87.cat  bos.msg.Fr_FR.net.tcp.client 4.2.0.0
   /usr/lib/nls/msg/fr_FR/sendmail87.cat  bos.msg.fr_FR.net.tcp.client 4.2.0.0
   /usr/lib/nls/msg/hu_HU/sendmail87.cat  bos.msg.hu_HU.net.tcp.client 4.2.0.0
   /usr/lib/nls/msg/It_IT/sendmail87.cat  bos.msg.It_IT.net.tcp.client 4.2.0.0
   /usr/lib/nls/msg/it_IT/sendmail87.cat  bos.msg.it_IT.net.tcp.client 4.2.0.0
   /usr/lib/nls/msg/Ja_JP/sendmail87.cat  bos.msg.Ja_JP.net.tcp.client 4.2.0.0
   /usr/lib/nls/msg/ja_JP/sendmail87.cat  bos.msg.ja_JP.net.tcp.client 4.2.0.0
   /usr/lib/nls/msg/ko_KR/sendmail87.cat  bos.msg.ko_KR.net.tcp.client 4.2.0.0
   /usr/lib/nls/msg/pl_PL/sendmail87.cat  bos.msg.pl_PL.net.tcp.client 4.2.0.0
   /usr/lib/nls/msg/ru_RU/sendmail87.cat  bos.msg.ru_RU.net.tcp.client 4.2.0.0
   /usr/lib/nls/msg/Sv_SE/sendmail87.cat  bos.msg.Sv_SE.net.tcp.client 4.2.0.0
   /usr/lib/nls/msg/sv_SE/sendmail87.cat  bos.msg.sv_SE.net.tcp.client 4.2.0.0
   /usr/lib/nls/msg/ZH_CN/sendmail87.cat  bos.msg.ZH_CN.net.tcp.client 4.2.0.0
   /usr/lib/nls/msg/zh_CN/sendmail87.cat  bos.msg.zh_CN.net.tcp.client 4.2.0.0
   /usr/lib/nls/msg/Zh_TW/sendmail87.cat  bos.msg.Zh_TW.net.tcp.client 4.2.0.0
   /usr/lib/nls/msg/zh_TW/sendmail87.cat  bos.msg.zh_TW.net.tcp.client 4.2.0.0
   /etc/sendmail.cf                       bos.net.tcp.client.4.2.1.0
   /usr/lib/sendmail > /usr/sbin/sendmail bos.net.tcp.client.4.2.1.0
   /usr/sbin/mailq > /usr/sbin/sendmail   bos.net.tcp.client.4.2.1.0
   /usr/sbin/newaliases > /usr/sbin/sendmail bos.net.tcp.client.4.2.1.0
   /usr/sbin/sendmail                     bos.net.tcp.client.4.2.1.0
   ```

3. 要查找提供 **/usr/sbin/which_fileset** 命令的位置，请输入:

   ```
   which_fileset /usr/bin/which_fileset
   ```

屏幕显示:

```
/usr/sbin/which_fileset      bos.rte.install 4.2.1.0
```

who 命令

用途

识别当前登录的用户。

语法

who [**-a** | **-b -d -i -l -m -p -q -r -s -t -u -w -A -H -T**] [*File*]

who am { **i** | **I** }

描述

who 命令显示关于当前在本地系统上的所有本地用户的信息。显示以下内容: 登录名、tty、登录日期和时间。输入 `who am i` 或 `who am I` 显示您的登录名、tty、您登录的日期和时间。如果用户是从一个远程机器登录的,那么该机器的主机名也会被显示出来。

who 命令也能显示自从线路活动发生以来经过的时间、命令解释器 （shell）的进程标识、登录、注销、重新启动和系统时钟的变化,还能显示由初始化进程生成的其它进程。

who 命令的一般输出格式如下:

```
名称 [状态] 线路 时间 [活动] [Pid] [出口] （主机名）
```

其中:

名称	标识用户的登录名。
状态	表明线路是否对每个用户都是可写的(请参阅 **-T** 标志)。
线路	标识为在 **/dev** 目录中查找到线路名。
时间	表示用户登录的时间。
活动	表示自从在那个用户线路上的最后一个活动发生以来的小时和分钟数。这里的一个 .（点）表明在上一分钟里的线路活动。如果线路保持静止已经超过 24 小时或自从最近系统启动以来还没有被使用,项就被标记旧的。
Pid	标识用户登录 shell 的进程标识。
项	标识进程终止状态（请参阅 **-d** 标志）。要得到关于终止值的更多信息,请参考 **wait** 子例程或 **/usr/include/sys/signal.h** 文件。
出口	标识已结束进程的出口状态（请参阅 **-d** 标志）。
主机名	指出用户从其进行登录的机器名。

要获取信息,**who** 命令通常检查 **/etc/utmp** 文件。如果用 *File* 参数指定另一个文件,**who** 命令改为检查那个文件。这个新文件通常是 **/var/adm/wtmp** 或 **/etc/security/failedlogin** 文件。

如果 *File* 参数指定了不止一个文件名,则只会使用最后的文件名。

注: 这个命令仅标识在本地节点上的用户。

标志

-a 处理 **/etc/utmp** 文件或有全部信息的指定文件。等同于指定 **-bdlprtTu** 标志。

-b 指出最近系统启动的时间和日期。

-d 显示没有被 init 重新生成的所有到期的进程。退出字段用于显示死进程并包含死进程的终止和退出值(由 wait 进程返回的)。(这个标志用于通过察看应用程序返回的错误号来确定一个进程的结束原因。)

-l 列出任何登录进程。

-m 仅显示关于当前终端的信息。**who -m** 命令等同于 **who am i** 和 **who am I** 命令。

-p 列出任何当前活动的和以前已由 init 生成的活动进程。

-q 打印一份在本地系统上的用户和用户数的快速清单。

-r 显示当前进程的运行级别。

-s 仅列出名字、线路和时间字段。这个标志是缺省值；因此，**who** 和 **who -s** 命令是等效的。

-t 显示 root 用户上一次用 **date** 命令对系统时钟做的更改。如果 **date** 命令自从系统安装以来还没有被运行过， **who -t** 命令就不产生输出。

-u 或 **-i** 显示每个当前用户的用户名、tty、登录时间、线路活动和进程标识。

-A 显示在 **/etc/utmp** 文件中的所有记帐项。这些项是通过 **acctwtmp** 命令生成的。

-H 显示一个头（标题）。

-T 或 **-w** 显示 tty 的状态并如下显示谁能够对 tty 写入:

- **+** 对任何人读可写。
- **-** 仅对 root 用户或其所有者可写。
- **?** 遇到线路故障。

出口状态

这个命令返回以下退出值:

0 成功完成。

>0 发生错误。

示例

1. 要显示关于谁正在使用本地系统节点的信息，请输入:

   ```
   who
   ```

   显示类似于以下内容的信息:

   ```
   joe lft/0 Jun 8 08:34
   ```

2. 要显示您的用户名，请输入:

   ```
   who am i
   ```

   显示类似于以下内容的信息:

   ```
   george lft/0 Jun 8 08:34
   ```

3. 要显示登录、注销、系统启动和系统关闭的历史记录，请输入:

   ```
   who /var/adm/wtmp
   ```

   显示类似于以下内容的信息:

   ```
   hank   lft/0   Jun   8   08:34  (ausnix5)
   john   lft/0   Jun   8   08:34  (JIKey)
   mary   lft/0   Jun   8   08:22  (machine.austin.ibm)
   jan    pts4    Jun   8   09:19  (puff.wisc.edu)
   ```

4. 要显示本地系统节点的运行级别，请输入:

   ```
   who  -r
   ```

显示类似于以下内容的信息:

```
. run-level 2 Jun 8 04:15 2 0 s
```

5. 要显示任何当前活动的和以前已由 init 生成的活动进程，请输入:

```
who  -p
```

显示类似于以下内容的信息:

```
srcmstr   .    Jun 8  04:15  old  2896
cron      .    Jun 8  04:15  old  4809
uprintfd  .    Jun 8  04:15  old  5158
```

6. 要通过指定 **-bdlprtTu** 标志来处理 **/var/adm/wtmp** 文件，请输入:

```
who  -a /var/adm/wtmp
```

显示类似于以下内容的信息:

```
   .        system boot Jun 19 10:13
   .        run-level 2 Jun 19 10:13
   .            .       Jun 19 10:14    old
   .            .       Jun 19 10:14    old
   .            .       Jun 19 10:14    old
rc        -     .       Jun 19 10:13    old
   .            .       Jun 19 10:16    old
   .            .       Jun 19 10:14    old
srcmstr   -     .       Jun 19 10:14    old
rctcpip   -     .       Jun 19 10:14    old
rcdce     -     .       Jun 19 10:14    old
rccm      -     .       Jun 19 10:15    old
dceupdt   -     .       Jun 19 10:15    old
rcnfs     -     .       Jun 19 10:15    old
cron      -     .       Jun 19 10:16    old
piobe     -     .       Jun 19 10:16    old
qdaemon   -     .       Jun 19 10:16    old
writesrv  -     .       Jun 19 10:16    old
uprintfd  -     .       Jun 19 10:16    old
   .            .       Jun 19 10:16    old
LOGIN     - lft0        Jun 19 10:16    old
   .            .       Jun 19 10:16    old
   .            .       Jun 19 10:16    old
```

文件

/etc/utmp	包含用户和记帐信息。
/etc/security/failedlogin	包含所有无效登录的历史记录。
/var/adm/wtmp	包含自从上次文件创建以来的所有登录历史记录。
/usr/include/sys/signal.h	包含一个终止值的列表。

相关信息

date 命令、**mesg** 命令、**whoami** 命令、**su** 命令。

wait 子例程。

whoami 命令

用途

显示登录名。

语法

whoami

描述

whoami 命令显示登录名。与使用命令 **who** 并指定 **am i** 不同，**whoami** 命令在有 root 用户权限时也有效，因为它不检查 **/etc/utmp** 文件。

文件

/etc/passwd 包含用户标识。

相关信息

who 命令。

whodo 命令

用途

列出系统中用户正执行的任务。

语法

whodo [**-h**] [**-l**] [*User*]

描述

打印关于一个终端的所有进程及子进程的信息。

缺省情况下，每个活动登录用户的命令所生成的输出将会包含终端名、用户标识、登录日期和时间。输出以日期、时间和机器名开头。这个信息之后是一个与那个用户标识相关的活动进程记录。每个记录显示终端名、进程标识、使用的 CPU 分秒数和进程名。

标志

-h 禁止打印到输出上的标题。

-l 生成长格式的输出。打印系统上当前活动的一个摘要。摘要包含以下内容:

用户 登录用户。

tty 用户使用的 tty 名称。

注册 @
用户登录日期和时间。

空闲 自程序上次试图读取终端开始持续的分钟数。

JCPU 该终端上的所有进程及其子进程使用的系统部件时间。

PCPU 当前活动进程所使用的系统部件时间。

什么 当前进程的名称和参数。

摘要标题行显示当前日期和时间、系统已经启动的时间、登录到系统的用户数。

参数

User 限制为仅输出与 *User* 指定的用户相关所有会话。一次指定的用户名不能超过一个。

出口状态

0 命令成功完成。

>0 发生一个错误。

示例

1. 当在主机 “linguist” 上调用 **whodo** 命令而不带任何标志和参数，输出看起来类似于下面的内容:

```
2002 年 7 月 28 日 16:27:12 星期日
linguist

lft0 jeffg 8:15
         ? 4136  0:00 dtlogin
         ? 3408  4:55 dtsession
         ? 2072  4:37 dtwm
         ? 17310  0:00 dtexec
         ? 20904  5:53 dtterm
     pts/0 22454  0:00 ksh
     pts/0 4360  0:07 ksh
     pts/0 25788  0:00 whodo
         ? 23672  0:00 dtexec
         ? 27536  0:00 dtterm
     pts/3 21508  0:00 ksh
         ? 23888  0:00 dtexec
         ? 24384  2:49 dtterm
     pts/2 24616  0:00 ksh
     pts/2 25002  0:04 ksh
     pts/2 26110  0:00 ksh
         ? 25276  0:00 dtexec
         ? 27090  0:31 dtterm
     pts/1 24232  0:00 ksh
     pts/1 23316  0:01 ksh
         ? 12566  4:23 dtfile
         ? 21458  1:35 dtfile

pts/0 jeffg 8:16
     pts/0 22454  0:00 ksh
     pts/0 4360  0:07 ksh
     pts/0 25788  0:00 whodo
```

```
pts/1 jeffg 17:8
     pts/1 24232  0:00 ksh
     pts/1 23316  0:01 ksh

pts/2 jeffg 17:20
     pts/2 24616  0:00 ksh
     pts/2 25002  0:04 ksh
     pts/2 26110  0:00 ksh

pts/3 root 16:26
     pts/3 21508  0:00 ksh
```

2. 在主机 “linguist” 上的命令 `whodo -l` 产生以下输出:

```
  04:33 下午   向前 20 天, 22 小时, 51 分钟  5 用户
用户     tty            注册 @       空闲      JCPU      PCPU 什么
jeffg    lft0    02 年 07 月 08      21 天                     /usr/sbin/getty /de
jeffg    pts/0   02 年 07 月 08                14:00         7 whodo -l
jeffg    pts/1   02 年 07 月 16      10 天        44         9 /usr/bin/ksh
jeffg    pts/2   02 年 07 月 12         11      8:39         4 /usr/bin/ksh
root     pts/3       04:26 下午          7                     -ksh
```

文件

/usr/sbin/whodo	包含 **whodo** 命令。
/etc/utmp	包含用户列表。

相关信息

ps 命令、**who** 命令。

---

whois 命令

用途

根据用户标识或别名标识一个用户。

语法

whois [**-h** *HostName*] [**.** | **!**] [*****] *Name* [**. . .**]

whois ?

描述

/usr/bin/whois 命令搜索一个用户名目录并显示 *Name* 参数指定的用户标识和别名的信息。**whois** 命令试图到达 ARPANET 主机 `internic.net`，在那里它检查用户名数据库以获取信息。**whois** 命令应该仅能由 ARPANET 上的用户使用。请参考 RFC 812 以获取更完整的信息和对 **whois** 命令的最近更改。

> **注**: 如果您的网络是在美国国内网络上，例如 ARPANET，主机名和 `internic.net` 一样是硬编码的。

Name [**. . .**] 参数代表用户标识、主机名、网络地址或用于执行目录搜索的别名。**whois** 命令执行通配符搜索，查找与任选的 ...（三个句点）之前的字符匹配的所有名称。

标志

.	强制执行“仅按名称”搜索，查找 *Name* 参数指定的名称。
!	显示在 *Name* 参数中指定的别名或句柄标识的帮助信息。
*****	显示一个组或组织的完整会员资格列表。如果有很多成员，这可能要花费一些时间。
?	向 ARPANET 主机请求帮助。
-h *HostName*	指定一个备用的主机名。在 ARPANET 上的缺省主机名是 `internic.net`。通过指定 **-h** *HostName* 标志，可以联系其它主要的 ARPANET 用户名数据库（`nic.ddn.mil`）。

示例

1. 要显示用户名称为 Smith 的 ARPANET 注册用户的相关信息，请输入:

   ```
   whois Smith
   ```

2. 要显示使用句柄 Hobo 的 ARPANET 注册用户的相关信息，请输入:

   ```
   whois  !Hobo
   ```

3. 要显示用户名称为 John Smith 的 ARPANET 注册用户的相关信息，输入:

   ```
   whois  .Smith, John
   ```

4. 要显示用户名称或句柄以字母 HEN 开头的 ARPANET 注册用户的相关信息，请输入:

   ```
   whois HEN  ...
   ```

5. 要获取关于 **whois** 命令的帮助信息，请输入:

   ```
   whois  ?
   ```

相关信息

who 命令。

named.conf 文件格式。

*《AIX 5L V5.2 系统管理指南: 通信与网络》*中的『网络概述』。

---

whom 命令

用途

操作消息处理程序（MH）地址。

语法

whom [**-alias** *File* ...] [**-nocheck** | **-check**] [{ **-draftfolder** **+***Folder* | **-nodraftholder** | *File* }{ **-draftmessage** *Message* | **-draft***File* }]

描述

whom 命令进行以下操作:

- 将一个消息头扩展到一个地址集。
- 列出一条消息的建议收件人地址。
- 验证该地址可由传送服务投递。

注: **whom** 命令不能保证一定能投递到那些被列为可投递的地址。

消息可驻留在一个草稿文件夹或一个文件中。要指定消息驻留在什么地方，请使用 **-draft**、**-draftfolder** 或 **-draftmessage** 标志。

如果不指定 **-draftfolder** 标志或者如果 **$HOME/.mh_profile** 文件中的 `Draft-Folder:` 项没有定义， **whom** 命令会在 MH 目录中搜索 **draft** 文件。在 **-draftfolder +***Folder* 标志之后指定一个与指定 **-draftmessage** 标志一样的消息。

标志

-alias *File*
指定一个用于搜索邮件别名的文件。缺省情况下，系统搜索 **/etc/mh/MailAliases** 文件。

-draft
使用在 *UserMhDirectory***/draft** 文件中的头信息，如果它存在的话。

-draftfolder +*Folder*
使用来自于指定文件夹中草稿消息的头信息。如果指定了一个不存在的草稿文件夹，系统会为您创建一个。

-draftmessage *Message*
使用来自于指定草稿消息的头信息。

-help
列出命令语法、可用的开关（切换）和版本信息。

注: 对于 MH，这个标志的名字必须完全拼写出来。

Message
指定消息草稿。用以下内容来指定消息:

Number 消息号。

cur 或 **.**（句点）
当前消息。这是缺省值。

first 文件夹中的第一条消息。

last 文件夹中的最后一条消息。

next 当前消息之后的消息。

prev 当前消息之前的消息。

-nodraftfolder
撤销 **-draftfolder +***Folder* 标志的上一次出现。

注: 其它两个标志，**-check** 和 **-nocheck**，也是可用的。这些标志对于 **whom** 命令如何完成验证没有影响。**-check** 和 **-nocheck** 标志仅为了兼容而提供。

概要文件项

在 *UserMhDirectory***/.mh_profile** 文件中输入了以下项:

`Draft-Folder:` 为草稿设置缺省的文件夹。
`postproc:` 指定用于公布消息的程序。

示例

要列出并验证一条消息的建议收件人地址，请在各自的提示上输入消息的地址和主题，如下:

```
到:  d77@nostromo
主题:  a test
```

当再一次提示时，输入消息文本:

```
-------输入 初始 文本
test
-------
```

在 whatnow 提示之后，输入 **whom** 命令:

```
whatnow>>> whom
```

然后消息的建议收件人地址显示为:

```
lance...
d77@nostromo...  可投递
```

文件

$HOME/.mh_profile	指定 MH 用户概要文件。
/usr/bin/whom	包含 **whom** 命令。

相关信息

ali 命令、 **post** 命令、 **whatnow** 命令。

mh_alias 文件格式、**mh_profile** 文件格式。

*《AIX 5L V5.2 系统用户指南: 通信与网络》*中的『邮件概述』。

---

wlmassign 命令

用途

手工给“工作负荷管理”类分配进程或取消早先对进程的手工分配。

语法

wlmassign [**-s** | **-S**] [**-u** | *Class_Name*] [*PID_List*] [**-g** *Pgid_List*]

描述

wlmassign 命令:

- 将由进程标识符（pid）和／或进程组标识符（pgid）指定的进程集分配给指定的超类和／或子类，从而覆盖自动类分配或早先的手工分配。
- 取消早先对在 *pid_list* 或 *pgid_list* 中指定的进程的手工分配。

wlmassign 命令允许使用 PID 和／或 PGID 列表来指定进程。这些列表的格式是:

pid[, **pid**[, **pid**[...]]]

或

pgid[, **pgid**[, **pgid**[...]]]

必须指定有效的超类或子类名以手工将目标进程分配给一个类。如果目标类是一个超类，根据该超类的子类的分配规则，每个进程被分配给指定超类的子类中的一个。

手工分配会一直有效（并且进程保持在其手工分配的类中），直到:

- 进程终止
- 工作负荷管理（WLM）被停止。当重新启动 WLM 时，在 WLM 停止时起作用的手工分配丢失了。
- 已给其分配进程的类被删除

- 新的手工分配覆盖了先前的一个分配。
- 用 **-u** 标志取消了对进程的手工分配。
- 进程调用 **exec()** 例程。

要手工将目标进程分配给一个类，必须被指定有效的超类或子类名。分配可以在超类和／或子类级别完成或取消。当对进程的手工分配被取消、或进程调用 **exec()** 时，进程会符合自动分类；如果进程所在的类启用了继承，那么它将保留在那个类里，否则进程将会被根据分配规则重新分类。自动分配（继承和规则）、继承和手工分配之间的交互作用在 *AIX 5L Version 5.2 System Management Concepts: Operating System and Devices* 中的 concept article 中有详细的说明。

对于手工分配:

- 如果 *Class_Name* 是一个超类名，在列表里的进程就会被分配到这个超类中。然后，使用目标超类的子类的分配规则，为每个进程确定子类。
- 如果类名是一个子类名（supername.subname），则缺省情况下进程会被分配到超类和子类中。仅能通过指定 **-S** 标志将进程指定给超类，或仅能通过指定 **-s** 标志将进程指定给子类。

```
wlmassign  super1.sub2  -S  pid1
```

等同于:

```
wlmassign super1 pid1
```

要将进程分配给类，或取消先前的手工分配，用户必须同时具有进程和目标类的权限。这些约束解释为以下内容:

- root 用户可以将任何进程分配给任何类。
- 对一个给定超类的子类上具有管理特权的一个用户（也就是说，用户名或组名与在超类的 **adminuser** 和 **admingroup** 属性中指定的用户名和组名匹配）能手工将任何进程从这个超类其中的一个子类重新指定给该超类的另一个子类。
- 用户能手工将自己的进程（同一个真实或有效用户标识）分配给他们有手工分配特权的类（即，用户名或组名与在这个超类或子类的 **authuser** 和 **authgroup** 属性指定的用户名或组名相匹配）。

这在能手工将进程分配给类的用户中定义了 3 个级别的特权，root 处于最高级别。用户要修改或终止一个手工分配，他们必须至少有与进行上一个手工分配的用户相同的特权级别。

注: **wlmassign** 命令处理当前装入的 WLM 配置。如果当前配置是一个集合，并且对集合中所有配置里都不存在的类进行分配，则当不包含类的配置激活（类被删除）时，该分配将会丢失。

标志

-g *Pgid_list*	显示以下列表是一个 pgid 列表。
-S	指定分配仅在超类级别完成或取消。这个标志是与格式如 *supername.subname* 的子类名一起使用。
-s	指定分配仅在子类级别完成或取消。这个标志是与格式如 *supername.subname* 的子类名一起使用。
-u	取消任何对 pid_list 或 pgid_list 中的进程的手工分配。如果 **-s** 或 **-S** 标志其中的任何一个都没使用，这时会取消超类和子类级别的手工分配。

相关信息

chclass 命令、**lsclass** 命令、**mkclass** 命令和 **rmclass** 命令。

AIX 5L Version 5.2 System Management Concepts: Operating System and Devices 中关于 Worldload Management 的概念主题。

wlmcheck 命令

用途

检查自动分配规则和／或决定具有指定属性集的进程将归类的“工作负荷管理”类。

语法

wlmcheck [**-d** *Config*] [**-a** *Attributes*] [**-q**]

描述

没有参数的 **wlmcheck** 命令给出工作负荷管理（WLM）的状态并进行若干一致性检查:

- 显示 WLM 的当前工作状态（运行／不运行、主动／被动、 rsets 绑定活动、启用总体限制）。
- 显示报告上次载入错误（如果有的话）的状态文件。如果“current”配置是一个集合，这会应用于集合中的所有配置，并且报告 WLM 守护程序记录的消息。
- 检查属性和分配规则文件的一致性（例如，类的存在性、用户名和组名的有效性、应用程序文件名的存在性等等）。

如果没有指定 **-d** *Config* 标志，则对“当前”配置执行检查。

wlmcheck 命令能应用于一个配置集。在这种情况下，上面提及的检查在检查了集合自身之后执行对集合中所有配置的检查。超类名按“config/superclass”格式报告以表明它们所属的正常配置。

用 **-d** *Config* 来指定配置将执行对 *Config* 配置或集合的检查，而不是对“current”。这不更改状态文件报告和 WLM 守护程序日志报告，该日志仅适用于活动配置。

与 **-a** 标志一起使用时，**wlmcheck** 根据当前或指定配置或配置集的规则，显示具有 *Attributes* 指定属性的进程将会分配到的类。*Attributes* 字符串的格式类似于在 *rules* 文件中的一个项，但有以下不同:

- 类字段被省略（它实际上是 **wlmcheck** 的输出）
- 每个字段最多能有一个值。不允许使用排除（!）、属性分组（$）、逗号分隔的列表和通配符。对于 *type* 字段，允许使用与运算符“+”，因为对类型属性，一个进程同时能有几个可能的值。例如，一个进程可以是 32 位的进程并调用 plock，或是 64 位固定优先级的进程。
- 必须至少指定一个字段（有一个不同于连字符“-”的值）。

另外，开始的两个字段是强制性的。其它的字段，如果未提供，则缺省设置为连字符（-），它表示分配规则相应字段中的任何值都匹配。当在属性字符串里的一个或多个字段既不呈现也没有指定为连字符（-）时，字符串就可能匹配多个规则。在这种情况下，**wlmcheck** 显示与所有可能匹配相关的所有类。

有效属性字符串的示例:

```
$ wlmcheck -a "- root system /usr/lib/frame/framemaker - -"
$ wlmcheck -a "- - staff - 32bit+fixed"
$ wlmcheck -a "- bob"
```

标志

-d *Config*　使用 **/etc/wlm/Config**（它可以显示以时间为基础的配置集）中的而不是 **/etc/wlm/current** 中的 WLM 属性文件。

-a *Attributes*　为进程的分类属性传递一个值集用来决定进程将被放进哪个类中。这是一个检查方法，它可以检查分配规则是否正确和是否对进程进行预期的分类。

-q　禁止 WLM 最新活动／更新的状态输出和 WLM 守护程序记录的消息输出（安静方式）。

文件

classes 包含类的名称和定义。
limits 包含强加于类的资源限制。
rules 包含自动分配规则。
shares 包含分配给类的资源共享。

相关信息

chclass 命令、**lsclass** 命令、**mkclass** 命令、**rmclass** 命令。

rules 文件。

wlmcntrl 命令

用途

启动或停止“工作负荷管理器”。

语法

wlmcntrl [[**-a** | **-c** | **-p**] [**-T** [**class** | **proc**] [**-g**] [**-d** *Config_Dir*] [**-o** | **-q**]

wlmcntrl -u [**-S** *Superclass* | **-d** *Config_Dir*]

描述

wlmcntrl 命令停止、启动、更新或查询工作负荷管理器（WLM）的状态。当启动或更新 WLM 时，会预处理用于目标配置的 WLM 属性文件，并且将数据装入内核。WLM 能以两种不同的方式启动:

- 一种是主动方式，这种方式下 WLM 监视和调节各种类中的进程对 CPU、内存和磁盘 I/O 的使用率。
- 一种是被动方式，这种方式下 WLM 仅监视资源使用率而不干涉标准操作系统资源分配机制。

主动方式是 WLM 通常的操作方式。

类，在 **classes**、**limits** 和 **shares** 文件中分别描述它们的限制和份额。自动分配规则来自于 **rules** 文件。用于 WLM 配置的超类 **Config** 的类属性文件在 **/etc/wlm/Config** 子目录中。用于配置 Config 的超类 Super 的子类的属性文件位于 **/etc/wlm/Config/Super** 中。附随操作系统一起提供的标准配置在 **/etc/wlm/standard** 中。当前配置是在目录里用符号链接 **/etc/wlm/current** 指向的那一个。

当没有使用 **-d** *Config_dir* 标志时，**wlmcntrl** 使用在目录中符号链接 **/etc/wlm/current** 所指向的配置文件。

当使用了 **-d** *Config_dir* 标志时，**wlmcntrl** 使用在 **/etc/wlm/Config_dir** 中的配置文件并更新 **/etc/wlm/current** 符号链接以指向 **/etc/wlm/Config_dir**，使 **/etc/wlm/Config_dir** 成为当前配置。这是推荐的使 **/etc/wlm/Config_dir** 成为当前配置的方法。

当用 **-u** 标志更新 WLM 时，用 **-d** 标志可将一个空字符串作为 *Config_dir* 传递:

```
wlmcntrl -u -d ""
```

将会把当前配置的分配规则简单地刷新（重新装入）到内核中而不重新装入类定义。当 WLM 的一个早先的激活检测到不能访问一些应用程序文件时，这个会很有用。在系统管理员用规则或文件修复了问题后，这个命令仅能用于重新装入规则。

WLM 的配置 **Config** 也可以是一个基于时间的配置集合，在这个情况下子目录 **/etc/wlm/Config** 不包含属性文件，但包含配置列表和应用它们时的星期中的时间。属性文件仍然存在于集合的每个常规配置的子目录中。当 WLM 启动或更新时，这时一个集合、一个守护程序负责在一个适用的集合更改时切换集合的常规配置。

标志

-a
以主动方式启动 WLM 或将 WLM 从被动方式切换到主动方式。当除了 **-d**，**-g** 或**-T** 以外，没有指定别的标志时，这是缺省值。

-c
以“仅 CPU”方式启动 WLM 或将 WLM 从任何方式切换到“仅 CPU”方式。在这种方式里，WLM 对所有资源进行记帐，但仅调节 CPU 资源。

-d *Config_dir*
将 **/etc/wlm/Config_dir** 作为一个 WLM 配置的备用目录（包含类文件、限制文件、共享文件和规则文件）或配置集合（包含配置列表和它们应用时的时间范围）。这会使 **/etc/wlm/Config_dir** 成为当前配置。当以主动方式、“仅 CPU”或被动方式启动 WLM 时，或者当更新 WLM 时，这个标志是有效的。这个标志不能与 **-o** 和 **-q** 标志一起使用，或者不能在从一个方式（在主动、“仅 CPU”和被动之间）切换到另一个方式时使用。

-g
指示 WLM 忽略任何潜在的资源集合绑定。这意味着所有类都有权访问系统的整个资源集，而不管它们是否在使用一个受限制的资源集。

-o
停止“工作负荷管理器”。

-p
以被动方式启动 WLM 或将 WLM 从任何方式切换到被动方式。在这个方式下，WLM 对所有资源进行记帐，但不调节资源。

-q 查询 WLM 的当前状态。返回:

0 WLM 正在以主动方式运行。

1 WLM 没有启动。

2 WLM 正在以被动方式运行。

3 WLM 正在以主动方式运行而没有绑定 rset。

4 WLM 正在以被动方式运行而没有绑定 rset。

5 WLM 正在以“仅 CPU”的主动方式运行。

6 WLM 正在以“仅 CPU”的主动方式运行而没有绑定 rset。

16 WLM 正在以主动方式运行，进程总记帐关闭。

18 WLM 正在以被动方式运行，进程总记帐关闭。

19 WLM 正在以主动方式运行而没有绑定 rset，进程总记帐关闭。

20 WLM 正在以被动方式运行而没有绑定 rset，进程总记帐关闭。

21 WLM 正在以“仅 CPU”的主动方式运行，进程总记帐关闭。

22 WLM 正在以“仅 CPU”的主动方式运行而没有绑定 rset，进程总记帐关闭。

32 WLM 正在以主动方式运行，类总记帐关闭。

34 WLM 正在以被动方式运行，类总记帐关闭。

35 WLM 正在以主动方式运行而没有绑定 rset，类总记帐关闭。

36 WLM 正在以被动方式运行而没有绑定 rset，类总记帐关闭。

37 WLM 正在以“仅 CPU”的主动方式运行，类总记帐关闭。

38 WLM 正在以“仅 CPU”的主动方式运行而没有绑定 rset，类总记帐关闭。

48 WLM 正在以主动方式运行，类和进程总记帐关闭。

50 WLM 正在以被动方式运行，类和进程总记帐关闭。

51 WLM 正在以主动方式运行而没有绑定 rset，类和进程总记帐关闭。

52 WLM 正在以被动方式运行而没有绑定 rset，类和进程总记帐关闭。

53 WLM 正在以“仅 CPU”的主动方式运行，类和进程总记帐关闭。

54 WLM 正在以“仅 CPU”的主动方式运行而没有绑定 rset，类和进程总记帐关闭。

一条指示 WLM 当前状态的消息打印到 STDOUT 上。

-S *Superclass* 请求对限制为超类的子类的 WLM 更新。将该标志与 **-u** 标志一起使用。如果正在运行的配置是一个基于时间的配置集合，“超类”必须以“config/Superclass”的格式给出，其中“config”是“超类”所属集合的常规配置。如果“config”是集合的当前活动配置，变化将会立即生效，否则它们会在下次激活“config”时生效。

-T 类和进程合计都禁用，限制记帐和调节。

-T class 只禁用类合计，限制记帐和调节。

-T proc 只禁用进程合计，限制记帐和调节。

-u 更新 WLM。单个更新操作能够更改现有类的属性、限制和共享，和／或添加或移除类。如果正在运行的配置是一个集合，这个操作将该集合描述与所有该集合的配置内容一起刷新。一个拥有 root 权限的用户能够使用更新来切换至一个备用配置或配置集合。而且超类级管理员也能使用更新来更新仅他有管理访问权的超类的子类（使用 **-S** 标志）。

安全性

访问控制: 启动、停止、从一个方式切换至另一个方式，而且更新超类或一个配置集需要 root 特权。更新一个给定超类的子类仅需要管理用户或管理组特权（超类管理员）。任何用户都能够查询 WLM 状态。

文件

classes	包含类的名称和定义。
limits	包含强加到类上的资源限制。
rules	包含自动分配规则。
shares	包含分配到类的资源份额。
description	包含每个配置的描述文本。
groupings	包含用于配置的属性值分组。

相关信息

chclass 命令、**confsetcntrl** 命令、**lsclass** 命令、**lswlmconf** 命令、**mkclass** 命令和 **rmclass** 命令。

AIX 5L Version 5.2 System Management Concepts: Operating System and Devices 中的“工作负荷管理”concept article。

---

wlmmon 和 wlmperf 命令

用途

wlmmon 和 **wlmperf** 命令根据类来提供工作负荷管理（WLM）的图形化视图。

语法

wlmmon

wlmperf

描述

wlmmon 和 **wlmperf** 命令生成系统 WLM 活动的资源使用情况报告。**wlmperf** 命令，它是性能工具箱（PTX）的一部分，能根据由 PTX 守护程序产生的趋势记录生成涵盖分钟、小时、日、星期和月份为周期的报告。**wlmmon** 命令，它与基本 AIX 一起装入，仅生成最近 24 小时时间段内的报告而且没有使用选项。能够生成三种类型的视觉报告:

- 快照显示
- 详细显示
- 表格显示

能够将报告类型定制为在特定时间周期上涵盖指定的 WLM 类。另外，可以针对任何选定的显示类型比较（显示趋势）两个不同周期的 WLM 活动。

这些报告是从与用 **wlmstat** 命令一样的机制收集的数据中生成的。然而，**wlmmon** 和 **wlmperf** 命令使用守护程序产生的记录，该守护程序必须一直运行以收集 WLM 数据。对 **wlmmon** 命令来说，这个守护程序称为 **xmwlm**，并随基本 AIX 一起提供。对于 **wlmperf** 命令来说，这个守护程序称为 **xmtrend** 并随 PTX 一起提供。

分析概述

wlmstat 命令提供一个 WLM 活动的“每秒”视图时，它不适合于长期的分析。为了补充 **wlmstat** 命令，**wlmmon** 和 **wlmperf** 命令提供在更长时间周期内的 WLM 活动报告，而只有最小限度的系统影响。由这个工具所生成的报告是以由相关连的记录守护程序所完成的采样为基础的。这些守护程序以很高的速率（用秒来测量）对 WLM 和系统统计信息进行采样，但只以低速率（用分来测量）来记录超级样本值。对每个在记录周期内收集的统计信息来说，这些值表示最小、最大、平均和标准偏差值。

WLM 报告浏览器

在启动的基础上，显示“报告浏览器”。浏览器显示一个报告集合。用户可配置的显示类型，是以被选择用来生成报告的属性为基础的。

“报告浏览器”菜单选项:

新建 创建报告

关闭 退出浏览器

打开 显示一个选中的报告

属性 允许查看和编辑一个报告的属性

删除 删除一个选中的报告

报告属性面板

“报告属性面板”允许用户定义那些控制 WLM 数据的实际图形表示的属性。在这个面板里有三个标记过的选项卡窗格:

- 常规菜单
- 层 / 类菜单
- 高级菜单

报告名 一个用于命名报告的用户可编辑字段。报告应该以 **.rpt** 扩展名结束。

常规菜单: 第一个标记过的选项卡窗格允许用户编辑一个显示的常规属性，如下:

趋势框 表明将会生成一个选中类型的趋势报告。趋势报告允许在同一显示上进行两个不同时间周期的比较。选择这个框以使您能够编辑**第一个时间间隔结束**字段。

资源 允许选择要显示的 WLM 资源（例如 CPU 或内存）。请参考 WLM 用户指南和文档以获取关于可管理资源的信息。

时间间隔的宽度 表示涵盖于任何显示类型的时间周期，这些周期或从报告中最新的可用值开始测量，或从用户输入时间选择开始。从这个菜单中选择了时间间隔宽度。可用的选择是变化的，取决于使用的工具:

wlmmon
用于分钟和小时的多项选择

wlmperf
用于分钟、小时、日、星期和月份的多项选择

第一个时间间隔的结束 显示对生成一个趋势报告感兴趣的周期的结束时间。第一个时间间隔始终表示一个比上个时间间隔更早的时间周期。如果选择了“趋势框”，则仅能编辑这个字段。

最后一个时间间隔的结束　表示对趋势和非趋势报告感兴趣的周期的结束时间。最后的时间间隔始终表示用于生成一个显示报告的最新时间范围。存在这个字段的两个互斥的选择选项:

Latest　使用记录中可用的最新时间作为报告的结束时间。

Selected Time
允许用户输入最后时间间隔的结束时间。

层／类菜单:　第二个选项卡窗格允许用户定义包含在一个报告中的 WLM 层和类设置。

作用域　允许用户为显示选择一个以层或以类为基础的作用域。这个显示根据 AIX 发行版（AIX 4.3 类对 AIX 5.1 超类和子类定义）之间的层和类概念的变化而变化。

选择　允许选择包含和排除记录中的可用 WLM 层或类。

高级菜单:　“报告属性面板”里第三个选项卡窗格提供了高级选项，主要用于快照显示。对快照来说，提供给用户选择用于为显示着色的互斥方法。选项 1 忽略 WLM 环境中的配置里定义的最小和最大设置。选项 2 使用该最小和最大设置。

报告显示

有三种报告显示类型:

- 快照显示
- 详细显示
- 表格显示

三种显示中的每一种都有以下公共元素:

WLM 控制台　用于选择打印或关闭显示。

时间周期　显示在“报告属性面板”中定义的时间周期。对比较两个时间周期的趋势报告来说，显现两个时间显示。

层列　显示与一个类关联的层类数。对于 AIX 5.1，这个列有两个项，超类层（左）和子类层（右）。

类列　显示类名。

资源列　显示以已选择的图形报告的选择类型为基础的资源信息。将会在下面描述它们。

状态区域　显示全局系统性能度量的一个集合，这个度量也记录以在分析方面给与帮助。所显示的集合在各 AIX 发行版之间可能不同，但都会包含诸如运行、队列、交换队列和 CPU 忙之类的度量值。

快照显示:　这个显示是一个快速的“一切正常吗？”概述。该显示主要显示以用户指定的与已定义的目标份额之间的偏差为基础的类资源关系。要选择或调整这个显示的偏差参数，请使用“报告属性面板高级菜单”。

如果趋于用快照显示，则以前的（第一个）分析周期用箭头来指示，该箭头从以前的评估指向后来的（第二个）评估。如果在周期之间没有变化，就不显示箭头。

详细显示:　在这个显示中，资源列以条形图样式显示，与在指定时间周期上测量的资源活动的百分比一起显示。百分比是以由 WLM 子系统定义的总系统资源为基础计算出来的。如果趋于详细显示，则后来的（第二个）测量将显示在以前的（第一个）测量时间间隔之上。

表格显示:　显示报告的第三种类型是一个列表报告。在这个报告里，提供了以下字段:

采样数　这个周期中记录的样本数。

份额值　WLM 计算的份额值目标

平均值	在样本周期上计算平均值
标准偏差	计算标准偏差
定义最小值	在 WLM 限制中定义的类最小值
观察最小值	在时间周期上实际观察到的最小值
定义软性最大值	在 WLM 限制中定义的类软性最大值
定义硬性最大值	在 WLM 限制里定义的类硬性最大值
观察最大值	在时间周期上实际观察到的最大值

守护程序记录和配置

守护程序在 **/etc/perf/wlm** 目录中创建记录。对于基本的 AIX 工具 **wlmmon** 来说，这些记录限于最后的 24 小时周期。

对于“性能工具箱”工具 **wlmperf** 来说，这些记录限于 1 年。对于 PTX 来说，使用 **xmtrend** 守护程序并使用一个用于记录首选项的配置文件。这个配置文件的与 WLM 相关记录的一个样本位于 **/usr/lpp/perfagent.server/xmtrend_wlm.cf**。记录定制、启动和操作，与那些对 **xmtrend** 守护程序的描述一样。

对于基本 AIX 来说，使用 **xmwlm** 守护程序并且不能定制。

对于要创建的记录来说，**/etc/perf/wlm** 目录必须获得足够的磁盘分配，至少允许 10 MB 磁盘空间。另外，应该从一个 **/etc/inittab** 项启动守护程序以使记录能在系统重新启动之后自动重新启动。无论 WLM 子系统是在主动方式、被动方式或禁用（关闭）方式，守护程序都将运行。然而，当 WLM 关闭时记录活动是被限制的。

先决条件

Java 1.3 **perfagent.tools**

出口状态

如果没有找到 WLM 记录，工具会发出一个警告消息。

文件

/usr/bin/wlmmon	基本 AIX
/usr/bin/xmwlm	基本 AIX
/usr/bin/wlmperf	性能工具箱
/usr/lpp/perfagent.server/xmtrend.cf	性能工具箱
wlmmon 和 **xmwlm**	在 **perfagent.tools** 文件集中。
wlmperf 和 **xmtrend**	仅与“性能工具箱”产品介质在一起时可用。

相关信息

wlmstat、**wlmcntrl** 和 **topas** 命令。

wlmstat 命令

用途

显示 WLM 每类资源使用使用率统计信息。

语法

wlmstat [**-l** *Class* | **-t** *Tier*] [**-S** | **-s**] [**-c** | **-m** | **-b**] [**-B** *Device*] [**-T**] [**-a**] [**-w**] [**-v**] [*Interval*] [*Count*]

wlmstat [**-l** *Class* | **-t** *Tier*] [**-c** | **-m** | **-b**] [**-u**] [*Interval*] [*Count*]

描述

wlmstat 命令用符号的形式显示从内核检索的 WLM 数据结构的内容。如果指定 *Count*，**wlmstat** 循环 *Count* 次并在显示每个块操作之后睡眠 *Interval* 秒。如果没有指定 *Interval* 和 *Count*，则产生一个输出报告。如果指定 *Interval* 但没有给出 *Count*，**wlmstat** 在给定的时间间隔内连续地输出结果直到被一个信号（SIGINTR、SIGQUIT、SIGKILL 等等）停止。缺省情况下，**wlmstat** 用于显示每个超类和子类的所有资源的统计信息。可以指定标志来将统计信息焦点缩小到某个资源类型、层、超类或子类，并更改输出格式。

标志

-a
给出用于子类的绝对数字（相对于整个系统可用资源总数），用 0.01% 的解析。缺省情况下，显示的用于子类的数字是超类消耗的资源总数的一个百分比，用 1% 的解析。例如，如果一个超类有 7% 的 CPU 目标且被不带 **-a** 的 **wlmstat** 显示的用于一个子类的 CPU 百分比是 5%，带有 **-a** 的 **wlmstat** 显示用于子类的 CPU 百分比是 0.35%。

-b
仅显示磁盘 I/O 统计信息。

-B *Device*
显示磁盘 I/O 设备统计信息。传递一个空字符串（-B ""）显示类访问的所有磁盘的统计信息。

-c
仅显示 CPU 统计信息。

-l *Class*
显示用于 *Class* 名的统计信息。如果没有被指定，所有类与相应字段的一个摘要一起显示。

-m
仅显示物理内存统计信息。

-s
仅显示子类统计信息。

-S
仅显示超类统计信息。

-t *Tier*
显示仅用于指定的 *Tier* 的统计信息。

-T
显示自从 WLM 启动或类被创建以来资源使用率的总数，不管哪个是更迟创建的。单位是:

CPU 一个类消耗的总 CPU 时间，以毫秒记。

MEM 未使用。

DKIO 一个类发送／接收的为所有磁盘设备所访问的 512 字节块的总数。

-v

指定详细方式。这个标志，意在用于故障查找，也显示一些类属性、资源份额及限制和其它 WLM 参数，包括意在用于 AIX 支持工作人员的内部参数值。用户可能对下面的信息感兴趣:

列头	描述
类	类名
tr	层数（0 到 9）
i	继承属性值: 0 = no, 1 = yes。
#pr	类中的进程数。如果一个类没有分配（0）给它的进程，在其它列中显示的值可能是没有意义的。
CPU	类的 CPU 使用率（%）。
MEM	类的物理内存使用率（%）。
DKIO	类的磁盘 IO 带宽使用率（%）。
sha	份额数（ “-” 表示 -1）
min	资源最小限制（%）
smx	资源软性最大限制（%）
hmx	资源硬性最大限制（%）
des	（期望的）: WLM 使用份额数（%）计算的百分比目标（目标）
npg	类拥有的内存页数。

其它列仅属于内部使用且对管理员和最终用户来说没有意义。这个格式与一个资源选择器（**-c**，**-m**，或 **-b**）一起使用会比较好，否则行可能太长而不适合于显示终端的一行。

-w

显示内存*最高峰*，即自从 WLM 启动或类创建（无论哪个最后发生）后，一个类在任何给定时间内在内存中所拥有的页面最大值。

-u

显示每层和未使用的总资源。

显示

结果制成表格，有以下字段:

名称	类名
CPU	类消耗的总 CPU 时间百分比。
MEM	类消耗的物理内存百分比。
DKIO	类消耗的磁盘 IO 带宽百分比。这个数是类访问的所有磁盘设备上磁盘带宽的平均值，这个值通常不很重要。例如如果一个类消耗了磁盘带宽的 80% 和两个其它磁盘带宽的 5%，DKIO 列将显示 30%。有关每个设备使用率的详细信息，请使用 -B 设备选项。

示例

1. 要立刻获得 WLM 活动打印输出，请输入:

   ```
   wlmstat
   ```

这个命令产生以下输出:

```
                 CLASS CPU MEM DKIO
          Unclassified   0   0   0
             Unmanaged   0   0   0
               Default   0   0   0
                Shared   0   0   0
                System   0   0   0
                class1  12   0   0
      class1.Default     4   0   0
       class1.Shared     0   0   0
    class1.subclass1     4   0   0
    class1.subclass2     4   0   0
                class2  12   0   0
      class2.Default     4   0   0
       class2.Shared     0   0   0
    class2.subclass1     4   0   0
    class2.subclass2     4   0   0
```

2. 要获得超类 **class1** 的一个报告，请输入:

```
wlmstat -l class1
```

这个命令产生以下输出:

```
                 CLASS CPU MEM DKIO

                class1  12   0   0
      class1.Default     4   0   0
       class1.Shared     0   0   0
    class1.subclass1     4   0   0
    class1.subclass2     4   0   0
```

3. 要获得一分钟内每 10 秒更新一次的子类 **sclass1.subclass2** 的一个报告，请输入:

```
wlmstat -l class1.subclass2 10 6
```

这个命令产生以下输出:

```
                 CLASS CPU MEM DKIO
    class1.subclass2     4   0   0
    class1.subclass2     4   0   0
    class1.subclass2     4   0   0
    class1.subclass2     4   0   0
    class1.subclass2     4   0   0
    class1.subclass2     4   0   0
```

错误

如果 WLM 没有启动则 **wlmstat** 发出一个警告消息。

相关信息

wlmcntrl 命令。

---

write 命令

用途

打开一条通信线路以实时地将消息发送给系统上的其它用户。

语法

要查询主机上来自用户的等候应答的所有消息并以其句柄显示，请输入以下命令:

write -q [**-n** *Host*]

要应答由“实用程序”或“Shell 脚本”发送的“消息”，或“重新显示与给定句柄关联的消息”，请输入以下命令:

write -h*Handle,* { ok | cancel | query } [**-n** *Host*]

要有选择地在另一个主机或特别的设备上给用户发送消息，请输入以下命令:

write [**-r**] { [**-n** *Host*] *User* | *User***@***Host* } [*Line*]

描述

write 命令使消息能够在系统上实时发送。它提供与另一个登录用户类似交谈一样的通信。每个用户交替地从其它工作站发送或接收简短消息。通过在一个文件中输入完整的消息然后将该文件作为输入重定向到 **write** 命令就能发送长消息。

对另一个用户（由 *User* 参数指定），要接收消息，用户必须登录并消息许可权没有被拒绝。当一条消息被发送到一个未登录的用户时，会显示消息用户未登录。如果消息发送给一个用户，该用户通过将 **mesg** 命令设置为 no 而具有拒绝的消息许可权的用户，则显示消息 `write`: 许可权拒绝。

当发出 **write** 命令时，它会立即发送以下消息，同时带有消息收件人或目标的声音提醒（ASCII BEL 字符）:

```
消息 来自于 SenderID 在 SenderHostname 上（ttynn） [日期] ...
```

随着连接成功，**write** 命令发送两个 ASCII BEL 字符到两个工作站。蜂鸣声提醒发送方消息可以开始而且它提醒接收用户有消息进入。

当按下一次 Enter 键时发送一行。从发送方到接收方的通信链路保持打开并且发送会继续直到 Ctrl-D 组合键结束发送链路。随后将文本结束字符（<EOT>）发送到目标工作站并且终止 **write** 命令方式。

接收用户或目标用户能够通过向始发用户发送 **write** 命令进行响应。这打开了一条从接收方返回发送方的通信线路，使得消息响应能够返回。对于此类型的交换，下面的约定很有用: 当第一次写给其它用户时，在发送任何文本之前等待响应。用诸如 `o`（over）的信号结束消息以提醒其它人应答。当对话完成时使用 `oo`（结束并离开）。

如果在一行的开始有字符 `!`（感叹号），**write** 命令调用 shell 将行的剩余部分作为命令来执行。

当将一条消息写到一位登录到多于一个工作站或多重使用多于一个进程的用户时，**write** 命令使用在 **/etc/utmp** 文件中找到的第一个登录实例作为消息交付点（通常是登录或控制台 shell），并且提示以下消息:

```
用户标识已在多于一个位置登录。
您已连接到“工作站”。
其它位置是:
工作站
```

当接收到这条消息时，如果想把消息发送到一个不同于初始登录位置的位置，可以通过指定位置（例如，tty00）的 *Line* 来在不同位置联系目标用户。

允许写给其它用户的许可权由特殊用户用 **mesg** 命令授权或拒绝。一些命令在运行时拒绝消息许可权以阻止对其输出的干扰。具有 root 用户权限的用户能写给任何工作站而不考虑工作站的消息许可权。

可以用 **write** 命令同在其它主机上的用户对话。可以通过使用 **-n***HostName* 标志或 *User*@*Host* 参数标识在远程主机上的用户。为了写给远程主机上的用户，**writesrv** 守护程序必须同时运行在当前主机和远程主机上。

qdaemon 守护程序也用 **write** 命令来将消息发送到在其它主机上的用户并等待应答。仅有三种有效应答:

`ok`	原始写操作以 0 状态退出。
`cancel`	原始写操作以 1 状态退出。
`query`	显示与给定句柄相关联的消息。

参数

User	指定接收消息文本人的用户标识。
User@*Host*	指定接收消息文本人的用户标识和远程主机。
Line	联系在另一个位置（例如，`tty00`）的目标用户。

标志

-h *Handle*,*Reply*	通过使用带有应答选项的实用程序或 shell 脚本来应答用 write 发送的消息。要用于 *Handle* 变量的值在内部生成并在原始消息文本中提供给用户。应答可能是 `ok`、`cancel` 或 `query`。
-n*Host*	指定一个远程主机。*Host* 变量可以是别名或因特网地址。
-q	查询所有消息并等待一个主机上用户的应答，并根据它们的句柄来显示它们。
-r	生成消息句柄，把它放到消息头中，发送消息并等待应答。这个标志由 **qdaemon** 守护程序用于操作程序消息并能插入到 shell 脚本中。它不用于交互式对话。出口状态 0 表示应答是 `ok`，出口状态 1 表示应答是 `cancel`，出口状态 2 表示无法联系到用户。

注:

1. 为了使任何标志都有效，**writesrv** 守护程序必须运行在目标主机上。如果不是正在用 **-n** 标志或 **@***Host*，而是在用 **-h**、**-q** 或 **-r**，那么 **writesrv** 守护程序必须运行在当前主机上。
2. 如果机器上没有安装 TCP/IP，但是设置了 *HostName*，那么为了使用 **write** 命令和 **-h**、**-q** 或 **-r** 标志同本地用户对话，必须把当前主机名附加到 **/etc/hosts** 文件中 `loopback` 项的末尾。原始项应该是:

   ```
   127.0.0.1 loopback LocalHostName
   ```

   新项应该是:

   ```
   127.0.0.1 loopback LocalHostName HostName
   ```

出口状态

这个命令返回以下出口值:

0	成功完成。
>0	目标用户没有登录或拒绝许可权。

示例

1. 要将一条消息写到登录用户，请输入:

   ```
   write june
   ```

   按 Enter 键并输入，

   ```
   I need to see you! Meet me in the computer room at 12:30.
   ```

然后按 Ctrl-D 组合键以终止 **write** 命令方式。

如果您的用户标识是 karen 并且您正在使用工作站 tty3，june 的工作站显示:

```
消息 来自于 karen 在 trek tty3 8 月 17 11:55:24  ...
I need to see you!  Meet me in the computer room at 12:30.
<EOT>
```

2. 要保持一个对话，请输入:

```
write june
```

按 Enter 键并输入，

```
Meet me in the computer room at 12:30.
o
```

此命令启动对话。下一行开始的 o 表示消息结束。它告诉 June 您正在等待响应。如果希望继续，不要按 Ctrl-D。

现在 June 通过输入以下内容进行应答:

```
write karen
```

按 Enter 键并输入，

```
I'm running tests at 12:30. Can we meet at 3?
o
```

您可能回应:

```
OK--the computer room at 3.
oo
```

oo 指*结束并离开*，告诉 June 您没有什么其它要说的了。如果 June 也是以 oo 结束，那么你们都按 Ctrl-D 来结束对话。

3. 要把一条已准备的消息写给某人，请输入:

```
write june < message.text
```

它将 **message.text** 文件的内容写到 june 的工作站。

4. 要写给使用特定工作站的那个人，请输入:

```
write -n console
```

按 Enter 键并输入，

```
The printer in building 998 has jammed.
Please send help.
```

然后按 Ctrl-D 组合键。

这将消息写给登录到工作站 /dev/console 的人。

5. 要将消息发送到在主机 partya 上的用户 spuds，请输入:

```
write -n partya spuds
```

按 Enter 键并输入，

```
Your new tape has just arrived,
come see me to pick it up.
Thanks!
```

然后按 Ctrl-D 组合键。

或者

```
write spuds@partya
```

按 Enter 键并输入,

```
Your new tape has just arrived,
come see me to pick it up.
Thanks!
```

然后按 Ctrl-D 组合键。

6. 这里是一个 **qdaemon** 守护程序发送消息的示例:

```
消息 来自 mary 在 trek 上（tty10） 8 月 17 10:03:34 ...
使用 “write -h 6398492,reply” 来 应答
请在 rmt0 中插入 5 号 磁带。
<EOT>
```

要肯定地答复，请输入:

```
write -h 6398492,ok
```

然后按 Ctrl-D 组合键。

要否定地答复，请输入:

```
write -h 6398492,cancel
```

然后按 Ctrl-D 组合键。

使用 **-h** 标志时，没有必要提供主机名或用户标识。该信息通过句柄跟踪。

文件

/etc/hosts
包含 TCP/IP 主机信息。

/etc/utmp
包含用于 **who**、**write** 和 **login** 命令的用户和记帐信息。

相关信息

mesg 命令、**wall** 命令、**who** 命令、**writesrv** 命令。

*《AIX 5L V5.2 系统用户指南: 操作系统与设备》*中的『shell 概述』。

---

writesrv 守护程序

用途

允许用户将消息发送到远程系统并从远程系统接收消息。

语法

writesrv

描述

writesrv 守护程序允许用户将消息发送给在远程系统上的用户并用 **write** 命令接收远程系统上用户的响应。

writesrv 实用程序接收来自 **write** 命令的输入请求并创建服务器进程来处理这个请求。这个服务器进程与客户机进程（**write**）进行通信并提供任何请求的服务。

要执行这些服务，**writesrv** 守护程序创建一个连接到 **/etc/services** 文件中定义的端口的套接字。所有的服务请求都作为消息发送到这个套接字。

> **注**: 如果 **writesrv** 守护程序异常终止（例如系统崩溃、电源故障或 **kill -9** 命令），必须被手工清除 **/var/spool/writesrv** 目录以除去任何由 **writesrv** 守护程序留下的文件。

示例

1. 要从 **/etc/rc** 脚本启动 **writesrv** 守护程序，请输入:

   ```
   /usr/sbin/writesrv
   ```

   writesrv 守护程序从 **/etc/rc** 脚本启动。这是通常启动守护程序的方法。
2. 要用系统资源控制器（SRC）启动 **writesrv** 守护程序，请输入:

   ```
   startsrc -s writesrv &
   ```

   该命令使用 SRC 启动 **writesrv** 守护程序。

文件

/etc/services 包含网络服务目录。

相关信息

kill 命令、**write** 命令

AIX 5L Version 5.2 Guide to Printers and Printing 中的 Using Remote Host Access for Printing、Printer Overview for System Management 和 Remote Printing Overview。

AIX 5L Version 5.2 System Management Concepts: Operating System and Devices 中的 System Resource Controller Overview。

---

wsm 命令

用途

启动“基于 Web 的系统管理器”客户机会话。

语法

/usr/websm/bin/wsm -host *managing host*

/usr/websm/bin/wsm -lang *language*

/usr/websm/bin/wsm -port *port number*

/usr/websm/bin/wsm -profile *pathname of preference file*

/usr/websm/bin/wsm -user *username*

/usr/websm/bin/wsm -DdefaultTurners=*value*

/usr/websm/bin/wsm -DdrawTreeLine=*value*

/usr/websm/bin/wsm -Ddatadir=*path*

描述

wsm 命令用于启动“基于 Web 的系统管理器”客户机会话。

注: 必须指定该命令的全路径名，**/usr/websm/bin/wsm**。

标志

-host *managing host*	强制“基于 Web 的系统管理器”初始连接到指定主机。即使在运行“基于 Web 的系统管理器”时能轻易地管理其它主机，这个选项也允许用指定主机上设置的首选项来启动“基于 Web 的系统管理器”。
-lang *language*	指定显示消息用的语言。如果没有安装 **sysmgt.msg.***Language***.websm.apps** 文件集，消息将会以英文显示。
-port *port number*	使“基于 Web 的系统管理器”使用指定端口连接到任何其它主机。这个使用的端口号必须与在 **/etc/services** 文件中指定的用于 **wsmserver** 服务的受管机器上的端口号相匹配。
-profile *pathname of preference file*	指定一个备用的首选项文件。缺省首选项文件将会是一个在用户根目录中名为 **WebSM.pref** 的文件。使用这个选项使用户能够使用不同的首选项文件。如果用户管理用于不同客户机的不同机器集，这可能很有用。 **注**: 从本地机器或从 **-host** 参数指定的机器读取首选项文件。
-user *username*	使“基于 Web 的系统管理器”以给定用户名运行。将提示输入用户密码。
-DdefaultTurners=*value*	当值是 `true` 时，使用“Java 外观和感觉”调试器替代 Windows 调试器用于在“导航区域”和“内容区域”中的父树节点。在树对象之间没有绘制有角度的线。
-DdrawTreeLine=*value*	当值是 `true` 和 **-DdefaultTurners**=`true` 时，会导致在“导航区域”和“内容区域”中的树对象之间绘制有角度的线。
-Ddatadir=*path*	指定备用目录以寻找通常能在 **/var/websm/config/user_settings** 中找到的配置文件。

示例

1. 要指定一个备用首选项文件，请输入:

   `/usr/websm/bin/wsm -profile` *`pathname of preference file`*

2. 要指定一个备用配置文件，请输入:

   `/usr/websm/bin/wsm -Ddatadir=`*`pathname`*

相关信息

要获取关于安装“基于 Web 的系统管理器”的信息，请参阅*《AIX 5L V5.2 基于 Web 的系统管理器管理指南》*中的『第 2 章 安装和系统要求』。

---

wsmaccess 命令

用途

包装程序使用 **wsm** 命令来启用“辅助功能”部件。

语法

/usr/websm/bin/wsmaccess -host *managing hos*

/usr/websm/bin/wsmaccess -lang *language*

/usr/websm/bin/wsmaccess -port *port number*

/usr/websm/bin/wsmaccess -profile *pathname of preference file*

/usr/websm/bin/wsmaccess -user *username*

/usr/websm/bin/wsmaccess -DdefaultTurners=*value*

/usr/websm/bin/wsmaccess -DdrawTreeLine=*value*

/usr/websm/bin/wsmaccess -Ddatadir=*path*

描述

包装程序使用 **wsm** 命令以启用“辅助功能”部件。

注: 必须指定该命令的全路径名，**/usr/websm/bin/wsmaccess**。

标志

-host *managing host*	强制“基于 Web 的系统管理器”初始连接到指定主机。即使在运行“基于 Web 的系统管理器”时能轻易地管理其它主机，这个选项也允许用指定主机上设置的首选项来启动“基于 Web 的系统管理器”。
-lang *language*	指定显示消息用的语言。如果没有安装 **sysmgt.msg.***Language***.websm.apps** 文件集，消息将会以英文显示。
-port *port number*	使“基于 Web 的系统管理器”使用指定端口连接到任何其它主机。这个使用的端口号必须与在 **/etc/services** 文件中指定的用于 **wsmserver** 服务的受管机器上的端口号相匹配。
-profile *pathname of preference file*	指定一个备用的首选项文件。缺省首选项文件将会是一个在用户根目录中名为 **WebSM.pref** 的文件。使用这个选项使用户能使用一个不同首选项文件。如果用户管理用于不同客户机的不同机器集，这可能很有用。 **注**: 从本地机器或从 **-host** 自变量指定的机器读取首选项文件。
-user *username*	使“基于 Web 的系统管理器”以给定用户名运行。将提示输入用户密码。
-DdefaultTurners=*value*	当值是 `true` 时，使用“Java 外观感觉”调试器替代 Windows 调试器用于在“导航区域”和“内容区域”中的父树节点。在树对象之间没有绘制有角度的线。
-DdrawTreeLine=*value*	当值是 `true` 和 **-DdefaultTurners**=`true` 时，会导致在导航区域和内容区域中的树对象之间绘制有角度的线。
-Ddatadir=*path*	指定备用目录以寻找通常能在 **/var/websm/config/user_settings** 中找到的配置文件。

示例

1. 要指定一个备用首选项文件，请输入:

   `/usr/websm/bin/wsmaccess -profile` *`pathname of preference file`*

2. 要指定一个备用配置文件，请输入:

   `/usr/websm/bin/wsmaccess -Ddatadir=`*`pathname`*

相关信息

要获取关于安装基于 Web 的系统管理器的信息，请参阅《*AIX 5L V5.2 基于 Web 的系统管理器管理指南*》中的『第 2 章 安装和系统要求』。

---

wsmserver 命令

用途

配置“基于 Web 的系统管理器”服务器功能。

语法

/usr/websm/bin/wsmserver -enable

/usr/websm/bin/wsmserver -disable

/usr/websm/bin/wsmserver -start

/usr/websm/bin/wsmserver -enablehttps [*port_number*]

/usr/websm/bin/wsmserver -disablehttps

/usr/websm/bin/wsmserver -starthttps

/usr/websm/bin/wsmserver -sslalways

/usr/websm/bin/wsmserver -ssloptional

描述

wsmserver 命令用来控制“基于 Web 的系统管理器”使用的服务器进程。服务器用于启用 applet 和“客户机 — 服务器”执行方式。另外，如果已安装安全性功能，则可以配置 SMGate 实用程序。

注：必须指定该命令的全路径名，**/usr/websm/bin/wsmserver**。

标志

-enable 启用 applet 和“客户机 — 服务器”方式。

-disable 禁用 applet 和“客户机 — 服务器”。

-start 启动“基于 Web 的系统管理器”服务器。

通常这仅能由 inetd 使用。

仅当已安装安全性功能时，可以使用下面的标志:

-enablehttps [*port_number*] 启动 SMGate 实用程序。可以指定一个用于 SMGate 服务器的任选的 *port_number*。如果指定 port_number，SMGate 服务器监听该端口，而不是缺省的 9092。

-disablehttps 禁用 SMGate 实用程序。

-starthttps 启动 SMGate 实用程序。通常这是由 **init** 进程启动的。

-sslalways 仅允许安全连接。这个标志用于有安全性配置的系统。

-ssloptional 允许到“基于 Web 的系统管理器”的安全和非安全连接。

示例

1. 要启用用于 applet 和“客户机 — 服务器”方式的“基于 Web 的系统管理器”，请输入:

   ```
   /usr/websm/bin/wsmserver -enable
   ```

2. 要启用 SMGate 实用程序，请输入:

   ```
   /usr/websm/bin/wsmserver -enablehttps
   ```

相关信息

要获取关于安装“基于 Web 的系统管理器”的信息，请参阅*《AIX 5L V5.2 基于 Web 的系统管理器管理指南》*中的『第 2 章 安装和系统要求』。

---

wtmpfix 命令

用途

通过更正日期和时间戳记不一致性来操作连接时计算。

语法

/usr/sbin/acct/wtmpfix [*File* ...]

描述

wtmpfix 命令由 **runacct** 过程调用，用来检查标准输入或包含以 **wtmp** 格式记录的 *File*，并更正那些可能使 **acctcon1** 或 **acctcon2** 命令失败的问题。**wtmpfix** 命令更正日期和时间戳记的不一致性，并把正确的记录写到标准输出。当 **acctcon1** 命令运行时，如果日期和时间戳记不一致，**acctcon1** 命令会生成一个错误并停止运行。

wtmpfix 命令也检查名称字段的有效性以确保它仅由字母数字字符、$（美元符号）或空格构成。如果名称无效，**wtmpfix** 命令将登录名状态更改为 **INVALID** 并将一条诊断消息写入标准错误。这样，**wtmpfix** 命令减少了 **acctcon2** 命令失败的可能性。

每次设置日期时（在系统启动时或使用 **date** 命令设置时），将一对日期更改记录写入 **/var/adm/wtmp** 文件中。第一个记录是旧日期，用 *old time* 字符串表示。*old time* 字符串放置在行字段中并且将 **OLD_TIME** 标志放在类型字段中。第二个记录是新日期，用 *new time* 字符串表示。*new time* 字符串放置在行字段中并且将 **NEW_TIME** 标志放在类型字段中。**wtmpfix** 命令用这些记录使文件中的所有日期和时间戳记同步。

标志

无。

参数

File 指定文件以检查是否包含 **wtmp** 格式的记录。

安全性

访问控制：这些命令应该只将执行（x）访问权授予 **adm** 组的成员。

示例

1. 要将 **wtmp** 格式的二进制记录转换成称为 `dummy.file` 的 ASCII 记录，请输入:

   ```
   /usr/sbin/acct/fwtmp < /var/adm/wtmp > dummy.file
   ```

   二进制 **wtmp** 文件内容被重定向到哑元 ASCII 文件。

2. 要将 ASCII 格式的 `dummy.file` 转换成 **wtmp** 格式的称为 `/var/adm/wtmp` 的二进制文件，请输入 `fwtmp` 命令并使用 `-ic` 开关:

   ```
   /usr/sbin/acct/fwtmp -ic < dummy.file  > /var/adm/wtmp
   ```

   哑元 ASCII 文件被重定向到二进制 **wtmp** 文件。

文件

/usr/sbin/acct/wtmpfix	包含 **wtmpfix** 命令。
/var/adm/wtmp	包含包括一个旧日期和一个新日期的日期更改记录。
/usr/include/utmp.h	包含包括原因、日期和时间的历史记录。

相关信息

acctcon1 或 **acctcon2** 命令、**acctmerg** 命令、**acctwtmp** 命令、**fwtmp** 命令、**runacct** 命令。

*《AIX 5L V5.2 系统管理指南: 操作系统与设备》*中的『设置记帐系统』描述建立记帐系统必须执行的步骤。

参阅 *AIX 5L Version 5.2 System Management Concepts: Operating System and Devices* 中的 Accounting Overview，获取能自动运行的或从键盘输入的记帐命令列表，并获取关于每日和每月报告的准备以及记帐文件的信息。

---

wump 命令

用途

启动“猎取 wumpus”游戏。

语法

wump

描述

wumpus 是一种生活在洞穴中的动物，这些洞穴有许多由隧道互相连接的房间。您在房间之间移动，试图用箭射杀 wumpus 并避免被 wumpus 吃掉或掉进无底的陷井中。还有“超级蝙蝠”，可以将您抓起并将您放在随意选择的房间中。为了在房间之间移动并射出箭，游戏会用适当的问题提示您并按您的指令执行。例如:

```
您正在 14 号房间。
我想要一个草图。
这里有通往 1 13 18 号房间的隧道。
移动还是射击？（m-s）m
哪个房间？1
您正在 1 号房间。
我想要一个草图。
这里有通往 14 17 18 号房间的隧道。
移动还是射击？（m-s）m
```

```
哪个房间? 17
您正在 17 号房间。
您掉到陷井里了!
进行另外一次游戏? (y-n)
```

在上面的示例里，您从房间 14 出发。计算机显示我想要一个草图。这是一个附近有陷井的暗示。选择移动到房间 1。又一次被警告有陷井。然后选择移动到房间 17，在那儿落入了陷井并死了。

在游戏开始，提示指令? (y-n)。选择 y 提供警告、如何移动和如何射击的解释。

游戏结束并提示进行另外一次游戏? (y-n)，如果:

- 杀死了 wumpus。
- wumpus 吃了您。
- 落入了无底陷井。
- 射光了箭。

要在任何时间退出游戏，按中断（Ctrl-C）组合键。

文件

/usr/games 包含系统游戏的位置。

相关信息

arithmetic 命令、**back** 命令、**bj** 命令、**craps** 命令、**fish** 命令、**fortune** 命令、**hangman** 命令、**moo** 命令、**number** 命令、**quiz** 命令、**ttt** 命令、**turnoff** 命令、**turnon** 命令。

X 命令

用途

启动 X 服务器。

语法

X [**-a** *Number*] [**-auth** *FileName*] [**-bc** | **+bc**] [**-bp** *Color*] [**-broadcast**] [**-bs** | **-nobs**] [**-c** *Volume*] [**-cc** *VisualType* [**:***Display*]] [**-class** *DisplayClass*] [**-co** *File*] [**-cookie** *XDMAuthenticationBit*] [**-D** *File*] [**-d** *Depth* [**:***Display*]] **-displayID** *DisplayID*] [**-f** *Number*] [**-fc** *Font*] [**-fn** *Font*] [**-fp** *Font*] [**-help**] [**-I**] [**-indirect** *HostName*] [**-layer** *#* [**:***Display*]] [**-logo** | **nologo**] [**-n :***Number*] [**-once**] [**-P** *RowColumn Display* }] [**-pbuffer** *level* [**:display** *name* | **:display** *number*]] [**-p** *Number*] [**-port** *PortNumber*] [**-query** *HostName*] [**-r** | **r**] [**-s** *Number*] [**-secIP** [*PermissionCode*]] [**-secLocal** [*PermissionCode*]] [**-secSMT** [*PermissionCode*]] [**-stereo** [**:***Display*]] [**-su**] [**-T**] [**-t** *Number*] [**-to** *Number*] [**-v**] [**-wm**] [**-wp** *Color*] [**-wrap** | [**-wrapx**] [**-wrapy**]] [**-x** *ExtensionName*] [**-xkbdir** *Directory*] [**-xkbmap** *FileName*] [[**+**|**-**]**accessx**] [**-ar1** *Milliseconds*] [**-ar2** *Milliseconds*] [**-sp** *FileName*] [**+/- xinerama** *FileName*]

描述

X 命令启动 X 服务器，一个运行在位图终端上的显示服务器。X 服务器将输入和输出请求分配到（或从输入和输出请求分配）程序，该程序位于主机系统或通过网络与其连接的系统上。

通过使用 Ctrl+Alt+Backspace 组合键结束一个增强 X-Windows 会话。

能指定一个或多个显示设备。如果没有指定设备，缺省值是全部。缺省配置顺序是由适配器插槽顺序决定的。在第一个插槽里的适配器被初始化为最左边的屏幕，在第二个插槽里的适配器初始化到下一个屏幕的右边。要重新安排屏幕的布局，请使用 **-P** 标志。**-P** 标志将设备的行和列同设备名关联在一起。能够用 **lsdisp** 命令来决定设备名。

两个屏幕可以排列成垂直的或水平的。以下示例显示用 **-P** 标志指定一个水平排列:

```
-P11 ppr0 -P12 ppr1
```

在第二个 **-P** 标志右边的 2 表明第二个监视器视图是沿 x 轴的。这会产生水平排列:

```
显示器                          显示器
        1                               2
```

要以垂直排列查看两个监视器，**-P** 标志应该是:

```
-P11 ppr0 -P21 ppr1
```

在第一个位置的 2 表明监视器使用沿 y 轴的垂直配置:

```
显示器
        1
显示器
        2
```

在水平配置里，当鼠标正在“显示器 1”中从左向右移动并到达“显示器 1”和 2 的边界时，光标会继续沿同样的 y 轴位置进入“显示器 2”中。当光标到达“显示器 2”的边界且设置了 **-wrapx** 标志时，它出现在“显示器 1”的最左边 y 轴的同样位置。如果没有设置 **-wrapx** 标志，鼠标会停止在“显示器 2”的远离您的一边。

在垂直配置里，当鼠标正在“显示器 1”中从上向下移动并到达“显示器 1”和“显示器 2”的边界时，光标会继续沿同样的 x 轴位置进入“显示器 2”中。当光标到达“显示器 2”的底部且已设置 **-wrapy** 标志时，它会出现在“显示器 1”顶部的 x 轴同样位置。如果没有设置 **-wrapy** 标志，鼠标会停止在“显示器 2”的底部。

标志

-a *Number*
指定鼠标移动的加速倍数。例如，值为 5 会使光标的移动速度是鼠标移动速度的 5 倍。缺省值是 4 个像素；任何指定值必须是大于 0 的正值。

-auth *FileName*
为 X 指定文件，从该文件中读取 MIT（麻省理工学院）魔力 cookie。

-bc
关闭与增强 X-Windows 版本 1.1 的向后兼容性。

+bc
打开与增强 X-Windows 版本 1.1 的向后兼容性。这是缺省值。

-bp *Color*
为该显示指定一个黑像素颜色。缺省值是与屏显示关的。

-bs
在所有屏幕上启用后备存储器支持。后备存储器支持在缺省情况下是禁用的。

-c *Volume*
指定键单击音量。

-cc *VisualType* [**:***Display*]

指定用于显示名指定的屏幕的根窗口的视觉类型。不是所有视觉类型在一切深度上都可用于所有的适配器。**:***Display* 参数是可选的，但当使用 multihead 选项时会很有用。**:***Display* 参数是与在 **lsdisp** 命令中已说明的同样的显示名。如果没有提供显示数或显示名，为所有屏幕选择指定的视觉。

要为缺省视觉指定视觉类型和深度，请分别使用 **-cc** 和 **-d** 标志。

VisualType 参数的值指定为一个字符串或数字，如下所示:

```
String  Numeric equivalent
StaticGray  0
GrayScale  1
StaticColor  2
PseudoColor  3
TrueColor  4
DirectColor  5
```

-co *File*

设置红、绿、蓝（RGB）颜色数据库名。这是颜色数据库的缺省标志。

-D *File*

指定颜色定义数据库文件的全路径名。缺省值是 **/usr/lib/X11/rgb**。

-d *Depth*[**:***Display*]

为显示名指定的屏幕指定根深度。不是所有视觉类型在一切深度上都可用于所有的适配器。

:*Display* 参数是可选的，但是当使用 multihead 选项时会很有用，且必须和 **-P** 标志传递的值相对应。**:***Display* 参数是与在 **lsdisp** 命令中已说明的同样的显示名。在缺少 **:***Display* 参数时，选择指定的深度用于所有 multihead 选项中选择的显示，正如在 **-P** 标志中指定的一样。

-f *Number*

指定蜂鸣声音量。缺省值是 -1 或中等音量。所支持的值如下:

值	设置
0	关闭
1-33	低
-1 或 **34-66**	中
67-100	高

-fc *Font*

为光标符号和光标掩码指定光标字体。缺省值取决于操作系统和显示器。

-fn *Font*

指定缺省文本字体。缺省值取决于操作系统和显示器。

-fp *Font*

指定字体路径。

-I

导致忽略所有保留命令行变量。（大写 i）

-help

打印一条使用情况消息。

-layer *#*[**:***Display*]

指定缺省的视觉应该在 *#* 图层。**:***Display* 参数是与在 **lsdisp** 命令中已说明的同样的显示名。为没有覆盖层，或少于 8 位覆盖层的适配器指定这个标志是无效的。指定这个标志为高于支持图层数的 *#* 会导致缺省视觉驻留在屏幕的缺省图层（和没有使用 **-layer** 标志一样）。

-logo

打开在屏幕保护程序中显示的 X Window System 徽标。当前还没有方法从客户机来更改它。

-n :*Number*

指定连接数。*Number* 参数的有效值是 0 到 255。缺省值是下一个可用的数。程序使用 *Number* 参数与一个特定 X 服务器通信。例如，命令:

```
X -n :18
```

指定与激活的 X 服务器的通信是通过 unix:18 或通过 *Hostname*:18 实现。

-nobs

在所有屏幕上禁用后备存储器支持。这是缺省值。

nologo

关闭显示在屏幕保护程序中的 X Window System 徽标。当前还没有方法从客户机更改它。

-once

在第一个会话结束之后指示服务器退出。通常，服务器自动地启动会话。

-P*RowColumn Display*

在 multihead 配置中指定显示的物理位置。*Row* 参数指示显示所在的行。*Column* 参数指示显示所在的列。

Display 参数是与 **lsdisp** 命令的第一列输出所显示的一样的显示器设备名。出现在命令行上的第一个 **-P***RowColumn Display* 描述 X 服务器的屏幕 0、第二个描述屏幕 1，等等。

-P 标志用于支持多个显示器。

-pbuffer *level* [**:display** *name* | **:display** *number*]

为 **:display** 指定的屏幕指定 **pbuffer** 内存分配级别。这个标志仅在与 GLX 扩展同时使用才有用。

level 参数表明为 pbuffer 保留的帧缓冲区内存的相对数量。指定的值必须在 [0..2] 范围内。 值为 0 表明不应为 pbuffer 保留内存。值为 1 表明应保留很少的内存。值为 2 表明应保留大量内存。不是所有适配器都支持 pbuffer。对那些支持的，也不是所有的屏幕配置都支持 pbuffer。为 pbuffer 保留的帧缓冲区内存的实际数量是设备相关的，并可能受其它因素影响，例如屏幕分辨率或缺省像素深度。

:display 参数是可选的，但是当使用 multihead 选项时会很有用。 **:display** 参数是与在 **lsdisp** 命令中已说明的同样的显示名。如果没有提供显示器 *number* 或 *name*，为所有屏幕选择指定的 **pbuffer** 宽度。

-p *Number*

指定在 X Window System 徽标位置更改的时间间隔（分钟）。这个标志与 **-s**（屏幕保护程序超时）标志一起使用，控制清屏。

-r

禁用自动重复。缺省情况下自动重复是启用的。

r

打开自动重复。

-s *Number*

指定在清屏之前等待的分钟数。缺省值是 10 分钟。如果这个值设置为 0，屏幕保护程序会被禁用。

-secIP [*PermissionCode*]

在因特网套接字上设置本地访问控制。*PermissionCode* 是 3 个八进制数字，该数字能够设置读、写和执行位。如果在安全性标志之后没有指定 *PermissionCode*，那么该套接字的许可权将设置为缺省的 0。

-secLocal [*PermissionCode*]

在 unix 套接字上设置访问控制。*PermissionCode* 是 3 个八进制数字，该数字能够设置读、写和执行位。如果在安全性标志之后没有指定 *PermissionCode*，那么该套接字的许可权将设置为缺省的 0。

-secSMT [*PermissionCode*]
在共享内存传输套接字上设置访问控制。*PermissionCode* 是 3 个八进制数字，该数字能够设置读、写和执行位。如果在安全性标志之后没有指定 *PermissionCode*，那么该套接字的许可权将设置为缺省的 0。

-stereo [**:***Display*]
为 *Display* 指定的屏幕配置图形适配器，以得到最优的立体声支持。

支持的屏幕将配置适配器以提供可获得的对立体声的最佳支持。这可能会减小其它资源，例如结构内存。受影响的实际内存数是设备相关的，并可能受其它因素影响，例如屏幕分辨率或缺省像素深度。

Display 参数是可选的，但是当使用 multihead 选项时会很有用。*Display* 参数是与在 **lsdisp** 命令中已说明的同样的显示名。如果没有提供显示数或显示名，**-stereo** 标志适用于所有支持的屏幕。

不支持的屏幕将会忽略 **-stereo** 标志。

-su
在所有屏幕上禁用 save under 支持。

-T
禁用 Ctrl+Alt+Backspace 组合键，缺省情况下它能结束 AIXwindows 会话和所有由它打开的窗口。

-t*Number*
指定鼠标阈值。缺省值是 2 像素。只有在鼠标在一个时间间隔里移动超出鼠标阈值时加速才会生效，且仅适用于阈值之外的数。

-to *Number*
指定在连接检查之间过去的分钟数。缺省值是 60 分钟。指定值必须大于 0。

-v
指定在 **-s** 标志指定的时间到期之后用当前背景色替换显示。缺省情况下，如果没有使用 **-v** 标志，在 **-s** 标志指定的时间到期之后用背景平铺绘制整个屏幕。

-wm
强制所有窗口的缺省后备存储器都有 **WhenMapped** 值。这是把后备存储器应用于所有窗口一个简便的方法。

-wp *Color*
指定白色像素显示颜色。缺省值取决于显示器。

-wrap
指定当鼠标热点到达任意根窗口的左边或右边或顶部或底部时鼠标的行为。如果这个标志已设置且鼠标热点到达最左边的根窗口的左边界，鼠标会自动定位到最右边的根窗口的右边界相同的 y 位置。

相反地，如果这个标志已设置且鼠标热点到达最右边的根窗口的右边界，鼠标会自动定位到最左边的根窗口的左边界相同的 y 位置。如果没设置这个标志，鼠标会停止在任何根窗口的左边界或右边界。

如果这个标志已设置且鼠标热点到达最顶部的根窗口的上边界，鼠标会定位到最底部的根窗口的下边界相同的 x 位置。

相反地，如果这个标志已设置且鼠标热点到达最底部的根窗口的下边界，鼠标会定位到最顶部的根窗口的上边界相同的 x 位置。

-wrap 标志用于支持多显示器。

-wrapx
指定当鼠标热点到达任意根窗口的左边或右边时鼠标的行为。如果这个标志已设置且鼠标热点到达最左边的根窗口的左边界，鼠标会定位到最右边的根窗口的右边界相同的 y 位置。相反地，如果这个标志已设置且鼠标热点到达最右边的根窗口的右边界，鼠标会定位到最左边的根窗口的左边界相同的 y 位置。如果没设置这个标志，鼠标会停止在任何根窗口的左边界或右边界。

-wrapx 标志用于支持多显示器。

-wrapy
指定当鼠标热点到达任意根窗口的上下边界时鼠标的行为。如果这个标志已设置且鼠标热点到达最顶部的根窗口的上边界，鼠标会定位到最底部的根窗口的下边界相同的 x 位置。

相反地，如果这个标志已设置且鼠标热点到达最底部的根窗口的下边界，鼠标会定位到最顶部的根窗口的上边界相同的 x 位置。如果没设置这个标志，鼠标会停止在任何根窗口的上边界或下边界。

-wrapy 标志用于支持多显示器。

-x *ExtensionName*
指定在服务器初始化时应该装入扩展名。这对大的扩展，例如 Display PostScript Level 2（**dps**），特别有用。这个标志可以与多重扩展名称一起多次指定。

-query *HostName*
启用增强 X-Windows 显示管理器控制协议（**XDMCP**）并将一个**查询**数据包发送到指定的主机。

-query 标志用于 **XDMCP**。

-broadcast
启用 **XDMCP** 并将 **BroadcastQuery** 数据包向网络进行广播。选择第一个响应的显示管理器用于会话。

-broadcast 标志用于 **XDMCP**。

-indirect *HostName*
启用 **XDMCP** 并将 **IndirectQuery** 数据包发送到指定的主机。

-indirect 标志用于 **XDMCP**。

-port *PortNumber*
为 **XDMCP** 指定备用端口号。这个标志必须在任何 **-query**、**-broadcast** 或 **-indirect** 标志之前先指定。通常，服务器一个接一个地启动会话。在第一个会话结束之后，这个标志指示服务器退出。

-port 标志用于 **XDMCP**。

-class *DisplayClass*
为附加的显示限定符设置值，该显示限定符由特定显示选项的资源查找中的 **XDMCP** 使用。

-class 标志用于 **XDMCP**。

-cookie *XDMAuthenticationBits*
在测试 XDM-AUTHENTICATION－1 时，指定在服务器和管理器之间要共享的专用密钥。

-cookie 标志用于 **XDMCP**。

-displayID *DisplayID*
允许显示管理器标识每个显示以使它能找到 **-cookie** 标志指定的共享密钥。

-displayID 标志用于 **XDMCP**。

+/- xinerama
启用／禁用全屏或虚拟大屏幕（VLS）。允许用户将 multihead 环境中的所有显示器作为大屏幕对待。

Xkeyboard 标志

-xkbdir *Directory*	指定用于键盘布局文件的基本目录。
-xkbmap *FileName*	指定在启动时要装入的键盘描述。
[**+**|**-**]**accessx**	启用（**+**）或禁用（**-**）AccessX 组合键。
-ar1 *Milliseconds*	设置以毫秒计算的时间长度，在这个时间内在自动重复启动前必须按下一个键。
-ar2 *Milliseconds*	设置在自动重复生成击键之间应该过去的时间长度（毫秒）。

安全性扩展标志

-sp *FileName* 使服务器试图按安全性策略文件用下面描述的格式读取和解释 *FileName*。文件在服务器启动时读取且在每次服务器复位时重读。

安全性策略文件的语法如下。符号表示法: “*”表示 0 或多次出现的先行元素，“+”表示一次或多次出现。要解释 *foo*/*bar*，应忽略 / 之后的文本；它用来区分下一节中 *foo* 的实例。

```
policy file ::= version line other line*
version line ::= string/v '\n'
other line  ::= comment | access rule | site policy | blank line
comment ::= # not newline* '\n'
blank line ::= space '\n'
site policy ::= sitepolicy string/sp '\n'
access rule ::= property property/ar window perms '\n'
property ::= string
window ::= any | root | required property
required property ::= property/rp | property with value
property with value ::= property/rpv = string/rv
perms ::= [ operation | action | space ]*
operation ::= r | w | d
action ::= a | i | e
string ::= dbl quoted string | single quoted string | unqouted string
dbl quoted string ::= space " not dqoute* " space
single quoted string ::= space ' not squote* ' space
unquoted string ::= space not space+ space
space ::= [ ' ' | '\t' ]*
```

Character sets:

```
not newline ::= any character except '\n'
not dqoute  ::= any character except "
not squote  ::= any character except '
not space   ::= any character except those in space
```

与前面描述的语法相关联的语义如下。

version line

文件中的第一行，指定文件格式版本。如果服务器不能识别版本 *string***/v**，它就忽略文件的其余部分。这里所描述的用于文件格式的版本字符串是 `version-1`。

一旦通过 *version line*，与上面语法不匹配的行会被忽略。

comment

忽略这些行。

sitepolicy

忽略当前行。它们打算指定 XC-QUERY-SECURITY-1 权限方法使用的站点策略。

access rule

这些行指定服务器应该如何对不可信客户机的请求作出反应，这些请求会影响名为 *property***/ar** 的 X Window 属性。本节余下部分描述 *access rule* 的解释。

为了将 *access rule* 应用到给定的 *property***/ar** 实例，*property***/ar** 必须在 *window* 指定的窗口集合中的窗口上。如果 *window* 是 **any**，规则应用于任何窗口上的 *property***/ar**。如果 *window* 是 **root**，规则仅应用于 root 窗口上的 *property***/ar**。

如果 *window* 是 *required property*，应用以下规则。如果 *required property* 是 *property***/rp**，在窗口也有 *property***/rp** 时应用规则，而不管它的值。如果 *required property* 是 *property with value*，*property***/rpv** 也必须有 *string***/rv** 指定的值。在这种情况下，属性类型必须是 STRING 和格式 8，且应该包含一个或多个空终止字符串。如果任一字符串都匹配 *string***/rv**，则应用此规则。

字符串匹配的定义是区分大小写字符的简单比较，而且做以下处理：在 *string***/rv** 中出现的字符“*”是一个意为“任何字符串”的通配符。一个 *string***/rv** 能在字符串中的任何位置包含多个通配符。例如，`x*` 与以 **x** 开始的字符串相匹配，`*x` 与以 **x** 结束的字符串相匹配，`*x*` 与包含 **x** 的字符串相匹配，`x*y*` 与以 **x** 开始且后面包含 **y** 的字符串相匹配。

可能有多个 *access rule* 行用于一个已知的 *property***/ar**。规则按它们在文件中出现的顺序来测试。使用适用的第一个规则。

perms 指定不可信客户机可能试图的操作和服务器为响应那些操作产生应采取的操作。

operation

可以是 **r**（读取）、**w**（写）或 **d**（删除）。以下信息显示“X 协议”属性请求如何映射到这些在“X 财团”服务器中实现的操作。

GetProperty

r 或 **r** 和 **d**（如果 delete = True）

ChangeProperty

w

RotateProperties

r 和 **w**

DeleteProperty

d

ListProperties

none，不可信客户机始终能列出所有属性

action 可以是 **a**（允许）、**i**（忽略）或 **e**（错误）。

Allow 执行请求就好像它是由一个可信的客户机发出的。

Ignore 将请求作为空操作。在 GetProperty 中，忽略意味着如果属性存在则返回一个空属性值，而不管它的实际值。

Error 指定不执行请求并返回一个 `BadAtom` 错误，并将 `atom` 设置为属性名。Error 操作对于所有属性是缺省值，包括那些没有在安全性策略文件中列出来的属性。

action 应用于其后的所有操作，直至遇到下一个 *action*>。因此，`irwad` 意味着忽略读和写、允许删除。

GetProperty 和 RotateProperties 可以执行多个操作（**r** 和 **d**，或**r** 和 **w**）。如果将不同的操作应用于操作，大部分繁重的操作将应用于整个请求；不存在部分请求执行。严重性排列次序是：allow < ignore < error。 因此，对一个属性来说如果 *perms* 是 `ired`（忽略读、错误、删除），且一个不可信客户机在那个属性上用 `delete = True` 尝试 GetProperty，会返回一个错误，但属性值没有。类似地，如果 RotateProperties 的任一属性都不允许读和写，会返回一个错误且不更改任何属性值。

安全性策略文件示例如下:

```
version-1

# Allow reading of application resources, but not writing.
property RESOURCE_MANAGER     root      ar iw
property SCREEN_RESOURCES     root      ar iw

# Ignore attempts to use cut buffers.  Giving errors causes apps to crash,
# and allowing access may give away too much information.
property CUT_BUFFER0          root      irw
property CUT_BUFFER1          root      irw
property CUT_BUFFER2          root      irw
property CUT_BUFFER3          root      irw
property CUT_BUFFER4          root      irw
property CUT_BUFFER5          root      irw
property CUT_BUFFER6          root      irw
property CUT_BUFFER7          root      irw

# If you are using Motif, you probably want these.

property _MOTIF_DEFAULT_BINDINGS        rootar iw
property _MOTIF_DRAG_WINDOW   root      ar iw
property _MOTIF_DRAG_TARGETS  any       ar iw
property _MOTIF_DRAG_ATOMS    any       ar iw
property _MOTIF_DRAG_ATOM_PAIRS         any ar iw

# The next two rules let xwininfo -tree work when untrusted.
property WM_NAME              any       ar




# Allow read of WM_CLASS, but only for windows with WM_NAME.
# This might be more restrictive than necessary, but demonstrates
# the required property facility, and is also an attempt to
# say "top level windows only."
property WM_CLASS             WM_NAME   ar

# These next three let xlsclients work untrusted.  Think carefully
# before including these; giving away the client machine name and command
# may be exposing too much.
property WM_STATE             WM_NAME   ar
property WM_CLIENT_MACHINE    WM_NAME   ar
property WM_COMMAND           WM_NAME   ar

# To let untrusted clients use the standard colormaps created by
# xstdcmap, include these lines.
property RGB_DEFAULT_MAP      root      ar
property RGB_BEST_MAP         root      ar
property RGB_RED_MAP          root      ar
property RGB_GREEN_MAP        root      ar
property RGB_BLUE_MAP         root      ar
property RGB_GRAY_MAP         root      ar
```

```
# To let untrusted clients use the color management database created
# by xcmsdb, include these lines.
property XDCCC_LINEAR_RGB_CORRECTION    rootar
property XDCCC_LINEAR_RGB_MATRICES      rootar
property XDCCC_GRAY_SCREENWHITEPOINT    rootar
property XDCCC_GRAY_CORRECTION          rootar


# oddball property names and explicit specification of error conditions
property "property with spaces"         'property with "'aw er ed

# Allow deletion of Woo-Hoo if window also has property OhBoy with value
# ending in "son".  Reads and writes will cause an error.
property Woo-Hoo              OhBoy = "*son"ad
```

相关信息

aixterm 命令、**xclock** 命令、**xhost** 命令、**xinit** 命令、**xlsfonts** 命令、**xwd** 命令、**xwud** 命令。

lsdisp shell 命令。

x_add_fs_fpe 命令

用途

将网络字体服务器添加到字体路径。

语法

x_add_fs_fpe*Host Port Position TypeName*

描述

x_add_fs_fpe 命令将字体路径元素添加到选择的网络类型名的字体路径，以便字体服务器来访问字体。

Host 指定字体服务器驻留的系统名称。
Port 指定字体服务器端口号。这一号码必须在 **/etc/services** 文件中并以十进制指定。
Position 指定在字体路径里插入这些元素的位置。
TypeName 指定网络类型的名称。每个网络类型有一个由一个或多个字体路径元素构成的字体路径。指定字体路径元素将要添加至的网络类型的名称，或选择通过指定 `All` 将它添加到全部网络类型名。如果一个字体路径元素被添加到 `All` 网络类型，它将会被设置在每个字体路径的末尾。

安全性

访问控制：只有 root 用户应拥有该命令的执行（x）访问权。

示例

要将字体服务器添加到用于网络类型 `x_st_mgr.ether` 的字体路径的开始，输入：

```
x_add_fs_fpe winter 7500 1 x_st_mgr.ether
```

在该示例中，主机 winter 上的字体服务器已经添加到用于网络类型 x_st_mgr.ether 的字体路径的开始。字体服务器端口是 7500。

文件

/usr/lpp/x_st_mgr/bin/x_add_fs_fpe	包含 **x_add_fs_fpe** 命令。
/etc/x_st_mgr/ether.cf	包含网络类型 **x_st_mgr.ether** 配置文件（样本）。

相关信息

aixterm 命令、**bootpd** 守护程序、**login** 命令、**x_add_nfs_fpe** 命令、**x_rm_fpe** 命令。

---

x_add_nfs_fpe 命令

用途

将 NFS/TFTP 访问的字体目录添加到字体路径。

语法

x_add_nfs_fpe *Host Directory Method Position TypeName*

描述

x_add_nfs_fpe 命令将字体路径元素添加到选择的网络类型名的字体路径。这个字体目录将用网络文件系统（NFS）或次要文件传输协议 （TFTP）访问。

Host	为字体目录指定要访问的系统名称。
Directory	指定包含字体的目录的完整路径。
Method	指定用于访问字体的 nfs 或 tftp。
Position	指定在字体路径里插入该元素的位置。
TypeName	指定网络类型的名称。每个网络类型有一个由一个或多个字体路径元素构成的字体路径。指定字体路径元素将要添加至的网络类型的名称，或选择通过指定 All 将它添加到全部网络类型名。如果一个字体路径元素被添加到 All 网络类型，它将会被放置在每个字体路径的末尾。

安全性

访问控制：只有 root 用户应拥有该命令的执行（x）访问权。

示例

要将 /usr/lib/X11/fonts/100dpi 中的字体添加到网络类型 x_st_mgr.ether，输入：

```
x_add_nfs_fpe cedar /usr/lib/X11/fonts/100dpi nfs Last \ x_st_mgr.ether
```

在这里字体路径元素 /usr/lib/X11/fonts/100dpi 被添加到网络类型 x_st_mgr.ether 的字体路径的末尾。字体目录在主机 cedar 上，且使用 NFS 访问它。

文件

/usr/lpp/x_st_mgr/bin/x_add_nfs_fpe	包含 **x_add_nfs_fpe** 命令。

/etc/x_st_mgr/ether.cf	包含网络类型 **x_st_mgr.ether** 配置文件（样本）。

相关信息

aixterm 命令、**bootpd** 守护程序、**login** 命令、**x_add_fs_fpe** 命令、**x_rm_fpe** 命令。

---

x_rm_fpe 命令

用途

从字体路径除去字体路径元素。

语法

x_rm_fpe *TypeName Position Method Host Post Directory*

描述

x_rm_fpe 命令从选择的网络类型名称的字体路径中除去字体路径元素。

TypeName	指定从中除去元素的网络类型名称。
Position	指定元素在字体路径中的位置。
Method	指定用于访问字体路径元素的方法。有效选项是: **tcp** 用于网络字体服务器; **default** 用于初始的缺省字体路径元素; **nfs** 用于 NFS; 以及 **tftp** 用于TFTP。
Host	指定在字体路径元素中指定的系统名称。对使用缺省方法的元素来说，指定 **None**。
Port	指定在字体路径元素中指定的服务器端口号。对使用 `nfs` 或 `tftp` 方法的元素来说，指定 **None**。
Directory	指定包含字体的目录的完整路径。对于网络字体服务器元素，指定 **None**。

安全性

访问控制: 只有 root 用户应拥有该命令的执行（x）访问权。

示例

要从网络类型 `x_st_mgr.ether` 的字体路径除去字体元素 `/usr/lib/X11/fonts/100dpi`，输入:

```
x_rm_fpe x_st_mgr.ether 3 nfs waco None /usr/lib/X11/fonts/100dpi
```

该示例中，在使用 NFS 的主机`waco` 上访问的字体路径元素 `/usr/lib/X11/fonts/100dpi` 已经从用于网络类型 `x_st_mgr.ether` 的字体路径的第三个位置除去。因为对 NFS 没有使用端口号，这个参数被设置成 `None`。

文件

/usr/lpp/x_st_mgr/bin/x_rm_fpe	包含 **x_rm_fpe** 命令。
/etc/x_st_mgr/ether.cf	包含网络类型 **x_st_mgr.ether** 配置文件（样本）。

相关信息

aixterm 命令、**bootpd** 守护程序、**login** 命令、**x_add_nfs_fpe** 命令、**x_rm_fpe** 命令。

xargs 命令

用途

构造参数列表并运行命令。

语法

xargs [**-p**] [**-t**] [**-e** [*EOFString*]] [**-E** *EOFString*] [**-i** [*ReplaceString*]] [**-I** *ReplaceString*] [**-l** [*Number*]] [**-L** *Number*] [**-n** *Number* [**-x**]] [**-s** *Size*] [*Command* [*Argument* ...]]

注: 不要在小写标志和参数之间放置空格。

描述

生成的命令行长度是 *Command* 和每个作为字符串对待的 *Argument*（包括每个字符串的空字节终结符）的大小的总和（以字节为单位）。**xargs** 命令限制命令行的长度。当构造的命令行运行时，组合的 *Argument* 和环境列表不能超出 **ARG_MAX** 字节。在这个约束下，如果不指定 **-n** 或 **-s** 标志，缺省命令行长度至少是由 **LINE_MAX** 指定的值。

标志

-e[*EOFString*]
废弃的标志。请使用 **-E** 标志。

将 *EOFString* 参数用作逻辑 EOF 字符串。如果不指定 **-e** 或 **-E** 标志，则假定下划线（_）为逻辑 EOF 字符串。如果不指定 *EOFString* 参数，则禁用逻辑 EOF 字符串能力，且下划线按照字面含义使用。**xargs** 命令读取标准输入直到达到 EOF 或指定的字符串。

-E *EOFString*
指定逻辑 EOF 字符串以替换缺省的下划线（_）。 **xargs** 命令读取标准输入直到达到 EOF 或指定的字符串。

-i[*ReplaceString*]
废弃的标志。请使用 **-I**（大写 i）标志。

如果没有指定 *ReplaceString* 参数，则使用字符串 "{}"。

注: **-I**（大写 i）和 **-i** 标志是互相排斥的；最后指定的标志生效。

-I *ReplaceString*
（大写 i）。插入标准输入的每一行作为 *Command* 参数的自变量，把它插入每个发生 *ReplaceString* 的 *Argument* 中。*ReplaceStrings* 不能在超过 5 个自变量中使用。在每个标准输入行开始的空字符被忽略。每个 *Argument* 能包含一个或多个 *ReplaceStrings*，但不能大于 255 字节。**-I** 标志同样打开 **-x** 标志。

注: **-I**（大写 i）和 **-i** 标志是互相排斥的；最后指定的标志生效。

-l[*Number*]
（小写的 L）。废弃的标志。请使用 **-L** 标志。

如果没有指定 *Number* 参数，使用缺省值 1。**-l** 标志同样打开 **-x** 标志。

注: **-L**、**-l**（小写的 L）和 **-n** 标志是互相排斥的；最后指定的标志生效。

-L *Number*
用从标准输入读取的非空参数行指定的数量运行 *Command* 命令。如果保留少于指定的 *Number*，*Command* 参数的最后调用可以有少数几个参数行。行以第一个换行字符结束，除非该行的最后一个字符是一个空格或制表符。后续的空格表示延续至下一个非空行。

注: **-L**、**-l**（小写的 L）和 **-n** 标志是互相排斥的；最后指定的标志生效。

-n *Number*

运行 *Command* 参数，且使用尽可能多的标准输入自变量，直到 *Number* 参数指定的最大值。如果满足以下条件，则 **xargs** 命令使用更少的自变量:

1. 如果积累的命令行长度超出了由 **-s** *Size* 标志指定的字节。
2. 最后的迭代有少于 *Number*（但是非零）的自变量保留。

注: **-L**、**-l**（小写的 L）和 **-n** 标志是互相排斥的；最后指定的标志生效。

-p

询问是否运行 *Command* 参数。它显示构造的命令行，后跟一个 `?...`（问号和省略号）提示。输入肯定的、特定于语言环境的响应以运行 *Command* 参数。任何其它响应都会引起 **xargs** 命令跳过那个特定的参数调用。每个调用都将询问您。 **-p** 标志同样打开 **-t** 标志。

-s *Size*

设置构造的 *Command* 行的最大总大小。*Size* 参数必须是正整数。如果满足以下条件，则使用更少的自变量:

1. 自变量的总数超出 **-n** 标志指定的自变量数。
2. 总行数超出 **-L** 或 **-l**（小写 L）标志指定的行数。
3. 累积由 *Size* 参数指定的字节数之前达到 EOF。

-t

启用跟踪方式，并在运行之前将构造的 *Command* 行回送到标准错误。

-x

如果有任何 *Command* 行大于 **-s** *Size* 标志指定的字节数，停止运行 **xargs** 命令。如果指定 **-l**（大写 i）或 **-l**（小写 L）标志，则打开 **-x** 标志。如果没有指定 **-i**、**-l**（大写 i）、**-l**（小写 L）、**-L** 或 **-n** 标志，则 *Command* 行的总长度必须在 -s *Size* 标志指定的限制内。

退出状态

该命令返回下列退出值:

0 所有 *Command* 参数的调用都返回退出状态 0。

1-125 不能汇编满足指定需求的命令行，一个或多个 *Command* 参数的调用返回一个非零的退出状态，或发生一些其它的错误。

126 *Command* 已找到但不能被调用。

127 找不到 *Command*。

如果不能汇编满足指定需求的命令行，则不能调用这个命令，命令的调用被一个信号终止，或一个命令调用以退出状态 255 退出。**xargs** 命令将写一条诊断消息并退出而不处理任何保留的输入。

示例

1. 要对名称在文件中列出的文件使用命令，输入:

   ```
   xargs lint -a <cfiles
   ```

   如果 `cfiles` 文件包含下面的文本:

   ```
   main.c readit.c
   gettoken.c
   putobj.c
   ```

   xargs 命令就构造并运行下面的命令:

   ```
   lint -a main.c readit.c gettoken.c putobj.c
   ```

   如果 `cfiles` 文件包含比列出在单一 shell 命令行上的文件名更多的文件名（最多 **LINE_MAX**），**xargs** 命令会用列出的文件名运行 **lint** 命令。然后它使用余下的文件名构造并运行另一个 **lint** 命令。根据在 `cfiles` 文件中列出的名称，命令看起来可能类似于如下所示的内容:

```
lint -a main.c readit.c gettoken.c . . .
lint -a getisx.c getprp.c getpid.c . . .
lint -a fltadd.c fltmult.c fltdiv.c . . .
```

该命令序列同用所有的文件名运行 **lint** 命令一次不完全一样。**lint** 命令检查文件之间的交叉参考。然而，在这个示例中，它不能在 main.c 和 fltadd.c 文件之间，或任意两个在分开的命令行上列出的文件之间进行检查。

由于这个原因，仅当所有的文件名都在一行上列出时，才可能运行命令。要将这个指定到 **xargs**命令，通过输入以下命令使用 **-x** 标志:

```
xargs  -x lint -a <cfiles
```

如果在文件 cfiles 中的所有文件名没有在一个命令行上列出，**xargs** 命令显示一条错误消息。

2. 要构造包含一定数量文件名的命令，输入:

```
xargs  -t  -n 2 diff <<EOF
starting chap1 concepts chap2 writing
chap3
EOF
```

该命令序列构造并运行每个包含两个文件名的 **diff** 命令（**-n 2**）:

```
diff starting chap1
diff concepts chap2
diff writing chap3
```

-t 标志使在运行 **xargs** 命令前显示每个命令，这样您可以看到正在发生的事件。<<EOF 和 EOF 模式匹配字符定义一个 *here document*，它把在结尾行之前输入的文本用作对 **xargs** 命令的标准输入。

3. 要把文件名插入命令行的中间，输入:

```
ls | xargs  -t  -I {} mv {} {}.old
```

该命令序列通过在每个名字结尾添加 .old 来重命名在当前目录里的所有文件。**-I** 标志告诉 **xargs** 命令插入有 { } （花括号）出现的 **ls** 目录列表的每一行。如果当前目录包含文件 chap1、chap2 和 chap3，这会构造下面的命令:

```
mv chap1 chap1.old
mv chap2 chap2.old
mv chap3 chap3.old
```

4. 要运行个别选择的文件上的命令，输入:

```
ls | xargs  -p  -n 1 ar r lib.a
```

该命令序列允许选择文件以添加到 lib.a 库。**-p** 标志告诉 **xargs** 命令去显示每个它构造的 **ar**命令，并询问是否想运行它。输入 y 来运行命令。如果不想运行这个命令按任意其它键。

会显示一些类似于下面的内容:

```
ar r lib.a chap1 ?...
ar r lib.a chap2 ?...
ar r lib.a chap3 ?...
```

5. 要构造包含特定数量自变量的命令并将那些自变量插入一个命令行的中间，输入:

```
ls | xargs -n6 | xargs -I{} echo {} - some files in the directory
```

如果当前目录包含从 chap1 到 chap10 的文件，构造的输出将会是下列内容:

```
chap1 chap2 chap3 chap4 chap5 chap6 - some files in the directory
chap7 chap8 chap9 chap10 - some file in the directory
```

文件

/usr/bin/xargs 包含 **xargs** 命令。

相关信息

ar 命令、**diff** 命令、**echo** 命令、**ksh** 命令、**lint** 命令、**ls** 命令、**mv** 命令。

*《AIX 5L V5.2 系统用户指南: 操作系统与设备》*中的『shell 概述』和『命令概述』。

AIX 5L Version 5.2 General Programming Concepts: Writing and Debugging Programs 中的 Input and Output Handling Programmer's Overview。

xauth 命令

用途

编辑和显示用于连接 X 服务器的授权信息。

语法

xauth [**-f** *AuthFile*] [**-v** | **-q**] [**-i**] [**-b**] [*CommandArgument* ...]

描述

xauth 命令通常用于编辑和显示用于连接 X 服务器的授权信息。这个程序从一个机器抽取授权记录并把它们合并进另外一个里（例如，当使用远程登录或给其它用户授予访问权时）。

下面的命令可以通过交互方式输入到 **xauth** 命令行上或输入到脚本中。注意: 本程序不与 X 服务器保持联系。

add *DisplayName ProtocolName Hexkey*
通过使用给定协议和键数据，用于指示的显示的授权条目被添加到授权文件。数据被指定为十六进制数字表示的偶数长度字符串，每一对代表一个八位元。每对的第一个数字给出八位元的最重要的 4 位，该对的第二个数字给出了最不重要的 4 位。例如，一个 32 字符的十六进制键代表 128 位值。仅由单一句点构成的协议名被看作 **MIT-MAGIC-COOKIE-1** 的一个缩写。

extract *FileName DisplayName...*
用于每个指定显示的授权条目写入显示的文件。抽取的条目可使用 **merge** 和 **nmerge** 命令读回。如果文件名仅由单一的破折号构成，条目会写入二进制输出。

generate *DisplayName ProtocolName* [*trusted* | *untrusted*] [*timeout seconds*] [*group group-id*] [*data hexdata*]

这个命令类似于 **add**。主要的不同是，不是要求用户提供键数据，而是它连接到 *displayname* 指定的服务器并使用 SECURITY 扩展来获得键数据以存储到授权文件中。如果不能联系服务器，或如果它不支持 SECURITY 扩展，则命令失败。否则，使用给定协议表明的显示的授权条目被添加到授权文件。仅由单一句点构成的协议名被看作 MIT-MAGIC-COOKIE-1 的一个缩写。

如果使用可信的选项，那么使用该授权进行连接的客户机将会像往常一样对显示执行全负荷运行。如果使用不可信的选项，那么用该授权进行连接的客户机将被认为是不可信的，并被防止偷窃或破坏属于可信客户机的数据。请参阅 SECURITY 扩展规范，以获取不可信客户机上强制的限制的全部详细信息。缺省值是不可信。

超时选项指定该权限将会在多长时间（以秒记）有效。如果该权限保持为未使用（没有客户机同它连接）超过了这一时间周期，服务器将会清除该权限，并且将来试图用它来连接也会失败。注意：服务器执行的清除操作不删除来自授权文件的授权条目。缺省超时是 60 秒。

组选项指定使用该权限进行连接的客户机应当属于的应用程序组。请参阅应用程序组扩展规范以获取更多详细信息。缺省值不属于一个应用程序组。

数据选项指定服务器应当用于生成该权限的数据。注意：这同写入授权文件的数据不同。此数据的解释依赖于授权协议。*hexdata* 的格式同在 **add** 命令里描述的 *hexkey* 的格式相同。缺省值是不发送数据。

list [*DisplayName*...]

用于每个指定显示（或如果都没有被指定，则为所有显示）的授权条目以正文格式打印到标准输出。关键数据总是以 **add** 命令给出的十六进制格式显示。

merge [*FileName*...]

从指定的文件读取授权条目并将其合并到授权数据库，取代任何匹配的现有的条目。如果文件名仅由单一的破折号构成，且如果以前尚未读取二进制输入，则读取二进制输入。

[n]extract *Filename DisplayName*...

用于每个指定显示的授权条目写入显示的文件。条目以一种适合于非二进制传输（例如安全电子邮件）的数字格式写入。抽取的条目可使用 **merge** 和 **nmerge** 命令读回。如果文件名仅由单一的破折号构成，条目会写入标准输出。

[n]list [*DisplayName*...]

用于每个指定显示（或如果都没有被指定，则为所有显示）的授权条目以 **nextract** 命令使用的数字格式打印到标准输出。关键数据总是以 **add** 命令给出的十六进制格式显示。

[n]merge [*FileName*...]

从指定的文件读取授权条目并将其合并到授权数据库，取代任何匹配的现有的条目。使用在 **extract** 命令的描述中给出的数字格式。如果文件名仅由单一的破折号构成，且如果以前尚未读取标准输入，则读取标准输入。

remove *DisplayName*...

匹配指定显示的授权条目会从授权文件中除去。

source *FileName*

指定的文件作为包含 **xauth** 命令的脚本来执行。忽略空白行和以“#”（井字符）开头的行。单一破折号可用于表明标准输入（如果尚未读取它）。

info

无论有没有发生任何更改，从什么位置读取 **xauth** 命令，描述授权文件的信息都打印到标准输出。

exit 如果发生任何改变，授权文件被写出（如果允许），并且程序退出。文件结束符作为隐式退出命令对待。

quit 程序退出，忽略任何改变。这也可以通过按中断字符完成的。

help [*String*] 在标准输出中打印以给定字符串开头的全部命令（或如果没有给定字符串，则为全部命令）的描述。

? 在标准输出中打印有效命令的一个简短列表。

add、[**n**]**extract**、[**n**]**list**、[**n**]**merge** 和 **remove** 命令的显示名称使用同 **DISPLAY** 环境变量和公共 *display* 命令行自变量相同的格式。特定于显示的信息（例如屏幕数）的是不必要的并且被忽略。相同机器连接（例如本地主机套接字、共享内存和网际协议 *HostName LocalHost*）引用为 `HostName/unix:DisplayNumber`，这样不同机器的本地条目能存储在一个授权文件里。

> **注**：有不安全网络的用户应该谨慎使用加密的文件传输机制来在机器之间复制授权条目。类似地，MIT-MAGIC-COOKIE-1 协议在不安全环境中不是很有用。对附加安全性感兴趣的站点可能需要使用诸如 Kerberos 的加密授权机制。当前在协议名里不允许使用空格。可以添加引号机制。

标志

下面的选项和 **xauth** 命令一起使用。它们可单独给定（例如，**-q -i**）也可组合指定（例如，**-qi**）。

-f *AuthFile* 指定要使用的授权文件的名称。缺省情况下，**xauth** 使用 **XAUTHORITY** 环境变量或用户主目录中的 .xauthority 指定的文件。

-v 表明 **xauth** 应该详尽地操作并打印指示不同操作（例如，有多少记录被读取或写出）结果的状态消息。如果 **xauth** 从它的标准输入读取命令且它的标准输出定向到终端，则这是缺省值。

-q 表明 **xauth** 应该安静地运行而且不打印未经请求的状态消息。如果 **xauth** 命令在命令行上给出或如果标准输出没有定向到终端，则这是缺省值。

-i 表明 **xauth** 应该忽略任何权限文件锁定。通常，**xauth** 拒绝读或编辑任何被其它程序（通常是 **xdm** 或另一个 **xauth**）锁定的授权文件。

-b 表明 **xauth** 在处理之前应试图中断任何授权文件锁定。这一选项仅用来清除旧的锁定。

示例

xauth 命令的最常见运用是为当前显示抽取目录，把它复制到其它机器并把它合并到远程机器上的用户授权文件：

```
% xauth extract \- $DISPLAY | rsh otherhost xauth merge \-
```

文件

$HOME/.Xauthority 如果 **XAUTHORITY** 环境变量没有定义，则包含缺省授权文件。

xclock 命令

用途

连续显示当前日期和时间

语法

xclock [*-Xtoolkitoption* ...] [**-analog** | **-digital**] [**-chime**] [**-hd** *Color*] [**-help**] [**-hl** *Color*] [**-padding** *Number*] [**-update** *Seconds*]

描述

xclock 命令从系统时钟获取时间，然后以数字或模拟时钟的形式显示并更新它。选择 **-analog** 或 **-digital** 标志，分别以模拟或数字格式显示时钟。也可选择标志来指定时钟的表示方式，包括声音信号和更新频率、颜色以及边框宽度等。

该命令使用 Athena 时钟窗口，它能理解核心资源名称和类。要指定这些资源，需要了解组成 **xclock** 命令的窗口的层次结构。在下面的示例中，缩进项表明了这个层次结构。首先给出的是小窗口类名，随后是小窗口实例名:

```
XClock xclock
        Clock clock
```

下面的示例演示了为该客户机指定资源的可能的方法:

```
xclock.clock.background
XClock*background
xclock*background
```

注: 将资源指定为 xclock.background 适用于 xclock 的前一个版本，但不适用于此版本。

标志

-Xtoolkitoption 除了已列出的特定标志外，**xclock** 命令还接受所有标准的 X Toolkit 命令行选项标志。请参阅『增强 X-Windows协议、工具箱和扩展功能列表』，获取关于可用选项的详细信息。

-analog 设置模拟显示方式，它是缺省的显示方式。绘制常规的 12 小时制时钟的钟面，带有显示分钟的分针和每隔一个小时的敲钟记号。

-chime 指定每半小时一次，一小时两次发出声音信号。

-digital 设置 24 小时制数字显示方式。以数字形式显示日期和时间。

-hd *Color* 指定模拟方式中以彩色显示时指针的颜色。缺省为黑色。

-help 打印允许的选项的简短摘要。

-hl *Color* （小写的 HL）指定模拟时钟指针边缘的突出显示的颜色。缺省为黑色。

-padding *Number* 指定窗口边框和时钟文本或图像之间填充区的宽度，以像素为单位。缺省值为 8。

-update *Seconds* 指定 **xclock** 命令更新其显示的频率，以秒为单位。如果 **xclock** 窗口由被覆盖更改为显露在外，那么 **xclock** 命令立即重新显示。如果指定的更新频率小于 30 秒，则使秒针处于模拟方式。缺省的更新频率是 60 秒。

.Xdefaults 关键字

使用下面的关键字来设置 **xclock** 命令的缺省值。

analog（Boolean 类） 指定模拟时钟，而不是数字时钟。缺省值为 true。

chime（Boolean 类） 指定每半小时或每一小时是否响铃。

fontSet（FontSet 类） 指定数字时钟的字体集。变宽字体可能不会始终正确显示。

foreground（Foreground 类） 指定在彩色显示钟刻度线的颜色。如果指定 **reverseVideo**，那么缺省为白色，否则缺省为黑色。

hands（Foreground 类） 指定模拟时钟以彩色显示时指针内部的颜色。如果指定了 **reverseVideo**，那么缺省为白色，否则缺省为黑色。

highlight（Foreground 类）　指定用于突出显示时钟指针的颜色。如果指定了 **reverseVideo**，那么缺省为白色，否则缺省为黑色。

height（Height 类）　指定时钟的高度。模拟时钟的缺省值为 164 像素。数字时钟的缺省值为按所选字体显示时容纳时钟所需的高度。

padding（Margin 类）　指定内部填充量，以像素为单位。缺省值为 8。

update（Interval 类）　指定 **xclock** 命令更新其显示的频率，以秒为单位。

width（Width 类）　指定时钟的宽度。模拟时钟的缺省值为 164 像素。数字时钟的缺省值为按所选字体显示时容纳时钟所需的宽度。

环境变量

DISPLAY　获取缺省主机和显示器编号。

XENVIRONMENT　获取存储在 RESOURCE_MANAGER 属性中覆盖全局资源的资源文件的名称。

示例

1. 要指定数字时钟显示，请输入:

   ```
   xclock -digital
   ```

2. 要指定模拟时钟使用红色指针，请输入:

   ```
   xclock -hd red
   ```

文件

/usr/lib/X11/app-defaults/XClock　指定所需的资源。

---

xcmsdb 命令

用途

装入、查询或除去存储在屏幕根窗口上属性中的“屏幕颜色个性化数据”（Screen Color Characterization Data）。

注: 只有 X11R5（AIXwindows V1.2.3）中才支持 **xcmsdb** 命令。

语法

xcmsdb [**-display** *Display*] [[**-query**] [**-remove**] [**-color**]] | [**-format 32** | **16** | **8**] [*FileName*]

描述

xcmsdb 命令用于装入、查询或除去存储在屏幕根窗口上属性中的“屏幕颜色个性化数据”。“屏幕颜色个性化数据”是 **Xlib** 的一个主要部分，它是在设备无关和设备相关的颜色规范之间进行正确转换所必需的。**Xlib** 使用 **XDCCC_LINEAR_RGB_MATRICES** 和 **XDCCC_LINEAR_RGB_CORRECTION** 属性来存储彩色监视器的颜色个性化数据。它使用灰度监视器的 **XDCCC_GRAY_SCREENWWHITEPOINT** 和 **XDCCC_GRAY_CORRECTION** 属性。因为 **Xlib** 允许添加“屏幕颜色个性化函数集”（Screen Color Characterization Function Set），所以被添加的函数集可以将它们的“屏幕颜色个性化数据”存储在其它属性中。该实用程序不使用其它属性；因此，需要使用函数集提供的类似的实用程序，或使用示例 **xprop** 实用程序。

如果没有指定 **-query** 或 **-remove** 标志选项，那么 *FileName* 参数（如果没有给定输入文件，也可以是标准输入）的 ASCII 可读内容就会被适当转换以存储在属性中。

> **注：** 虽然支持 **libX11.a** 中的 Xcms API；但是，客户机端颜色名数据库 **/usr/lib/X11/Xcms.txt** 和设备颜色个性化文件 **/usr/lib/X11/XcmsIBM5081.dcc** 却是作为不支持的样本而提供的。

标志

-display*Display*	指定正在转换到的服务器。
- query	读取或试图读取屏幕的根窗口上的 XDCCC 属性。如果读取成功，那么它会将数据转换为可读性更好的格式，然后将数据发送到标准输出。
-remove	除去或试图除去屏幕的根窗口上的 XDCCC 属性。
-color	设置 **-query** 和 **-remove** 选项，只检测 **XDCCC_LINEAR_RGB_MATRICES** 和 **XDCCC_LINEAR_RGB_CORRECTION** 属性。如果没有设置 **-color** 选项，那么 **-query** 和 **-remove** 选项检测所有属性。
-format 32 | 16 | 8	指定 **XDCCC_LINEAR_RGB_CORRECTION** 属性的属性格式（每条为 32、16 或 8 位）。已编码的浮点值的精度将随着每个条目的位数增加而增加。缺省为每个条目 32 位。

参数

FileName	指定屏幕颜色个性化数据文件的 ASCII 可读内容。

示例

1. 使用下面的示例，通过指示 **xcmsdb** 命令从文件中读取“屏幕颜色个性化数据”，从而将其应用于根窗口：

   ```
   xcmsdb /usr/lib/X11/XcmsIBM5081.dcc
   ```

2. 在根窗口应用了“屏幕颜色个性化数据”之后，使用下面的示例指示 **xcmsdb** 命令读回数据（如果数据存在）：

   ```
   xcmsdb -query
   ```

---

xdm 命令

用途

通过 XDMCP 支持来管理 X 显示器集合

语法

xdm [**-config** *ConfigurationFile*] [**-debug** *DebugLevel*] [**-nodaemon**] [**-error** *ErrorLogFile*] [**-resources** *ResourceFile*] [**-server** *ServerEntry*] [**-udpPort** *PortNumber*] [**-session** *SessionProgram*] [**-xrm** *ResourceSpecification*]

描述

xdm（X Display Manager，X 显示器管理器）命令管理 X 显示器集合，这些显示器可能在本地主机上，也可能在远程服务器上。**xdm** 命令设计是按照 X 终端以及 X 联盟标准 XDMCP（*X Display Manager Control Protocol*，*X 显示器管理器控制协议*）的要求进行的。**xdm** 命令提供类似于字符终端上的 **init**、**getty** 及 **login** 命令所提供的服务：提示输入登录名和密码、认证用户以及运行会话。

会话由特定进程的生命期定义；在传统的基于字符的终端范畴中，会话就是用户的登录 shell。在 **xdm** 上下文中，它是任意的会话管理器。这是因为在视窗环境中，用户的登录 shell 进程并不需要任何类似终端的界面来进行连接。如果不能使用真正的会话管理器，那么窗口管理器或终端仿真器一般作为*会话管理器*使用，这就意味着终止进程就会终止用户会话。

当会话结束时，**xdm** 复位 X 服务器并（可选）重新启动整个进程。

当 **xdm** 命令通过 XDMCP 方式接收 **Indirect** 查询时，它可以运行 **chooser** 进程来为显示器执行 XDMCP **BroadcastQuery**（或对指定主机的 XDMCP 查询），并提供可能支持 XDMCP 显示器管理的主机的菜单。这种功能在本身不提供主机菜单的 X 终端中非常有用。

因为 **xdm** 命令要提供用户可见的第一个界面，所以它被设计为简单易用，容易按照特定站点的需求进行定制。

典型用法

xdm 命令被设计为在多种环境中运行。

首先，请设置 **xdm** 配置文件。创建一个目录（通常为 **/usr/lib/X11/xdm**）以存储所有相关文件。下面是一个合理的配置文件，它可以被命名为 **xdm-config**:

```
DisplayManager.servers:       /usr/lib/X11/xdm/Xservers
DisplayManager.errorLogFile:  /usr/lib/X11/xdm/xdm-errors
DisplayManager*resources:     /usr/lib/X11/xdm/Xresources
DisplayManager*startup:       /usr/lib/X11/xdm/Xstartup
DisplayManager*session:       /usr/lib/X11/xdm/Xsession
DisplayManager.pidFile:       /usr/lib/X11/xdm/xdm-pid
DisplayManager._0.authorize:  true
DisplayManager*authorize:     false
```

该文件包含对其它文件的引用。某些资源用一个 *（星号）注明，表示组件被分开。这些资源可以指定为专门用于某个显示器，方法是用显示器名替换 *（星号），不过通常这都没什么用。请参阅下一页上的参考资料部分，那里有完整的论述。

第一个文件（**/usr/lib/X11/xdm/Xservers**）包含要管理的，而且没有使用 **XDMCP** 的显示器的列表。多数工作站只有一个显示器，标号为 0（零），这样文件看起来就和下面的差不多:

```
:0 Local local /usr/bin/X11/X -force
```

这将使 **/usr/bin/X11/X** 一直运行在这个显示器上，并管理一个连续运行的会话周期。

/usr/lib/X11/xdm/xdm-errors 文件包含来自 **xdm** 的错误消息以及由 **Xsetup**、**Xstartup**、**Xsession** 或 **Xreset** 脚本输出到标准错误的任何信息。如果在启动 **xdm** 命令时遇到了任何问题，请检查 **/usr/lib/X11/xdm/xdm-errors** 文件，检查 **xdm** 命令是否与问题有任何关系。

下一个配置条目 **/usr/lib/X11/xdm/Xresources** 使用 **xrdb** 命令作为资源数据库装入到显示器上。由于认证小窗口在启动前读取该数据库，它通常会包含该小窗口的各项参数。

标志

所有选项（除 **-config** 之外）指定的值都可以在配置文件中作为资源指定。

-config *ConfigurationFile*
命名配置文件，该文件指定控制 **xdm** 命令行为的资源。**/usr/lib/X11/xdm/xdm-config** 文件为缺省值。

-debug *DebugLevel*
指定 **DisplayManager.debugLevel** 资源的数值。非零值使 **xdm** 在终端上打印调试语句，并禁用 **DisplayManager.daemonMode** 资源，强制 **xdm** 同步运行。这些错误消息可能并不清楚。要解释这些消息，请查看 **xdm** 命令的 X11R4 源代码。

-nodaemon
指定 **DisplayManager.daemonMode** 资源的值为 False。这将禁用通常的守护程序行为（在该行为中，**xdm** 命令关闭所有文件描述符，解除它与控制终端的关联，并在它第一次启动时将其自身置于后台）。

-error *ErrorLogFile*
指定 **DisplayManager.errorLogFile** 资源的值。该文件包含来自 **xdm** 的错误消息，以及由会话过程中运行的各种脚本和程序写入标准错误的任何信息。

-resources *ResourceFile*
指定 **DisplayManager*resources** 资源的值。该文件使用 **xrdb** 命令装入，指定认证小窗口的配置参数。

-server *ServerEntry*
指定 **DisplayManager.servers** 资源的值。请参阅服务器规范一节，获取对该资源的描述。

-udpPort *PortNumber*
指定 **DisplayManager.requestPort** 资源的值。它将设置 **xdm** 命令监控 **XDMCP** 请求时使用的端口号。**XDMCP** 使用已注册的众所周知的 UDP 端口 177。除非在调试时，否则请不要更改这项资源。

-session *SessionProgram*
指定 **DisplayManager*session** 资源的值。这表明在用户登录后作为会话运行程序。

-xrm *ResourceSpecification*
允许指定任意一个资源，就像在多数 X 工具箱应用程序中一样。

资源

在很多阶段中，**xdm** 的操作都可以通过使用其配置文件（在 X 资源格式中）来控制。某些资源修改所有显示器上 **xdm** 的行为，而其它资源只修改一个显示器上 xdm 的行为。当操作与特定显示器有关系时，显示器名称就会被插入到“DisplayManager”和最终的资源名段之间的资源名中。例如，**DisplayManager.expo_0.startup** 是定义“expo:0”显示器上的启动 shell 文件的资源名。由于资源管理器使用冒号来分隔资源名称和它的值，并使用点号来分隔资源名称的各部分，所以 **xdm** 在生成资源名称时会用下划线来替换点号和冒号。

DisplayManager.servers
指定一个由服务器项组成的文件名，每项一行（如果值由斜杠开头），或一个单独的服务器项。请参阅服务器规范一节获取详细信息。

DisplayManager.requestPort
表明 **xdm** 命令用来侦听输入的 **XDMCP** 请求的 UDP 端口号。除非需要调试系统，否则请将这个值保留为缺省的 177。

DisplayManager.errorLogFile
将错误消息重定向为到指定的文件，而不是到控制台。该文件亦包含由 **Xsetup**、**Xstartup**、 **Xsession** 和 **Xreset** 文件定向到标准错误的任何输出，所以它也将包含在这些脚本中问题的描述。

DisplayManager.debugLevel
如果该资源的整数值大于 0（零），那么 **xdm** 命令就会输出大量的调试信息。它还会禁用守护程序方式，而守护程序方式将丢弃上面的信息并允许非 root 用户运行 **xdm** 命令，通常这个命令都没什么用。

DisplayManager.daemonMode
xdm 命令试图将自己插入到一个与任何终端都无关的守护程序进程中。这是通过派生父进程并任其退出，然后关闭文件描述符并释放控制终端来实现的。在某些环境中，这不是我们所期望的（特别是在调试时）。可以通过将该资源设置为 False 来禁用这个功能。

DisplayManager.pidFile
创建指定的文件名，从而包含 **xdm** 主进程的进程标识的 ASCII 表示形式。**xdm** 命令还使用该文件上的文件锁定来试图清除在同一台机器上运行的多个守护程序，这会产生不可预测的后果。

DisplayManager.lockPidFile
控制 **xdm** 命令是否使用文件锁定来防止多个显示器管理器同时运行。

DisplayManager.authDir
命名一个目录，**xdm** 命令会在初始化会话时在该目录中存储授权文件。缺省值为 **/usr/lib/X11/xdm**。

DisplayManager.autoRescan
布尔值，它控制会话结束而且文件被更改后 **xdm** 命令是否重新扫描配置、服务器、访问控制和认证密钥文件。缺省值为 True。可以向主进程发送一个 **SIGHUP** 信号，强制 **xdm** 守护程序重新读取这些文件。

DisplayManager.removeDomainname
在为 **XDMCP** 客户机计算显示器名称时，名称解析程序一般会为终端创建一个全限定主机名。因为这有时会令人困惑，当设置了这个变量时，如果主机名的域名部分与本地主机的域名相同，**xdm** 命令就删除主机名的域名部分。缺省值为 True。

DisplayManager.keyFile
XDM-AUTHENTICATION-1 样式 **XDMCP** 认证要求在 **xdm** 守护程序和终端之间共享专用密钥。该资源指定包含这些值的文件。文件中的每个条目都由一个显示器名称和共享密钥组成。缺省情况下，**xdm** 命令不包括对 XDM-AUTHENTICATION-1 的支持，因为它需要数据加密方法（data encryption method，DES），而由于美国出口限制的约束，DES 一般是不允许分发的。

DisplayManager.accessFile
为了防止出现未授权的 **XDMCP** 服务和实现 **XDMCP IndirectQuery** 请求的转发，该文件包含一个主机名的数据库（该数据库允许对本机器进行直接访问），或包含一个主机列表（其内容为查询应该转发到的主机）。该文件的格式在『XDMCP 访问控制』一节中描述。

DisplayManager.exportList
一个用空白分隔的附加环境变量的列表，将传输到 **Xsetup**、**Xstartup**、**Xsession** 和 **Xreset** 程序。

DisplayManager.randomFile
一个将进行校验和的文件，生成授权密钥的种子值。该文件应当是经常更改的文件。缺省值为 **/dev/mem**。

DisplayManager.choiceTimeout
用户从选择器选择了主机后要等待显示器响应的秒数。如果显示器在此期间发送一条 `XDMCP IndirectQuery` 消息，那么请求就会被转发到所选择的主机。否则，就会假定它来自新的会话并再次提供选择器。缺省值为 15。

DisplayManager.DISPLAY.resources
指定 **xrdb** 命令要装入的文件的名称作为显示器的屏幕 0 根窗口上的资源数据库。登录小窗口 **Xsetup** 和 **chooser** 程序使用该文件中设置的资源。该资源数据刚好在认证过程开始前装入，所以它能够控制登录窗口的外观。请参阅认证客户机一节，其中描述了适合放在该文件中的各种资源。对该资源来说没有缺省值，不过通常都用 **/usr/lib/X11/xdm/Xresources** 作为名称。

DisplayManager.DISPLAY.chooser
指定程序运行以提供一个主机菜单，包含重定向到特殊的主机名 **CHOOSER** 的间接查询。**/usr/lib/X11/xdm/chooser** 为缺省值。请参阅『**XDMCP 控制访问**』和『**选择器**』一节。

DisplayManager.DISPLAY.xrdb
指定用于装入资源的程序。缺省情况下，**xdm** 命令使用 **/usr/bin/X11/xrdb**。

DisplayManager.DISPLAY.cpp
指定由 **xrdb** 命令使用的 C 预处理器的名称。

DisplayManager.DISPLAY.setup
指定提供登录窗口前运行（作为 root 用户）的程序。该资源可以用于更改登录窗口周围的屏幕外观，或打开其它窗口（例如，在这里可能希望运行 **xconsole**）。缺省情况下不运行任何程序。在这里使用的文件名一般为 **Xsetup**。请参阅设置程序一节。

DisplayManager.DISPLAY.startup
指定在认证过程成功后运行（作为 root 用户）的程序。缺省情况下不运行任何程序。这里使用的文件名一般为 **Xstartup**。请参阅启动程序一节。

DisplayManager.DISPLAY.session
指定要运行的会话（在不作为 root 用户运行时）。缺省情况下将运行 **/usr/bin/X11/xterm**。名称一般为 **Xsession** 脚本。请参阅会话程序一节。

DisplayManager.DISPLAY.reset
指定在会话结束后运行（作为 root 用户）的程序。缺省情况下不运行任何程序。名称一般为 **Xreset** 脚本。请参阅重新设置程序一节。

DisplayManager.DISPLAY.openDelay
通过指定连续两个请求之间相隔的时间长度（按秒计算），在试图打开非让步服务器时控制 **xdm** 命令的行为。

DisplayManager.DISPLAY.openRepeat
通过指定要发出请求的次数，在试图打开非让步服务器时控制 **xdm** 命令的行为。

DisplayManager.DISPLAY.openTimeout
通过指定在真正请求打开时要等待的秒数（即 **connect(2)** 系统调用占用的最长时间），在试图打开非让步服务器时控制 **xdm** 命令的行为。

DisplayManager.DISPLAY.startAttempts
通过指定在放弃访问服务器之前整个过程要完成的次数，在试图打开非让步服务器时控制 **xdm** 命令的行为。在设置显示器管理器 **openRepeat** 资源指定的尝试次数后，或者如果显示器管理器 **openTimeout** 资源指定的秒数在任何特定的请求中耗尽，**xdm** 命令就会结束并重新启动服务器，试图重新连接。该过程会重复 *startAttempts* 次，直到显示器被声明为非活动或被禁用。尽管这种行为看起来可能具有任意性，但人们对它的开发已经颇具经验，而它对多数系统也非常适用。缺省情况下，*openDelay* 值为 **5**、*openRepeat* 值为 **5**、*openTimeout* 值为 **30**、*startAttempts* 值为 **4**。

DisplayManager.DISPLAY.pingInterval
要了解远程显示器何时消失，可以使用 **xdm** 命令定期地 ping 它们，方法是使用 X 连接和 **XSync** 调用。该资源指定两次尝试 ping 之间的时间（按分钟计）。缺省情况下，它的设置为 5 分钟。如果频繁地使用 X 终端，而这些终端可能会变为与管理主机分离，则可以增加这个值。

注：在终端被意外地禁用之后，AIXwindows会话可能会继续存在。**xdm** 命令不 ping 本地显示器。如果服务器为 NFS 服务挂起而且不响应 ping 操作，那么可以结束工作站会话。

DisplayManager.DISPLAY.pingTimeout
要了解远程显示器何时消失，可以使用 **xdm** 命令定期地 ping 它们，方法是使用 X 连接和 **XSync** 调用。该资源指定等待终端对请求作出响应的最长时间（按分钟计）。如果终端不响应，那么会话就会被声明为不活动的并结束。缺省情况下，它的设置为 5 分钟。如果频繁地使用 X 终端，而这些终端可能会变为与管理主机分离，则可以增加这个值。

注：在终端被意外地禁用之后，AIXwindows会话可能会继续存在。**xdm** 命令不 ping 本地显示器。如果服务器为 NFS 服务挂起而且不响应 ping 操作，那么可以结束工作站会话。

DisplayManager.DISPLAY.terminateServer
指定在会话结束后是否应该取消 X 服务器（而不是重新设置它）。如果服务器可能会随着时间变化持续增长而无约束，该选项可以用于限制服务器运行的时间。缺省值为 False。

DisplayManager.DISPLAY.userPath
xdm 命令将会话的 **PATH** 环境变量设置为该值。它应该为一个由冒号分隔的目录列表；请参阅《*AIX 5L V5.2 命令参考大全*》 **sh** 命令，了解完整的描述。**:/bin:/usr/bin:/usr/bin/X11:/usr/ucb** 是常用的设置。可以在构建时在AIXwindows系统配置文件中用 **DefaultUserPath** 资源指定缺省值。

DisplayManager.DISPLAY.systemPath
xdm 命令会将启动及重新设置脚本的 **PATH** 环境变量设置为该资源的值。可以在构建时用系统配置文件中的 **DefaultSystemPath** 资源条目来指定该资源的缺省值；通常可以选择使用 **/etc:/bin:/usr/bin:/usr/bin/X11:/usr/ucb**。注意：该条目中省略了 .（句号）（当前目录）。这对于 root 用户来说是一个很好的做法；它可以避免很多常见的“特洛伊木马（Trojan Horse）”系统侵入方式。

DisplayManager.DISPLAY.systemShell
xdm 命令会将启动及重新设置脚本的 **SHELL** 环境变量设置为该资源的值。缺省值为 **/bin/sh**。

DisplayManager.DISPLAY.failsafeClient
如果缺省会话没有能够运行，**xdm** 命令将返回到该程序。该程序的运行不需要参数，它使用会话可能使用的相同环境变量（请参阅『会话程序』一节）。缺省情况下将使用 **/usr/bin/X11/xterm**。

DisplayManager.DISPLAY.grabServer

DisplayManager.DISPLAY.grabTimeout

为了加强安全性，**xdm** 命令在读取登录名和密码时将抓取服务器和键盘。**grabServer** 资源将指定服务器在读取名称／密码的持续时间内是否应该被控制。如果设置为 False，那么在成功地获得键盘访问权后就会放弃服务器访问权，否则服务器访问权直到会话开始前才会被放弃。缺省值为 False。**grabTimeout** 资源指定 **xdm** 命令在访问权成功获取前要等待的最大时间。如果某个其它的客户机获取了服务器的访问权，抓取就会失败，这也可能归咎于网络延迟太严重。该资源的缺省值为 3 秒；提高该值时要非常小心，因为用户在显示器看到一个类似的窗口时可能会非常迷惑。如果获取访问权失败，**xdm** 命令就会变为不活动的，并重新启动服务器（如果可能）和会话。

DisplayManager.DISPLAY.authorize

DisplayManager.DISPLAY.authName

authorize 是一个布尔资源，它控制 **xdm** 命令是否为本地服务器连接生成并使用授权。如果使用了授权，那么 **xdm** 命令就使用由空白分隔的列表表明的授权机制作为 **authName** 资源的值。**XDMCP** 连接动态地指定支持哪种授权机制，所以在这种情况下 **authName** 资源会被忽略。如果 **authorize** 资源被设置给一个显示器，而授权不可用，那么用户就会看到 Login 小窗口中显示器另一条消息。缺省情况下，**authorize** 资源为 True；**authName** 为 **MIT-MAGIC-COOKIE-1**。

DisplayManager.DISPLAY.authFile

表明文件是用于将来自 **xdm** 命令的授权数据传输到服务器（使用 **-auth** 服务器命令行选项）。应该将其保存在一个目录中并限制写访问权（因为它可以被轻易地删除），结果就会禁用服务器中的授权机制。

DisplayManager.DISPLAY.authComplain

如果该值设置为 False，就会禁用登录窗口中的 **unsecureGreeting**。请参阅认证客户机一节。缺省值为 True。

DisplayManager.DISPLAY.resetSignal

xdm 命令为使服务器复位而发送的信号数。请参阅控制服务器一节。缺省值为 **1(SIGHUP)**。

DisplayManager.DISPLAY.termSignal

xdm 命令为结束服务器而发送的信号数。请参阅控制服务器一节。缺省值为 **15(SIGTERM)**。

DisplayManager.DISPLAY.resetForAuth

使 **xdm** 命令在设置了授权文件后发送 SIGHUP 到服务器，这样会在读取新的授权信息期间发生另一次服务器复位。缺省值为 False，这对所有 AIXwindows 服务器都适用。

DisplayManager.DISPLAY.userAuthDir

当 **xdm** 命令不能对通常的用户授权文件（**$HOME/.Xauthority**）进行写操作时，它会在该目录中创建一个唯一的文件名并将 **XAUTHORITY** 环境变量设置为所创建的文件的名称。它的缺省值为 **/tmp**。

XDMCP 访问控制

由 **DisplayManager.accessFile** 资源指定的数据库文件提供 **xdm** 命令用来控制来自请求 **XDMCP** 服务的显示器的访问。该文件包含三种类型的条目：

- 控制对 **Direct** 以及 **Broadcast** 查询的响应的条目。
- 控制对**Indirect** 查询的响应的条目。
- 宏定义。

Direct 查询条目包含一个主机名或模式（它通过包含一个或多个模式匹配字符，从而与主机名区分开来）。*（星号）匹配任何 0（零）的序列或多个字符，而 ?（问号）匹配任何单个字符。这些字符与显示器设备的主机名进行对比。如果条目为主机名，那么所有比较都是使用网络地址进行的，以便可以使用任何转换为正确的网络地址的名称。至于模式，在比较中只使用真正的主机名，这是为了确保不会去匹配别名。如果在主机名或模式前面加上了 !（感叹号）会导致与条目匹配的主机被排除在外。

Indirect 条目也包含主机名或模式，不过其后跟有一个主机名或宏的列表，**indirect** 查询应当发送到这个列表。

宏定义包含一个宏名称、主机名列表以及该宏扩展到的其它宏。为了区别宏和主机名，宏名称前以 %（百分号）字符开头。宏可以嵌套。

Indirect 条目也可以指定为让 **xdm** 命令运行 **chooser** 命令来提供要连接的主机的菜单。请参阅下一页的选择器一节。

在检查特定显示器主机的访问时，每个条目都要被依次扫描，第一个匹配的条目确定响应。例如，**Direct** 查询条目在扫描寻找 **Indirect** 条目时会被忽略。**Broadcast** 查询条目在扫描寻找 **Direct** 条目时会被忽略。

空行将被忽略。**#** 字符作为注释定界符，该行的剩余部分将会被忽略，行末的 \ (反斜杠) 将使其后的换行符被忽略，这使得间接主机列表可以跨越多行。

下面是一个示例 **Xaccess** 文件:

```
#
# Xaccess - XDMCP access control file
#

#
# Direct/Broadcast query entries
#

!xtra.lcs.mit.edu       # disallow direct/broadcast service for xtra
bambi.ogi.edu     # allow access from this particular display
*.lcs.mit.edu     # allow access from any display in LCS

#
# Indirect query entries
#

%HOSTS          expo.lcs.mit.edu xenon.lcs.mit.edu \\
               excess.lcs.mit.edu kanga.lcs.mit.edu

extract.lcs.mit.edu    xenon.lcs.mit.edu    #force extract to contact xenon
!xtra.lcs.mit.edu         dummy     #disallow indirect access
*.lcs.mit.edu     %HOSTS           #all others get to choose
```

选择器

对于不提供主机菜单以使用 **Broadcast** 或 **Indirect** 查询的 X 终端来讲，**chooser** 程序可以为它们实现这一点。在 **Xaccess** 文件中，请指定 **CHOOSER** 作为 Indirect 主机列表中第一个条目。**chooser** 程序发送一条 **Query** 请求到列表中的每个剩余的主机名，并提供一个包含所有响应该请求主机的菜单。

该列表可以包含 **BROADCAST** 一词，这样 **chooser** 就可以改为发送 **Broadcast** 查询，也可再次提供所有响应请求的主机的菜单。

下面是一个使用 **chooser** 的示例 **Xaccess** 文件:

```
extract.lcs.mit.edu       CHOOSER   %HOSTS         #offer a menu of these hosts
xtra.lcs.mit.edu          CHOOSER   BROADCAST      #offer a menu of all hosts
```

chooser 使用的程序由 **DisplayManager.DISPLAY.chooser** 资源指定。该程序的资源可以放在由 **DisplayManager.DISPLAY.resources** 资源命名的文件中。

chooser 使用 Motif **SelectionBoxWidget** 实现。请参阅 **XmSelectionBoxWidget Class** 文档，获取对资源和小窗口或配件名称的描述。

服务器规范

DisplayManager.servers 资源提供一份服务器规范，如果值以 /（斜杠）开头，则提供一个包含服务器规范（每行一个）的文件的名称。

每个规范表明一个应该一直受管而且不使用 **XDMCP** 的显示器。每个规范至少由三部分组成:

- 显示器名称
- 显示类
- 显示器类型
- 对于本地服务器来说，用于启动服务器的命令行。

0 号本地显示器的典型条目为:

```
:0 IBM-GT local /usr/bin/X11/X :0
```

显示器类型为:

local	本地显示器: **\fIxdm\fP** 必须运行服务器
foreign	远程显示器: **\fIxdm\fP** 对运行的服务器打开一个 X 连接

显示器名称必须是可以在 **-display** 选项中传输到X 程序的名称。该字符串用于生成特定于显示器的资源名称，所以在匹配名称时要注意（例如，如果其它资源被指定为 “`DisplayManager._0.session`”，则使用 “`:0 local /usr/bin/X11/X :0`”，而不是 “`localhost:0 local /usr/bin/X11/X :0`”。显示器类部分还用于特定于显示器的资源中作为资源的类。如果集合了很多的类似显示器（如一个 X 终端聚集），而且希望为这些显示器组设置资源，这非常有用。在使用 XDMCP 时，需要为显示器指定显示器类，因此，特定 X 终端的手册应该记录设备的显示器类字符串。如果没有，可以在调试方式中运行 **xdm** 命令，然后寻找它为该设备生成的资源字符串，其中将包括类字符串。

设置程序

Xsetup 文件在服务器复位后、但在登录窗口出现之前运行。该文件通常为 shell 脚本。它是作为 root 用户运行的，所以要注意安全性。这是修改 root 用户后台或激活应该在屏幕上与 Login 小窗口一同显示的其它窗口的地方。由于 **xdm** 占用了键盘的访问权，其它窗口将不能接收键盘输入。不过，它们可以使用鼠标进行交互；在这里要注意可能存在的安全性漏洞。如果设置了 **DisplayManager.DISPLAY.grabServer**，**Xsetup** 将根本无法连接到显示器。该程序的资源可以放在 **DisplayManager.DISPLAY.resources** 指定的文件中。

除了由 **DisplayManager.exportList** 指定的任何变量，还可以传输下列环境变量:

DISPLAY	指定相关的显示器名称。
PATH	指定 **DisplayManager.DISPLAY.systemPath** 的值。
SHELL	指定 **DisplayManager.DISPLAY.systemShell** 的值。
XAUTHORITY	指定它可以设置为一个权限文件。

认证客户机

MIT 认证小窗口已经被一个由标准的 Motif 窗口组成的认证客户机取代。下面是一个小窗口名（及其小窗口类）的列表:

```
outframe(xmFrameWidget)
  inframe(xmFrameWidget)
  inframe(xmFrameWidget)
    main(XmFormWidget)
      tframe(xmFrameWidget)
        greeting(xmLabelGadget)
      logoline(xmFormWidget)
```

```
  dpyname(xmLabelWidget)
userline(xmRowColumnWidget)
  userlabel(xmLabelWidget)
  username(xmTextWidget)
  passlabel(xmLabelWidget)
  password(xmTextWidget)
failsafeline(xmFormWidget)
  failsafe(xmToggleButtonWidget)
cancelline(xmFormWidget)
  cancel(xmPushButtonWidget)
message(xmLabelWidget)
```

认证客户机从键盘读取名称／密码对。将该客户机的资源放在 **DisplayManager.DISPLAY.resources** 指定的文件中。这些都具有合理的缺省值，所以没有必要指定它们中的任何一个。请查看 **/usr/lib/X11/xdm/Xresources** 以了解关于认证客户机资源的缺省值以及合适的小窗口类文档的更多信息。认证客户机还支持下列资源:

Xlogin*foreground	指定前景使用的颜色。
Xlogin*background	指定背景使用的颜色。
Xlogin*greeting	指定标识该窗口的字符串。缺省值为 AIXwindows 环境。
Xlogin*greetFont	指定用于显示欢迎信息的字体。
Xlogin*frameColor	指定用于显示欢迎信息的背景颜色。
Xlogin*titleMessage	指定标题中显示的字符串。缺省值为运行认证客户机的机器的主机名。
Xlogin*titleFont	指定用于显示标题的字体。
Xlogin*namePrompt	指定在提示输入用户名时显示的字符串。Xrdb 程序将从资源值删去尾部的空白。请在提示符后用转义的反斜杠添加空格。缺省值为 “login:”。
Xlogin*passwdPrompt	指定提示输入密码时显示的字符串。缺省值为 “password:”。
Xlogin*promptFont	指定用于显示两种提示符的字体。
Xlogin*failPrompt	指定用于错误处理（failsafe）按钮的标签。
Xlogin*failFont	指定用于错误处理按钮的字体。
Xlogin*cancelPrompt	指定用于取消按钮的标签。
Xlogin*cancelFont	指定用于取消按钮的字体。
Xlogin*fail	指定表明认证失败时显示的消息。缺省值为 “Login was incorrect.”。
Xlogin*messageFontlist	指定用于显示失败消息的字体。
Xlogin*failColor	指定用于显示失败消息的颜色。
Xlogin*failTimeout	指定失败消息显示的秒数。缺省值为三十秒。
Xlogin*sessionArgument	指定传输到会话程序的自变量。

Xlogin*XmText.translations

它将指定认证客户机所使用的转换。请参阅 X Toolkit 文档，获取关于转换的完整论述。缺省的转换表为:

```
Ctrl<Key>b: backward-character()\n\
Ctrl<Key>a: beginning-of-line()\n\
Ctrl<Key>e: end-of-line()\n\
Ctrl<Key>f: forward-character()\n\
Ctrl<Key>d: kill-next-character()\n\
Ctrl<Key>k: kill-to-end-of-line()\n\
Ctrl<Key>u: kill-to-start-of-line()\n
```

可以通过按 Xresources 中所定义方式替换 XmText 转换，将 XDM 设置为使用标准的 XDM 转换:

注: 由于 Motif 的缺省虚拟绑定的缘故，请使用 `<Key>osfHelp`，而不是 `<Key>F1`。

```
Xlogin*XmText.translations: #override\n\

<Key>osfHelp:    set-session-argument(failsafe) finish-field()\n\
Ctrl<Key>Return: set-session-argument(failsafe) finish-field()\n\
Ctrl<Key>H:      delete-previous-character() \n\
Ctrl<Key>D:      delete-character() \n\
Ctrl<Key>B:      move-backward-character() \n\
Ctrl<Key>F:      move-forward-character() \n\
Ctrl<Key>A:      move-to-beginning() \n\
Ctrl<Key>E:      move-to-end() \n\
Ctrl<Key>K:      erase-to-end-of-line() \n\
Ctrl<Key>U:      erase-line() \n\
Ctrl<Key>X:      erase-line() \n\
<Key>Return:     finish-field() \n
<Key>BackSpace:  delete-previous-character() \n\
<Key>Delete:     delete-previous-character() \n
```

除了典型的 XmText 操作之外，客户机还支持下列操作，以便与标准的 XDM 转换兼容:

delete-previous-character
删除光标前的字符。

delete-character
删除光标后的字符。

move-backward-character
将光标向后移动。

move-forward-character
将光标向前移动。

move-to-beginning
将光标移动到可编辑的文本的最前面。

move-to-end
将光标移动到可编辑的文本的最后面。

erase-to-end-of-line
删除光标后的所有文本。

erase-line
删除所有文本。

finish-field
如果光标在名称字段，继续转到密码字段；如果光标在密码字段，则检查当前的名称／密码对。如果名称／密码对有效，**xdm** 就启动会话。否则就会显示失败消息，并再次提示用户。

insert-char
插入键入的字符。

set-session-argument
指定启动时传输到会话的单一单词参数。请参阅『会话程序』和『典型用法』节。

启动程序

Xstartup 文件通常为一个 shell 脚本。因为它是作为 root 用户运行的，所以在它运行时要注意安全性。它通常包含一些命令，用于向 **/etc/utmp** 添加条目、从文件服务器安装用户的主目录、显示当天消息或在不允许登录时取消会话。

除了由 **DisplayManager.exportList** 指定的环境变量，还要传输下列变量:

DISPLAY	指定相关的显示名称。
HOME	指定用户初始工作目录。
USER	指定用户名。
PATH	指定 **DisplayManager.DISPLAY.systemPath** 的值。
SHELL	指定 **DisplayManager.DISPLAY.systemShell** 的值。
XAUTHORITY	可以设置为一个权限文件。

没有自变量被传输到脚本中。**xdm** 命令在启动用户会话之前一直等待，直到该脚本退出。如果该脚本的出口值为非零值，那么 **xdm** 命令中断会话并启动另一个认证周期。

会话程序

Xsession 程序建立用户会话的风格。它使用授权用户的许可权来运行。

除了由 **DisplayManager.exportList** 指定的任何变量，还可以传输下列环境变量:

DISPLAY	指定相关的显示名称。
HOME	指定用户初始工作目录。
USER	指定用户名。
PATH	指定 **DisplayManager.DISPLAY.userPath** 的值。
SHELL	指定用户的缺省 shell（从 **getpwnam**）。
XAUTHORITY	可以设置为一个非标准的权限文件。

在多数安装中，**Xsession** 程序应该在用户的主目录（**$HOME**）中查找 **.xsession** 文件，它包含用户可能希望作为会话使用的命令。如果不存在用户指定的会话，**Xsession** 程序还应该实现一个系统缺省会话。请参阅典型用法一节。

可以使用 “set-session-argument” 操作将自变量从认证小窗口传输到本程序。这可以用来选择会话的不同风格。通常，该功能用来在会话失败时让用户从普通的会话中脱离开来。这使得用户在失败时能够修复自己的 **.xsession**，而不需要管理员干涉。典型用法一节演示了本功能。

重新设置程序

Xreset 脚本在用户会话结束后运行。它作为 root 用户运行，应该包含撤销 **Xstartup** 中的命令的作用，方法是从 **/etc/utmp** 中除去条目或从文件服务器卸装目录。传输到 **Xstartup** 的环境变量也会被传输到 **Xreset**。该程序与 **Xstartup** 程序是对称的。

控制服务器

xdm 命令使用 POSIX 信号控制本地服务器。**SIGHUP** 信号应该会将服务器复位、关闭所有的客户机连接并执行其它的清理任务。**SIGTERM** 信号应该会取消服务器。如果这些信号没有执行预期的操作，**DisplayManager.DISPLAY.resetSignal** 和 **DisplayManager.DISPLAY.termSignal** 资源可以指定备用的信号。

要控制不使用 **XDMCP** 的远程终端，**xdm** 命令将搜索显示器上的窗口层次结构，并使用协议请求 **KillClient** 试图清理下一个会话的终端。这可能不会真正使所有客户机处于非活动状态，因为只有那些曾经创建了窗口的客户机才会被注意到。**XDMCP** 提供了一种更有保证的机制；当 **xdm** 命令关闭其初始连接时，会话结束，且终端被要求关闭所有其它连接。

控制 XDM

xdm 命令对两种信号作出响应：**SIGHUP** 和 **SIGTERM**。当发送 **SIGHUP** 之后，**xdm** 重新读取配置文件、访问控制文件以及服务器文件。对于服务器文件，它会查看是否添加或除去了条目。如果添加了新条目，**xdm** 命令就会在相关的显示器上启动一个会话。被除去的条目会立即禁用，这意味着正在进行的任何会话都会被结束而不会得到通知，而且不会启动新的会话。

当发送 **SIGTERM** 之后，**xdm** 命令将停止正在进行的所有会话并退出。这可以用在关闭系统的时候。

xdm 命令试图标记各种由 **ps** 命令《*AIX 5L V5.2 命令参考大全*》使用的子进程，方法是编辑已有的命令行自变量列表。由于 **xdm** 命令不能为这项任务分配额外的空间，所以用一个适当的长命令行（使用全路径名应该就足够了）来启动 **xdm** 命令非常有用。每个供显示使用的进程都会被标记为 **-display**。

其它可能性

可以使用 **xdm** 命令每次运行一个会话，方法是通过在命令行指定服务器来使用 **xinit** 命令选项或其它合适的守护程序：

```
xdm -server ":0 local /usr/bin/X11/X :0 -force"
```

也可以运行一个文件服务器和一组 X 终端。它的配置与前面样本中的一样，只是 **Xservers** 文件应该类似于下面的内容：

```
extol:0 VISUAL-19 foreign
exalt:0 NCD-19 foreign
explode:0 NCR-TOWERVIEW3000 foreign
```

这将使 **xdm** 命令管理所有这三个终端上的会话。请参阅控制 XDM一节，获取如何使用信号来启用或禁用这些终端的描述。

注： **xdm** 命令与其它视窗系统不能很好地共存。要在相同的硬件上使用多视窗系统，请使用 **xinit** 命令。

示例

1. 下面的 **xstartup** 样本脚本在 **/etc/nologin** 存在时防止进行登录。由于没有规定要在这里显示任何消息（没有显示文件的核心 X 客 户机），我们不推荐使用本示例中的设置，因为登录可能会失败而不作任何解释。因此，这不是一个完整的示例，而是一个关于可用的功能的演示。

```
#!/bin/sh
#
# Xstartup
#
# This program is run as root after the user is verified
#
if [ \-f /etc/nologin ]; then
        exit 1
fi
exit 0
```

2. 这个 **Xsession** 脚本识别前面的 **Xresources** 文件中的转换所指定的特殊的 **failsafe**（错误处理）方式，从而从一般的会话中脱离出来。

```
#!/bin/sh
exec > $HOME/.xsession-errors 2>&1
case $# in
1)
        case $1 in failsafe)
               exec aixterm -geometry 80x24-0-0
               ;;
        esac
esac
startup=$HOME/.xsession
resources=$HOME/.Xresources
if [ -f /usr/bin/X11/startx ]; then
       exec /usr/bin/X11/startx -t -wait
elif [ -f $startup]; then
       exec $startup
else
       if [ -f $resources ]; then
               xrdb -load $resources
       fi
       mwm &
       exec aixterm -geometry 80x24+10+10 -ls
fi
```

3. 要让 **xdm** 在系统启动时运行，以 root 用户的身份输入以下命令:

```
/usr/lib/X11/xdm/xdmconf
```

4. 要在重新引导时禁用 **xdm**，以 root 用户身份输入以下命令:

```
/usr/lib/X11/xdm/xdmconf -d
```

5. 在使用 **xdm** 管理显示器时，认证过程保证只有被允许的客户机才能连接到显示器。使用 X11 R4 和 X11 R5 库构建的客户机能够理解该协议。使用 X11 R3 或更早的库构建的客户机不支持这种认证协议，且不允许连接到 Xserver，除非已授予 **xhost** 许可权。可以输入以下命令来连接本地客户机:

```
xhost =localhost
```

或

```
xhost =machine
```

其中 *machine* 是本地客户机的主机名。

文件

/usr/lib/X11/xdm/xdm-config 缺省的配置文件。

/usr/lib/X11/xdm/Xaccess	缺省的访问文件，列出已授权的显示。
/usr/lib/X11/xdm/Xservers	缺省的服务器文件，列出了要管理的非 XDMCP 服务器。
$(HOME)/.Xauthority	用户授权文件，其中 **xdm** 存储客户机要读取的密钥。
/usr/lib/X11/xdm/chooser	缺省的选择器。
/usr/bin/X11/xrdb	缺省的资源数据库装入程序。
/usr/bin/X11/X	缺省的服务器。
/usr/bin/X11/xterm	缺省的会话程序和错误处理客户机。
/usr/lib/X11/xdm/A<host>\-<suffix>	存储授权文件的缺省位置。

相关信息

X 命令、**xinit** 命令和 **startx** 命令。

xfindproxy 命令

用途

定位代理服务。

语法

xfindproxy -manager *ManagerAddress* **-name** *ServiceName* **-server** *ServerAddress* [**-auth**] [**-host** *HostAddress*] [**-options** *Options*]

描述

xfindproxy 是一个用于定位可用代理服务的程序。它使用代理管理协议（Proxy Management Protocol）来与代理管理器通信。代理管理器跟踪所有可用的代理服务，在必要时启动新的代理，并确保在可能时共享代理。

如果 **xfindproxy** 成功地获得了代理地址，那么它就会将其打印到标准输出。代理地址的格式与使用的代理服务有关。例如，对于 LBX 的代理服务，代理地址就会是该代理的 X 显示器地址（即 `blah.x.org:63`）。

如果 **xfindproxy** 成功地获得了代理地址，那么它就会向**标准错误**打印一条错误消息。

标志

-manager	该自变量是必需的，它指定代理管理器的网络地址。地址的格式为标准的 ICE 网络标识（例如，`tcp/blah.x.org:6500`）。
-name	该自变量是必需的，它指定所需的代理服务的名称（例如 `LBX`）。名称不区分大小写。
-server	该自变量也是必需的，它指定目标服务器的地址。该地址的格式与用 **-name** 自变量指定的代理服务有关。例如，对于 LBX 的代理服务，地址为一个 X 显示地址（即 `blah.x.org:0`）。
-auth	该自变量是可选的。如果指定了该自变量，**xfindproxy** 将从标准输入读取两行内容。第一行为授权／认证名称。第二行为十六进制格式（xauth 使用的相同格式）的授权／认证数据。**xfindproxy** 会将该授权数据传输到代理，多数情况下，代理会使用该数据来向目标服务器对自身进行授权／认证。

-host
该自变量是可选的。如果 **xfindproxy** 启动一个新的代理服务，它就会传输所指定的主机。代理会选择限制所有到该主机的连接。如果 **xfindproxy** 定位了一个已经存在的代理，就会传输主机，但代理用来使用该主机的语义不定义。

-options
该自变量是可选的。如果 **xfindproxy** 启动一个新的代理服务，它就会传输所指定的任何选项。这些选项的语义与每个代理服务器有关，在这里不被定义。如果 **xfindproxy** 定位了一个已经存在的代理，就会传输这些选项，但没有定义代理如何使用这些选项的语义。

相关信息

proxymngr 命令。

---

xfs 命令

用途

为 X Window System 显示服务器提供字体。

语法

xfs [**-config** *ConfigurationFile*] [**-ls** *ListenSocket*] [**-port** *Number*]

描述

xfs 是 AIXwindows 字体服务器。它为 AIXwindows 显示服务器提供字体。

xfs 服务响应下列信号:

SIGTERM 使字体服务器完全退出。
SIGUSR1 使服务器重新读取其配置文件。
SIGUSR2 使服务器刷新任何可能有的高速缓存数据。
SIGHUP 使服务器复位，关闭所有活动连接，并重新读取配置文件。

服务器通常由系统管理员运行，并以引导文件（如 **/etc/rc.tcpip**）的方式启动。用户还可以启动专用字体服务器来使用特定的字体集。

配置语言是一个关键字和值对的列表。每个关键字后面都跟一个 =（等号）和期望的值。

以下列表显示了识别出的关键字和有效值的类型和描述:

#
当位于第一列时，是一个注释字符。

catalogue（字符串列表）
字体路径元素名称的排序列表。当前实现只支持一个单一的目录（ “all” ），它包含所有指定的字体。

alternate-servers（字符串列表）
该字体服务器的备用服务器列表。

client-limit（基数）
该字体服务器在拒绝服务之前将支持的客户机数。这对于调节每个单独的字体服务器上的负载十分有用。

clone-self（布尔值）
字体服务器在达到 client-limit（客户机限制）时是否应尝试克隆自身。

default-point-size（基数）
不指定的字体的缺省点大小（十分之一点）。

default-resolutions（分辨率列表）

缺省情况下服务器支持的分辨率。该信息可以用来作为预处理的提示，并用不指定分辨率的比例字体进行替换。

分辨率是一对用逗号分隔的 **x** 和 **y** 分辨率，以每英寸的像素数计算。多个分辨率之间用逗号分隔。

error-file（字符串）
错误文件的文件名。所有警告和错误都在这里记录。

port（基数）
侦听连接的服务器上的 TCP 端口。缺省值为 7100。

use-syslog（布尔值）
syslog 功能（支持的系统上）是否应用于处理错误。

deferglyphs（字符串）
设置延迟的字形获取和高速缓存的方式。值为 `none` 表明延迟的字形被禁用。`all` 意即为所有字体启用延迟的字形，而 `16` 意即只为 16 位字体启用延迟的字形。

可以用下面的几种格式之一来命名接受 TCP 连接的字体服务器:

```
tcp/hostname:port
tcp/hostname:port/cataloguelist
```

hostname 指定字体服务器运行所在的机器的名称（或十进制数字地址）。port 是字体服务器侦听连接所在的十进制 TCP 端口。cataloguelist 指定一个目录名的列表，其中“+”为分隔符。下面是几个示例:

```
tcp/expo.lcs.mit.edu:7100, tcp/18.30.0.212:7101/all
```

可以用下面的几种格式之一来命名接受 DECnet 连接的字体服务器:

```
decnet/nodename::font$objname
decnet/nodename::font$objname/cataloguelist
```

nodename 指定字体服务器运行所在的机器的名称（或十进制数字地址）。objname 是正常的、不区分大小写的 DECnet 对象名称。cataloguelist 指定一个目录名的列表，其中“+”为分隔符。

标志

-config *ConfigurationFile*
指定字体服务器将使用的配置文件。

-ls *ListenSocket*
指定一个已经设置为用作侦听套接字的文件描述符。该选项只用来让字体服务器本身在自动衍生其另一份副本时处理其它的连接。

-port *号*
指定服务器侦听连接的 TCP 端口号。

示例

```
#
# sample font server configuration file
#

# allow a max of 10 clients to connect to this font server
client-limit = 10

# when a font server reaches its limit, start up a new one
clone-self = on

# alternate font servers for clients to use
alternate-servers = hansen:7101,hansen:7102

# where to look for fonts
# the first is a set of Speedo outlines, the second is a set of
# misc bitmaps and the last is a set of 100dpi bitmaps
#
catalogue = /usr/lib/fonts/type1,
   /usr/lib/X11/ncd/fonts/misc,
   /usr/lib/X11/ncd/fonts/100dpi/
```

```
# in 12 points, decipoints
default-point-size = 120

# 100 x 100 and 75 x 75
default-resolutions = 100,100,75,75
```

文件

/usr/lib/X11/fs/config	缺省的配置文件

xget 命令

用途

通过安全的通信信道接收保密邮件。

语法

xget

描述

xget 命令用来通过安全的通信信道接收保密邮件。消息只能被预定的收件人读取。**xget** 命令将要求输入密码，并使您能够读取保密邮件。

xget 命令与 **enroll** 命令及 **xsend** 命令一起使用来发送和接收保密邮件。**enroll** 命令设置用于接收保密邮件的密码。**xsend** 命令发送只能由预定的收件人读取的邮件。

当发出 **xget** 命令时，会被提示输入加密密钥。输入以前用 **enroll** 命令设置的密码。

xget 命令的提示符是一个 ?（问号）。下面的子命令控制消息的处置:

q（退出）	将任何没有删除的邮件写入用户的邮箱并退出。按“文件结束”键（Ctrl-D）也有同样的效果。
n（删除）或 **d**（删除）或 **Enter**	删除当前消息并显示下一条消息。
!*Command*	运行指定的工作站命令。
s[*Filename*]	将消息保存到指定的 *File* 参数中，而不是缺省的邮件文件 **mbox** 中。
w[*Filename*]	将不包括标题的消息保存到指定的 *File* 参数中，而不是缺省的邮件文件 **mbox** 中。
? （帮助）	显示子命令摘要。

示例

1. 要接收保密邮件，输入:

   ```
   xget
   ```

   提示输入由 **enroll** 命令建立的密码。输入密码后，**xget** 命将显示提示符（?）和所有保密邮件的清单。
2. 要显示保密邮件，在 **xget** 提示符（?）下按 Enter 键。

   显示最新的消息后，一个 ?（问号）将显示 **xget** 命令正在等待一个 **xget** 子命令。输入 `help` 或 ?（问号）来列出可用的子命令。

3. 要将消息或文件保存到缺省的邮件文件，输入:

   ```
   xget
   ```

   在 ?（问号）提示符出现后按 Enter 键，直到显示所需的文件。在显示合适的文件后，输入:

   ```
   s
   ```

   在本示例中，文件保存在缺省的邮件文件 **mbox** 中。
4. 要将消息或文件保存到特定文件，输入:

   ```
   xget
   ```

   在 ?（问号）提示符出现后按 Enter 键，直到显示所需的文件。在显示合适的文件后，输入:

   ```
   s mycopy
   ```

   在本示例中，文件保存在名为 `mycopy` 的文件中，而不是在缺省的邮件文件中。
5. 要删除消息，输入:

   ```
   xget
   ```

   在 ?（问号）提示符出现后按 Enter 键，直到显示所需的文件。在显示合适的文件后，输入:

   ```
   d
   ```

   在本示例中，当前文件会被删除。

文件

/var/spool/secretmail/*User***.key**	包含 *User* 的加密密钥。
/var/spool/secretmail/*User***.[0-9]**	包含 *User* 使用的加密邮件消息。
/usr/bin/xget	包含可执行文件。

相关信息

enroll 命令、**mail** 命令、**xsend** 命令。

*《AIX 5L V5.2 系统用户指南: 通信与网络》*中的『邮件概述』。

*《AIX 5L V5.2 系统用户指南: 通信与网络》*中的『发送和接收邮件』。

*《AIX 5L V5.2 系统管理指南: 通信与网络》*中的『系统管理的邮件概述』。

xhost 命令

用途

控制什么人可以访问当前主机上的 增强 X-Windows。

语法

xhost [**+** | **-**] [*Name*]

描述

xhost 命令添加或删除 X 服务器接受连接的机器列表上的主机名。

该命令必须从有显示连接的机器上运行。可以通过使用 **-***Host* 参数，从访问列表中除去一个名称。不要从访问列表中除去当前的名称。如果已经这样做了，请在作出任何更改之前注销系统。

输入无变量的 **xhost** 命令将显示访问 X 服务器的当前主机名，并显示一条消息表明访问是否已启用。

为了安全起见，只能从*控制主机*运行影响访问控制的选项。对于工作站来说，这台机器也就是服务器。对于 X 终端来说，这台机器是登录主机。

要在缺省情况下启用远程名称，可以在 **/etc/X***?***.hosts** 文件中定义名称，其中 *?* 为启用访问的显示器号。

例如，显示器 `jeanne:0` 可以由使用 `jeanne` 的缺省主机名的系统上的 **/etc/X0.hosts** 文件中定义的系统访问。在显示名称和文件名中，0 表明已定义的远程系统允许通过 增强 X-Windows 访问的显示器号。

标志

+*Name* 定义要添加到 X 服务器访问列表的主机名（可以选用加号标记）。

- *Name* 定义要从 X 服务器访问列表中除去的主机名。已有的连接没有被中断，但将拒绝新的连接请求。注意：可以除去当前的机器；然而，不允许进行进一步的连接（包括试图将其添加回来）。再一次启用本地连接的唯一方法就是将服务器复位（因此也会中断所有连接）。

+ 指定访问不受限制。访问控制被关闭。

- 打开访问控制。

完整的 *Name* 有以下 *family*:*name* 语法：

inet 因特网主机

local 只包含一个名称，空字符串

注: 该系列是区分大小写的。名称的格式随系列而不同。

---

xinit 命令

用途

初始化 X Window System。

语法

xinit [[*Client*] *Options*] [- - [*Server*] [*Display*] *Options*]

描述

xinit 命令启动 AIXwindows 服务器和系统的第一个客户机程序，该程序不能从 **/etc/init** 或在使用多视窗系统的环境中直接启动 X 服务器。当第一个客户机退出后，**xinit** 命令将停止 X 服务器并结束。

如果命令行上没有给出任何特定的客户机程序，那么 **xinit** 命令查找一个要运行的文件来启动客户机程序。**xinit** 命令查找 **$XINITRC** 环境变量。如果文件不在该处，它就查找 **$HOME/.xinitrc** 文件。如果还没有找到文件，它遵循以下步骤：

1. **xinit** 命令接下来查看 **/usr/lib/X11/$LANG/xinitrc**。

2. 然后，它查看 **/usr/lpp/X11/defaults/$LANG/xinitrc**。
3. 最后，它查看 **/usr/lpp/X11/defaults/xinitrc**。

如果没有这样的文件存在，那么 **xinit** 缺省情况下将使用下面的命令:

```
aixterm \-geometry +1+1 \-n login \-display :0
```

如果命令行上没有给出任何特定的服务器程序，那么 **xinit** 命令遵循以下步骤:

1. **xinit** 命令查找一个作为 shell 脚本来运行的文件，以启动服务器。**xinit** 命令首先在 **$XSERVERRC** 环境变量中查找文件。
2. 如果文件不在该处，那么它查找 **$HOME/.xserverrc** 文件。
3. 如果还没有找到 **$HOME/.xserverrc** 文件，它会接着查找 **/usr/lpp/X11/defaults/xserverrc** 文件。
4. 最后，如果没有找到任何上述文件，**xinit** 命令在缺省情况下将运行 **X** 命令以启动 X 服务器并使用下面的命令:

   ```
   X :0
   ```

注意: 上述内容假定当前的搜索路径中有一个名为 X 的程序。然而，服务器通常都命名为 X *displaytype*，其中 *displaytype* 是由服务器驱动的图形显示的类型。因此，站点管理员应该制作一个到机器上适当类型的服务器的链接，或创建一个使用适当的服务器运行 **xinit** 命令的 shell 脚本。

> **注**: 如果试图启动 AIXwindows 而没有任何可用的指示器设备（如鼠标或输入板），AIXwindows 不会打开。某些设备可以被插入，但没有被定义，所以不能被系统使用，反过来也一样。

重要的一点是，由 **.xinitrc** 运行的程序如果不立即退出的话，应该在后台运行，这样它们就不会妨碍其它程序启动。然而，启动的上个永久程序（通常是窗口管理器或终端仿真器）应该被留在前台，这样脚本就不会退出（这表明用户已经完成操作，xinit 应该退出）。

在命令行上可以指定备用的客户机和/或服务器。期望的客户机程序及其自变量应该作为 **xinit** 的第一个命令行自变量给出。要指定特定的服务器命令行，在 **xinit** 命令行（在任何客户机和自变量后）添加一个 —（双破折号），后跟期望的服务器命令。

客户机程序名和服务器程序名都必须以 /（斜杠）或 .（句号）开头。否则，它们会被当作要添加到各自的启动行的参数处理。这使添加参数（例如，前景和背景颜色）而不必重新输入整个命令行成为可能。

如果没有给出明确的服务器名称，而第一个 —（双破折号）后的参数是 :（冒号）后跟一个数字，**xinit** 就会使用这个数字（而不是 0）作为显示号。所有保留的参数都会添加到服务器命令行。

以下环境变量由 **xinit** 命令使用:

DISPLAY	该变量设置客户机应该连接的显示器的名称。
XINITRC	该变量指定一个包含启动初始窗口的 shell 命令初始文件。缺省情况下，使用主目录中的 **.xinitrc**。
Options	列出希望指定客户机能够使用的任何选项。
Client	指定使用的客户机。例如 xterm 或 aixterm。指定的客户机必须以 .（点号）或 /（斜杠）开头。
Server	使用任何有效的 xserver。指定的客户机必须以 .（点号）或 /（斜杠）开头。

示例

1. 要启动一个名为 X 的服务器并运行用户的 **xinitrc** 程序（如果存在），否则启动 **aixterm** 命令，输入:

   ```
   xinit
   ```

2. 要在备用显示器上启动特定类型的服务器，输入:

   ```
   xinit -- /usr/bin/X11/X qdss:1
   ```

3. 要启动一个名为 X 的服务器并将给出的自变量添加到 缺省的 **xinitrc** 或 **aixterm** 命令，输入:

```
xinit -geometry =80x65+10+10 -fn 8x13 -j -fg white -bg navy
```

4. 要使用 **/Xsun -l -c** 命令来启动服务器并将 **-e widgets** 自变量添加到缺省的 **xinitrc** 或 **aixterm** 命令，输入:

```
xinit -e widgets -- ./Xsun -l -c
```

5. 要在显示器 1 上用 **-a 2 -t 5** 自变量启动名为 X 的服务器，然后在 **fasthost** 机器上启动一个远程 shell，在这台机器上运行 **cpupig** 命令，指示该机器反过来显示到本地工作站上，输入:

```
xinit /usr/ucb/rsh fasthost cpupig -display ws:1 -- :1 -a 2 -t 5
```

6. 以下 **.xinitrc** 脚本的样本将启动一个时钟和几个终端，并让窗口管理器作为最后一个应用程序运行。假定窗口管理器已经正确配置，然后用户选择 **Exit** 菜单项来结束 AIXwindows 会话。

```
xrdb -load $HOME/.Xresources
xsetroot -solid gray &
xclock -g 50x50-0+0 -bw 0 &
xload -g 50x50-50+0 -bw 0 &
xterm -g 80x24+0+0 &
xterm -g 80x24+0-0 &
mwm
```

7. 希望创建通用启动环境的站点可以只创建缺省的 **.xinitrc** 脚本，它会引用整个站点的启动文件:

```
#!/bin/sh . /usr/local/lib/site.xinitrc
```

8. 另一种方法是用特定的 shell 脚本编写一个启动 **xinit** 命令的脚本。这样的脚本通常都被命名为 x11、xstart 或 startx，它们为初学用户提供了一种简单的界面。

```
#!/bin/sh xinit /usr/local/lib/site.xinitrc -- /usr/bin/X11/X bc
```

文件

.xinitrc	包含缺省的客户机脚本文件。
aixterm	如果 **.xinitrc** 不存在，则它包含客户机运行的命令。
.xserverrc	包含缺省的服务器脚本。
X	如果 **.xserverrc** 不存在，则它包含服务器运行的命令。

相关信息

startx 命令、**X** 命令。

---

xkbcomp 命令

用途

编译 XKB 键盘描述

语法

xkbcomp [**-a**] [**-C**] [**-dflts**] [**-I** *Directory*] [**-l**] [**-m** *Name*] [**-merge**] [**-o** *OutputFile*] [**-opt** *Parts*] [**-R** *Directory*] [**-synch**] [**-w** *Level*] [**-xkb**] [**-xkm**] *Source* [*Destination*]

描述

xkbcomp 命令是一个键映射编译器，它将 XKB 键映射的描述转换为几种输出格式之一。**xkbcomp** 最普遍的用途是创建一个编译过的键映射文件（扩展名为 **.xkm**），它可以由能处理 XKB 的 X 服务器或实用程序直接读取。键映射编译器也可以生成 C 头文件或 XKB 源文件。由 **xkbcomp** 生成的 C 头文件可以包含到需

要内置缺省键映射的 X 服务器或实用程序中。由 **xkbcomp** 生成的 XKB 源文件是完全解析的，并且可以用来验证是否正确合并了通常构成 XKB 键映射的文件，或者用来创建包含键映射的全部描述的单个文件。

Source 可以指定一个 X 显示、**.xkb** 或 **.xkm** 文件；除非明确指定，*destination* 的格式由 source 的格式决定。缺省情况下，编译 **.xkb**（keymap 源）文件将生成一个 **.xkm**（编译后的键映射文件）。缺省情况下，如果源文件为 **.xkm** 文件或 X 显示，**xkbcomp** 将生成一个键映射源文件。

如果 *Destination* 为 X 显示，该显示的键映射就用编译过的键映射更新。

destination 的名称通常由 source 的名称决定，扩展名相应地加以替换。在编译一个来自包含多个映射的文件的单一映射时，**xkbcomp** 通过为使用的映射的名称添加适当的扩展名来构造目标文件名称。

标志

-a	显示所有键盘信息，报告隐式或派生的信息作为注释。只影响 **.xkb** 格式的输出。
-C	生成一个 C 头文件作为输出（扩展名为 **.h**）。
-dflts	计算任何丢失组件的缺省值，如键名称。
-I *Directory*	指定搜索键映射描述包含的文件要使用的顶级目录。
-l	列出在命令行上列出的任何文件中指定 *map* 模式的映射。
-m *Name*	指定要用多个条目从文件编译的映射。
-merge	用来自服务器的映射合并已编译的信息。
-o *Name*	指定生成的输出文件的名称。缺省值为带输出格式相应扩展名的源文件的名称。
-opt *Parts*	指定可选部件的列表。任何可选部件中的编译错误都不会是致命的。*部件*可能由字母 **c**、**g**、**k**、**s**、**t** 的任意组合而构成，分别指定 **c**ompatibility map（兼容性映射）、**g**eometry（几何图形）、**k**eycodes（键控代码）、**s**ymbols（符号）和 **t**ypes（类型）。
-R *Directory*	指定相关路径名的根目录。
-synch	强制同步 X 请求。
-w *Level*	在编译期间控制警告的报告。警告级别 0 将禁用所有警告；警告级别 10 将启用所有警告。
-xkb	生成键盘的源描述作为输出（扩展名为 **.xkb**）。
-xkm	生成一个编译后的键映射文件作为输出（扩展名为 **.xkm**）。

相关信息

X 命令。

---

xkbevd 守护程序

用途

XKB 事件守护程序

语法

xkbevd [**-help**] [**-cfg** *File*] [**-sc** *Command*] [**-sd** *Directory*] [**-display** *Display*] [**-bg**] [**-synch**] [**-v**]

描述

xkbevd 事件守护程序侦听指定的 XKB 事件，如果有事件出现，还会执行所请求的命令。配置文件由一个事件说明 / 操作对的列表和 / 或变量定义组成。

事件说明由一个简短的 XKB 事件名后跟一个字符串或标识符组成，后者充当括号中的限定符；空括号表明没有限定，并会指定应用到不与任何其它说明匹配的事件的命令。限定符的解释取决于事件类型。

- 响铃事件匹配对响铃这个名称的使用。
- 消息事件匹配消息字符串的内容。
- 慢速键事件接受 **press**、**release**、**accept** 或 **reject** 之内的任何命令。

其它的事件都不会被识别出来。

操作由一个可选的关键字后跟一个可选的字符串参数组成。**xkbevd** 可以识别下列操作:

- **none**
- **ignore**
- **echo**
- **printEvent**
- **sound**
- **shell**

如果没有指定该操作，那么字符串就会被作为一个要播放的声音文件的名称，除非它以一个感叹号开头，这样的话它就会被作为 shell 命令读取。

参数字符串中的变量定义被扩展了，来自上述事件中在参数字符串前的字段被传输到操作处理器。变量的常规语法无非有两种:

$*c*

或

$(*str*)

其中，*c* 为单独的字符，*str* 为任意长度的字符串。所有参数都有单独字符和长名称。根据不同的事件，识别出的参数的列表都有所不同。

ignore、**echo**、**printEvent**、**sound** 和 **shell** 操作与您预期名为 **ignore**、**echo**、**printEvent**、**sound** 和 **shell** 所做的是一样的，除了 **sound** 命令仅对 SGI 机器实现并测试过。

当前识别出的变量只有 *soundDirectory* 和 *soundCommand*。

标志

-bg
让 **xkbevd** 派生自身并在后台运行。

-cfg *File*
指定要读取的配置文件。如果没有指定配置文件，那么 **xkbevd** 就会按顺序寻找 **~/.xkb/xkbevd.cf** 和 **$(LIBDIR)/xkb/xkbevd.cf**。

-display *Display*
指定要使用的显示器。如果没有，**xkbevd** 就使用 **$DISPLAY**。

-help
打印使用信息。

-sc *Command*
指定要播放声音的命令。

-sd *Directory*
指定声音文件的顶级目录。

-synch
强制同步所有 X 请求。慢一点。

-v
打印更多的信息，包括调试消息。**-v** 的多个说明会使输出结果更多。

相关信息

X 命令。

xkbprint 命令

用途

打印 XKB 键盘描述。

语法

xkbprint [**-?** | **-help**] [**-color**] [**-dflts**] [**-diffs**] [**-eps**] [**-fit**] [**-full**] [**-grid** *Resolution*] [**-if** *FontName*] [**-label** *Type*] [**-lc** *Locale*] [**-level1**] [**-level2**] [**-lg** *Group*] [**-ll** *Level*] [**-mono**] [**-n** *Number*] [**-nkg** *Number*] [**-npk** *Number*] [**-o** *File*] [**-R** *Directory* [**-pict** *Which*]] *Source* [*OutputFile*]

描述

xkbprint 命令生成一个由 *Source* 指定的 XKB 键盘描述的可打印或封装的 PostScript 描述。*Source* 可以是任何编译过的 keymap、**.xkm** 文件，包括一个几何描述或 X 显示规范。如果指定了一个 *OutputFile*，那么 **xkbprint** 就会向它写数据。否则，**xkbprint** 将创建输出文件，根据请求的格式用 **.ps** 或 **.eps** 代替源文件的扩展。如果源为非本地 X 显示（例如 `:0`），**xkbprint** 就会将适合的前缀附加在显示规范上，用 **-**（横杠）代替冒号。对于本地显示，**xkprint** 将使用 `server-`*n*，其中 *n* 为显示号。

标志

-? | -help
打印使用消息。

-color
使用几何文件中指定的颜色进行打印；缺省情况下，**xkbprint** 打印一个键盘的黑白图像。

-dflts
试图计算任何丢失的组件（如密钥）的缺省名称。

-diffs
仅在符号明确的绑定时显示它们。

-eps
生成一个封装的 PostScript 文件。

-fit
安排页面上的键盘映象，这是缺省值。

-full
全尺寸打印键盘。

-grid *Resolution*
用键盘上的 *Resolution***mm** 分辨率打印网格。

-if *FontName*
如果不指定输出文件，就指定内部 PostScript 类型 1 字体来转储到指定的输出文件或 *fontName***.pfa** 上。如果转储的是内部字体，那么不打印任何键盘描述。

-label *Type*
指定要根据密钥打印的标签。有效地类型为:

- **none**
- **name**
- **code**
- **symbols**

-lc *Locale*
指定 KeySyms 应该解析的语言环境。

-level1
生成级别 level 1 的 PostScript。

-level2	生成级别 2 的 PostScript。
-lg *Group*	从 *Group* 开始，打印键盘组中的符号。
-ll *Level*	从换档级别 *Level* 开始打印符号。
-mono	生成键盘的黑白图像，这是缺省值。
-n *Number*	打印副本的 *Number*。
-nkg *Number*	打印 *Number* 键盘组中的符号。
-npk *Number*	指定每页要打印的键盘图像的 *Number*。对于 EPS 文件，它指定要打印的键盘图像的总数。
-o *File*	将输出写到 *File*。
-R *Directory*	使用 *Directory* 作为跟目录，所有路径名都根据 *Directory* 来解释。
-pict *Which*	控制统计图表（而不是可用的 keysym 名称）的使用。*Which* 的有效值为: • **all** • **none** • **common** (default).
-synch	强制同步 X 请求。
-w *Level*	设置警告级别。 • 0 就是没有警告 • 10 就是全都警告

相关信息

X 命令和 **xkbcomp** 命令。

---

xlock 命令

用途

锁定本地 X 显示器，直到输入密码。

语法

xlock [**-batchcount** *Number*] [**-bg** *Color*] [**-delay** *Users*] [**-display** *Display*] [**-fg** *Color*] [**-font** *FontName*] [**-info** *TextString*] [**-invalid** *TextString*] [**-mode** *ModeName*] [**+mono** | **-mono**] [**-username** *TextString*] [**-nice** *Level*] [**+nolock** | **-nolock**] [**-password** *TextString*] [**+remote** | **-remote**] [**+allowaccess** | **-allowaccess**] [**+allowroot** | **-allowroot**] [**+echokeys** | **-echokeys**] [**+enablesaver** | **-enablesaver**] [**-help**] [**-saturation** *Value*] [**-timeout** *Seconds*] [**+usefirst** | **-usefirst**] [**+v** | **-v**] [**-validate** *TextString*]

描述

xlock 命令锁定 X 服务器，直到用户从键盘输入密码。在 **xlock** 命令运行期间，所有新的服务器连接都会被拒绝。屏幕保护程序将被禁用，鼠标指针将被关闭，屏幕置空，只显示一个不断变化的模式。如果按下了键盘或鼠标按钮，就会启动 **xlock** 命令**提示用户输入密码**。

如果输入了正确的密码，屏幕就会解除锁定，X 服务器将被恢复。在输入密码时，可以使用 Ctrl-U 和 Ctrl-H 分别作为取消和删除。要返回到锁定的屏幕，请单击变化的模式的小图标。

要正常运转，**xlock** 需要在 root 用户的许可下运行，因为操作系统会限制对密码和访问控制文件的访问。要给 **xlock** root 用户许可，您可以进行下列步骤:

1. 登录为 root 用户。
2. 到包含 **xlock** 程序文件的目录。
3. 运行两个命令:
   a. **chown root xlock**
   b. **chmod u+s xlock**

标志

-batchcount *Number*
设置每个批处理中要处理任务的数量。*Number* 指取决于模式的不同任务。

qix 指按同一种颜色处理的行数。

hop 指按同一种颜色处理的像素数。

image 指屏幕上面一次显示的 sun 徽标数。

swarm 指蜜蜂的数量

life and blank
不应用。

-bg *Color*
设置密码屏幕上的背景颜色。

-delay *Number*
设置模式操作的速度，为一组 **hopalong** 像素、**qix** 行、**life** 生成**image** 位 和 **swarm** 动作之间延迟的微秒数。

在 **blank** 模式下，一定要将其设置为比较小的数字，因为键盘和鼠标只在每个延迟后检查动作。一个为零的延迟会在检查紧跟的循环中的鼠标和键盘输入时不必要地消耗处理单元，因为 **blank** 模式不起任何作用。

-display *Display*
设置 X11 显示器为锁定状态。**xlock** 命令将锁定服务器上所有可用的屏幕，并限制您只能锁定本地服务器，如 **unix:0**、**localhost:0** 或 **:0**（除非您设置了 **-remote** 标志）。

-fg *Color*
设置密码屏幕上文本的颜色。

-font *FontName*
设置提示屏幕上使用的字体。

-help
打印关于可用选项的一个简要描述。

-info *TextString*
定义包含信息的消息。缺省值为 `Enter password to unlock; select icon to lock`。

-invalid *TextString*
指定密码消息。缺省值为`Invalid login`。

-mode *ModeName*
指定下面六种显示方式之一:

blank 显示空的屏幕。

hop 显示 *Scientific American* 1986 年 9 月刊中的真实飞机几何图形。

image 显示几个随机出现的 sun 徽标。

life 显示 Conway 的生命游戏。

qix 显示旋转的线条。

swarm 显示跟着一只黄蜂的一群蜜蜂。

-nice *NiceLevel*
设置 **xlock** 进程的系统友好程度。

-password *TextString*
指定密码提示字符串。缺省值为 `Password:`。

-saturation *Value*
设置彩色斜坡的饱和度。如果值为 0（零）则为灰度，如果值为 1 则为丰富的颜色。如果值为 0.4，则为适中的颜色饱和度。

-timeout *Seconds*
设置密码屏幕超时前的秒数。

-username *TextString*
指定在用户名前面显示的消息。缺省值为 `Name:`。

-validate *TextString*
指定在验证密码时显示的消息。缺省值为 `Validating login...`。

-/+allowaccess 允许禁用访问控制表，但仍会使本地服务器提示输入密码。如果用 **-KILL** 命令取消了 **xlock**，就不会失去访问控制表。

在远程运行 **xlock** 命令时也需要这个标志，因为对控制列表的访问也被限制了。

-/+allowroot 允许用 root 用户密码解开服务器以及启动 **xlock** 命令的用户的锁定。

-/+echokeys 使 **xlock** 命令在屏幕上对每个输入到密码提示的字符显示闪烁的“?”（问号）字符。缺省值为不闪烁。

+/-enablesaver 启用缺省的屏幕保护程序。如果将延迟参数设得足够长，可能会使某些显示器上的荧光体烧坏。该标志可以用作附加的预防措施。

+/-mono 使 **xlock** 命令在彩色显示器上显示单色（黑白）像素，而不是缺省的彩色像素。

+/-nolock 使 **xlock** 命令只绘制图案而不锁定显示器。按键或鼠标点击都会终止屏幕保护程序。

+/-remote 允许对 X11 服务器进行远程锁定。该标志应当小心使用。它主要旨在锁定不能本地运行 **xlock** 命令的 X11 终端。如您锁定了工作站而不是自己的机器，别人就需要您的密码来解锁。 **-remote** 选项不会使您失去锁定到另一个 shell 的能力。

+/-usefirst 允许使用激活密码屏幕时敲入的那个键作为密码的第一个输入字符。缺省情况下将忽略第一次击键。

+/-v 负号前缀将启用详细模式，它会决定**xlock** 命令将使用的选项。加号前缀为缺省值。

---

xlsfonts 命令

用途

显示用于 X-Windows 的字体列表。

语法

xlsfonts [**-display** *Host*:*Display*] [-**l** [**l** [**l**]]] [**-m**] [**-C**] [**-1**] [**-w** *Width*] [**-n** *Columns*] [**-u**] [**-o**] [**-fn** *Pattern*]

描述

xlsfonts 命令列出与指定的 *Pattern* 参数匹配的字体列表。使用通配符“*”（星号）来匹配任意字符序列（包括没有字符），用“?”（问号）来匹配任意的单个字符。如果不给定模式，则采用“*”。

注: “*”和“?”字符都必须放入引号中，以防止 shell 扩展它们。

可以使用标志来指定服务器、要打印的列数和宽度、字体清单的大小、是否要为输出排序以及是否使用 **OpenFont** 代替 **ListFonts**。

标志

注: 使用 **xlsfonts** 命令的 **-l**（小写的 L）标志会占用服务器很长时间。这通常都发生在单线程非优先服务器，并不是一个程序错误。

-1 表明清单应当只使用一列。本标志与 **-n** *1* 标志相同。

-C 表明清单应当使用多列。本标志与 **-n** *0* 标志相同。

-display *Host*:*Display* 通过指定主机名和显示器号，标识要访问的 X 服务器。

-fn *Pattern* 指定 **xlsfonts** 将列出的字体名 *Pattern*。

-l [**l** [**l**]] （小写的 L）表明应当分别为每种字体生成中等、长和超长清单。

-m 表明长清单也应该打印每种字体的最小和最大边界。

-n *Columns* 指定用来显示输出的列数。缺省情况下，**xlsfonts** 命令试图使用与 **-w** *Width* 标志指定的字符数一样多的字体列。

-o 指示 **xlsfonts** 命令执行 **OpenFont**（如果合适，还执行 **QueryFont**），而不是 **ListFonts**。**-o** 标志在 **ListFonts** 或 **ListFontsWithInfo** 不能列出已知字体时很有用，这与变宽字体系统上的情况一样。

-u 表明输出应当保留为未排序。

-w *Width* 指定应当用于确定要打印多少列的字符宽度。缺省值为 79。

环境变量

DISPLAY 获取要使用的缺省主机和显示器。

示例

1. 要为每种字体指定中等大小的列表，请使用小写的 L 并输入:

   ```
   xlsfonts -l
   ```

2. 要为每种字体指定三列的列表，请输入:

   ```
   xlsfonts -n 3
   ```

3. 要显示名称中包含字符串 iso8859 的所有字体，请输入:

   ```
   xlsfonts -ll "*"iso8859"*"
   ```

4. 要列出名称中包含 rom1 和以下一个字符的所有字体，请输入:

   ```
   xlsfonts rom1"?"
   ```

   这样获取类似于以下清单:

   ```
   rom10  rom11  rom14  rom16  rom17
   ```

相关信息

X 命令、**xset** 命令。

---

xmbind 命令

用途

配置虚拟键绑定

语法

xmbind [**-display** *Host:Display:ScreenID*] [*FileName*]

描述

xmbind 命令是一种 X Windows System 客户机命令，它为 AIXwindows 应用程序配置虚拟键绑定。这个操作是启动时由 **mwm** 命令执行的，所以只有在 **mwm** 没有被使用或者希望不重新启动 **mwm** 就更改绑定时，才需要 **xmbind** 客户机。如果指定了文件，它的内容被用作虚拟键绑定。如果没有指定文件，那么使用用户主目录中的 **.motifbind** 文件。如果没有找到该文件，**xmbind** 命令就装入缺省的虚拟键绑定。

标志

-display *Host*:*Display*:*ScreenID*

指定使用的显示器。**-display** 选项有如下参数:

Host 指定网络上的有效系统的主机名。根据情况的不同，它可能是用户的主机名或远程系统的主机名。

Display 指定要用于显示输出的系统上的显示器号（通常为 0）。

ScreenID 指定要用于显示输出的屏幕号。对于单屏幕系统来说，该号码为 0。

参数

FileName 指定包含虚拟鼠标和按键事件的绑定的文件。

出口状态

本命令返回以下出口值:

0 指示成功完成。
>0 指示发生错误。

相关信息

X 命令。

---

xmkmf 命令

用途

从 **Imakefile** 创建 **Makefile**。

语法

xmkmf [**-a**] [*TopDir* [*CurDir*]]

描述

xmkmf 命令从随第三方软件一起提供的 **Imakefile** 创建 **Makefile**。在包含 **Imakefile** 文件的目录中不使用自变量或变量进行调用时，**imake** 命令会使用适合系统的自变量（在构建 X 时配置到 **xmkmf** 中）运行并生成 **Makefile**。

标志

-a 首先在当前目录中构建 **Makefile**，然后自动执行 **make Makefiles**、**make includes** 以及 **make depend**。这就是配置 MIT X 构建树之外的软件的方法。

变量

如果在 MIT X 构建树内操作（除非是 X 开发者，否则就大不可能），则指定 *TopDir* 和 *CurDir*。

TopDir 指定从当前目录到构建树顶端的相对路径名。
CurDir 指定从构建树顶端到当前目录的相对路径名。

如果当前目录有子目录，则 *CurDir* 变量是必需的；否则，**Makefile** 不能构建子目录。如果给定 *TopDir* 变量，**xmkmf** 就会假定系统上没有安装任何文件，并在构建树中搜索文件，而不是使用已经安装的版本。

相关信息

imake 命令、**make** 命令。

---

xmodem 命令

用途

用 **xmodem** 协议传输文件，在异步传输期间检测数据传输错误。

语法

xmodem { **-s** | **-r** } *FileName*

描述

xmodem shell 命令与异步终端模拟（Asynchronous Terminal Emulation，ATE）程序一起使用来传输文件，由 *FileName* 参数指定，使用 **xmodem** 协议。

xmodem 协议是一种 8 位传输协议，用于检测数据传输错误并重新传输数据。发送数据的工作站等待远程系统发送一个表明其能够接收数据的信号。

在接收系统得到数据后，它向发送系统返回一条确认消息。在 ATE 程序中，如果在文件传输开始后 90 秒内还没有收到数据，则接收系统超时。

用 **xmodem** 命令发送和接收数据是互补性操作。一个系统设置为接收时，另一个系统必须设置为发送。请将远程系统上的 **xmodem** 命令与本地系统上的 ATE Connected Main Menu 菜单中的 **send** 子命令或 **receive** 子命令结合使用。

要中断 **xmodem** 文件传输，请按 Ctrl-X 组合键。

标志

-r 从本地工作站接收数据。
-s 将数据发送到本地工作站。

示例

用 xmodem 协议发送文件

要使用 **xmodem** 协议发送文件 `myfile`，请使用 **ate** 命令和 **connect** 或 **directory** 子命令来建立到远程系统的连接。

1. 在登录到远程系统之后，按下 MAINMENU_KEY（通常为 Ctrl-V 组合键）以返回到本地系统上的 ATE 之前，请输入:

   ```
   xmodem -r myfile
   ```

   （在 shell 命令行上输入。）**xmodem** 协议在远程系统上启动接收方式。
2. 按 MAINMENU_KEY 返回到本地系统上的 ATE。

   显示 ATE Connected Main Menu。
3. 在 ATE Connected Main Menu 的提示符下输入 **send** 子命令:

   ```
   s myfile
   ```

   send 子命令指示本地系统发送 `myfile` 到远程系统。在传输文件之后，显示 ATE Connected Main Menu。

用 xmodem 协议接收文件

使用 **xmodem** 协议，用 **ate** 命令和 **connect** 或 **directory** 子命令建立到远程系统的连接，从远程系统接收 `infile` 文件。

1. 在登录到远程系统后，按下 MAINMENU_KEY（通常为 Ctrl-V 组合键）以返回到本地系统上的 ATE 之前，请输入:

   ```
   xmodem -s infile
   ```

   （在 shell 命令行上输入）。远程系统上将以发送方式启动 **xmodem** 协议。
2. 按 MAINMENU_KEY 返回到本地系统上的 ATE。

   显示 ATE Connected Main Menu。
3. 在 ATE Connected Main Menu 的提示符下输入 **receive** 子命令:

   ```
   r infile
   ```

   receive 子命令指示本地系统从远程系统接收 `infile`。在传输文件之后，显示 ATE Connected Main Menu。

文件

ate.def
: 包含 ATE 缺省值。

相关信息

ate 命令。

connect 子命令、**directory** 子命令、**modify** 子命令、**send** 子命令和 **receive** 子命令。

*《AIX 5L V5.2 系统用户指南: 通信与网络》*中的『如何编辑 ATE 缺省文件』解释如何永久地更改 ATE 缺省值。

*《AIX 5L V5.2 系统用户指南: 通信与网络》*中的『ATE 概述』介绍 ATE 程序、它的菜单和控制键。

*《AIX 5L V5.2 系统管理指南: 通信与网络》*中的『系统管理的 ATE 概述』讨论了与管理 ATE 有关的任务。

xmodmap 命令

用途

修改 X 服务器中的键映射。

语法

xmodmap [**-display** *Display*] [**-e** *Expression*] [**-grammar** | **-help**] [**-n**] [**-pk**] [**-pke**] [**-pm**] [**-pp**] [**-quiet** | **-verbose**] [*FileName*]

描述

xmodmap 命令编辑并显示键盘修饰符映射和客户机应用程序用来将事件键控代码转换为键符号的键映射表。它通常从会话启动脚本运行，根据用户的个人喜好配置键盘。

每次键控代码表达式被求值时，服务器都会在每个客户机上生成一个 **MappingNotify** 事件。所有更改都应该成批进行并一次完成。接收键盘输入并忽略 **MappingNotify** 事件的客户机不会注意到对键盘映射作出的任何更改。

FileName 参数指定包含要运行的 **xmodmap** 命令表达式的文件。该文件通常保存在用户主目录中，其名称类似于 **.xmodmaprc**。如果不指定任何文件，就从**标准输入**获取输入。

xmodmap 命令程序将读取表达式列表并在试图运行其中的任一命令之前对所有这些命令进行分析。这样，我们就能够引用被自然地重新定义过的键符号，而不用过多地担心名称冲突。

add	键符号名在该行被读取时求值。这允许您从修饰符除去键，而无需担心它们是否被再分配。
add *ModifierName* **=** *KeySymbolName...*	将给定的键符号添加到表明的修饰符映射。键符号名在读取了所有输入表达式之后求值，使编写交换键的表达式变得很容易。
clear *ModifierName*	为给定的修饰符除去修饰符映射中的所有条目，其中有效的名称为 **Shift**、**Lock**、**Control**、**Mod1**、**Mod2**、**Mod3**、**Mod4** 和 **Mod5**（大小写在修饰符名中并不重要，但它会影响其它所有名称）。例如，**clear Lock** 将除去所有绑定到 **shift lock** 修饰符的键。
keycode *Number* **=** *KeySymbolName...*	将键符号分配到表明的键控代码（可以由十进制、十六进制或八进制指定，并通过运行 **/usr/lpp/X11/Xamples/demos** 目录中的 **xev** 程序决定）。通常，只有一个键符号被分配到给定的代码。
keysym *KeySymbolName* **=** *KeySymbolName...*	左边的 *KeySymbolName* 将被转换为与其匹配的、用来执行相应的一组 **keycode** 表达式的键控代码。可以在 keysym 数据库 **/usr/lib/X11/XKeysymDB** 或头文件 **X11/keysymdef.h**（没有 *XK_* 前缀）中找到 keysym 名的列表。注意：如果同一个 keysym 被绑定到多个键上，就会为每个匹配的键控代码运行表达式。
pointer = default	将指示器映射设置回缺省设置（例如，按钮 1 生成代码 1，按钮 2 生成代码 2，依此类推）。
pointer = *Button1 Button2 Button3...*	将指针映射设置为包含指定的按钮代码。列表总是以第一个物理按钮开始。
remove *ModifierName* **=** *KeySymbolName...*	从表明的修饰符映射除去所有包含给定 keysym 的键。与 **add** 不同，keysym 名在读取该行时求值。这样可以从修饰符除去键，而不需担心它们是否被再分配。

以 ！（感叹号）开头的行将作为注释处理。

如果希望更改修饰符键的绑定，必须从相应的修饰符映射除去它。

标志

-display *Display*	指定要使用的主机和显示器。
-e *Expression*	指定要运行的表达式。可以从命令行指定任意数目的表达式。
-grammar	打印一条描述文件中使用的表达式语法的帮助消息，用 **-e** *Expressions* 标志将打印到标准错误。
-help	将一条关于命令行自变量的简要描述打印到标准错误。无论何时未处理的参数提供给 **xmodmap** 命令，都会执行这个操作。
-n	表明 **xmodmap** 命令不应该更改映射，而应该显示当给定此标志时将显示的内容。
-pk	表明当前的键映射表应该打印到标准输出。
-pke	表明当前的键映射表应该按照能够反馈到 **xmodmap** 的表达式形式打印到标准输出。此标志与 X11R5 有关。
-pm	表明当前的修饰符映射应该打印到标准输出。
-pp	表明当前的指针映射应该打印到标准输出。
-quiet	关闭详细日志记录。这是缺省值。
-verbose	表明 **xmodmap** 命令应该在分析其输入时打印日志信息。

示例

1. 下面的命令逆转生成的按钮代码，使得在 3 个按钮的指针上用左手的食指去按主按钮。

   ```
   xmodmap -e "pointer = 1 2 3 4 5"
   ```

2. 下面的命令将元信息连接到多语言键（有时标记为 Compose Character）。它还利用了这样一种情况：需要 Meta 键的应用程序只需要获取键控代码，并不需要键符号位于键映射表的第一列。这意味着，需要 Multi_key（包括缺省的修饰符映射）的应用程序不会注意到任何更改。

   ```
   keysym Multi_key = Multi_key Meta_L
   ```

3. 要在逗号和句点键交换时自动生成小于或大于字符，请用下面的脚本重新设置逗号和句点的绑定。

   ```
   !
   ! make shift-, be < and shift-. be >
   !
   keysym comma = comma less
   keysym period = period greater
   ```

4. 要交换 Control 和 Shift Lock 键的位置，请使用下面的脚本:

   ```
   !
   ! Swap Caps_Lock and Control_L
   !
   remove Lock = Caps_Lock
   remove Control = Control_L
   keysym Control_L = Caps_Lock
   keysym Caps_Lock = Control_L
   add Lock = Caps_Lock
   add Control = Control_L
   ```

相关信息

X 命令。

XNSquery 命令

用途

查询 Xerox 网络系统（Xerox Network Systems，XNS）主机或路由器以获取路由信息。

语法

/usr/sbin/XNSquery*Host*

描述

/usr/sbin/XNSquery 命令从远程 XNS 主机或路由器获取路由信息。本地主机向远程主机或路由器发送数据报，使用众所周知的路由信息套接字请求获取路由信息。远程主机返回系统上维护的路由表信息。该信息包括可以从远程主机访问的网络和所需的度量值。度量值是到达目标网络所需的路段或路由器的数目。

在这个 XNS 协议的操作系统实现中，如果可能，本地主机将所有传入的数据报转发到本地主机。这样，如果路由表的维护正确，本地主机可以作为路由器使用。

示例

要获取网络 40 上的远程主机 02.4e.5f.70.83.65 所包含的路由信息或路由表，请输入:

```
XNSquery 40:02.4e.5f.70.83.65
```

这将显示如下信息:

```
from 28H.24e5f7708365.1h
          8, metric 1
          110, metric 1
          40, metric 1
          120, metric 1
```

其中 8、110、40、120 是可以从主机 40:02.4e.5f.70.83.65 访问的网络，而 metric 1 指定主机和指定网络之间的路段或网关的数目。

相关信息

netstat 命令、**route** 命令和 **XNSrouted** 守护程序。

AIX 5L Version 5.2 Communications Programming Concepts 中的 Xerox Network Systems (XNS) Overview for Programming。

XNSrouted 守护程序

用途

管理 Xerox 网络系统（Xerox Network Systems，XNS）路由表。

语法

/usr/sbin/XNSrouted [**-s**] [**-q**] [**-t**] [*LogFile*]

描述

/usr/sbin/XNSrouted 守护程序在系统启动时被调用，用以管理网络系统（Network System，NS）路由表。该守护程序使用 Xerox NS 路由信息协议（Xerox NS Routing Information Protocol，RIP）来维护最新的内核路由表条目。

在正常操作中，**XNSrouted** 守护程序侦听路由信息包。如果主机连接到多个 NS 网络，它可以周期性地为任何直接连接的主机和网络提供其路由表的副本。

当启动了 **XNSrouted** 守护程序之后，它使用 **ioctl**（SIOGIFCONF）子例程来查找那些配置到系统中并标记出的直接连接的接口（软件回送接口将被忽略）。如果存在多个接口，则假定主机将在网络间转发包。**XNSrouted** 守护程序接着将传输每个接口上的请求包（如果接口支持，就使用广播包）并进入循环，侦听来自其它主机的请求和响应包。

当接收到请求包后，**XNSrouted** 守护程序根据它的内部表中维护的信息生成一个回复。生成的响应包中包含一个已知路由的列表，每个路由都使用一个路段计数度量来标记（计数值 16 或更大就被人为是无穷大）。与返回的每个路由相关的度量都提供一个*与发送者有关的*度量。

如果满足下列情况之一，就可以使用 **XNSrouted** 守护程序收到的响应包来更新路由表:

- 目标网络或主机不存在路由表条目，而且度量表明目标可以到达；也就是说，路段计数不是无穷大。
- 包的源主机与现有路由表条目中的路由器相同。也就是说，更新后的信息从目标的包路由使用的同一个网际路由器接收。
- 路由表中现有的条目至少在 90 秒内没有被更新，而且路由至少与当前的路由一样有效。
- 新的路由描述了一个比当前路由表中存储的到目的地的路由更短的路由。**XNSrouted** 守护程序在对比新路由的度量与表中存储的度量之后更新路由。

当应用了更新之后，**XNSrouted** 守护程序记录内部表中的更改，并为所有直接连接的主机和网络生成一个响应包。**XNSrouted** 守护程序在修改内核的路由表之前最多会等待 30 秒，从而允许解决可能出现的不稳定情况。

除了处理传入的包，**XNSrouted** 守护程序还会周期性地检查路由表条目。如果条目在 3 分钟内还没有被更新，条目的度量就会被设置为无穷大，并标记为删除。删除操作将被再延迟 60 秒，以确保其无效性传播到其它路由器。

每隔 30 秒，充当网际路由器的主机就会为所有直接连接的主机和网络提供路由表。

LogFile 参数解释 **XNSrouted** 守护程序的操作应该被记录到的文件的名称。该日志文件包含关于对路由表的任何更改的信息，以及最近发送和接收的、与更改的路由有关的消息的历史记录。

XNSrouted 守护程序不作为后台处理运行，除非在命令行上指定了 &（“与”记号）。键盘输入的中断会结束该进程。

标志

-s 不管 **XNSrouted** 守护程序是否充当网际路由器，都强制它提供路由信息。
-q 处理所有传入的 XNS 包，但不提供任何 XNS 路由信息。
-t 指定发送或接收的所有包都打印到标准输出。

相关信息

netstat 命令、**route** 命令和 **XNSquery** 命令。

AIX 5L Version 5.2 Communications Programming Concepts 中的 Xerox Network Systems (XNS) Overview for Programming。

xntpd 守护程序

用途

启动网络计时协议（NTP）守护程序

语法

xntpd [**-a**] [**-b**] [**-d**] [**-m**] [**-c** *ConfigFile*] [**-e** *AuthenticationDelay*] [**-f** *DriftFile*] [**-k** *KeyFile*] [**-l** *LogFile*] [**-p** *pidFile*] [**-r** *BroadcastDelay*] [**-s** *StatsDirectory*] [**-t** *TrustedKey*] [**-v** *SystemVariable*] [**-V** *SystemVariable*]

描述

xntpd 守护程序设置和维护一个与因特网标准时间服务器相一致的 Unix 系统时间。**xntpd** 守护程序是根据 RFC1035 定义的网络计时协议（NTP）版本 3 标准的完整实现，同时也保持与根据 RFC1059 和 RFC1119 分别定义的版本 1 和版本 2 服务器的兼容性。**xntpd** 守护程序用定点算法进行所有的计算，并不需要浮点代码。

xntpd 守护程序在启动时从配置文件（**/etc/ntp.conf** 是缺省值）中读取配置。可以在命令行重设这个配置文件的名称。同样也完全可以通过命令行指定一个工作配置，尽管可指定的配置会受到限制，但不需要使用配置文件。使用这种方法将 **xntpd** 守护程序配置为一个广播客户机或多点广播客户机，以便在运行时用侦听广播的方法确定所有的同级设备设备。您可以用 **ntpq** 命令（网络计时协议（NTP）查询程序）来显示 **xntpd** 守护程序的内部变量。您可以使用 **xntpdc** 命令更改配置选项。

xntpd 守护程序有几种工作方式，包含主动均衡／被动均衡、客户机／服务器和广播／多点广播。广播／多点广播客户机能够自动发现远程服务器，计算单向延时校正因子并自动进行自我配置。这种方式使得部署一组工作站而不用针对环境指定特定的配置文件或配置详细信息成为可能。

> **注**: 当正在运行 AIX 4.2.1 或更新版本的客户机方式中进行操作时，如果在 1000 秒的本地系统单位时间内没有已经配置好的服务器，**xntpd** 守护程序将会退出并报告错误。在运行 **xntpd** 前，用 **date** 或 **ntpdate** 命令来设置一个失真系统的时间。

标志

-a	以认证方式运行。
-b	侦听广播 NTP，如果可用则与之保持同步。
-c *ConfigFile*	指定一个备用的配置文件的名称。
-d	指定调试方式。这个标志可以多次出现（最多 10 次），每次出现都表明显示器的更多详细信息。
-e *AuthenticationDelay*	以秒为单位指定时间，这将用来计算这台计算机上的 NTP 加密字段。
-f *DriftFile*	指定偏差文件的位置。
-k *KeyFile*	指定包含 NTP 认证密钥的文件的位置。
-l *LogFile*	（小写的 L）指定使用日志文件而不是记录到 syslog 中。
-m	侦听多点广播消息，如果可用则与之保持同步。假设多点广播地址为 224.0.1.1。
-p *pidFile*	指定记录守护程序进程标识的文件的名称。无缺省值。
-r *BroadcastDelay*	如果校正程序失败，则指定缺省延时（以秒为单位）。通常，**xntpd** 守护程序会自动补偿广播／多点广播服务器和客户机之间的网络延时。
-s *StatsDirectory*	指定用于创建统计信息文件的目录。
-t *TrustedKey*	将指定的密钥号添加到可信的密钥列表。
-v *SystemVariable*	添加指定的系统变量。
-V *SystemVariable*	添加指定的、以缺省方式列出的系统变量。

参考时钟支持

为了方便配置，**xntpd** 守护程序尽可能以对待类似于一般 NTP 同级设备的方式对待参考时钟。它通过地址来引用参考时钟，这和引用普通的同级设备一样，尽管它使用无效的 IP 地址来将其与普通同级设备区分开。AIX 4.2支持一种基于系统时钟类型（类型 1）的参考时钟。

参考时钟地址格式为 127.127.*Type*。*Unit*，其中 *Type* 是一个表示时钟类型的整数，*Unit* 表示特定于类型的单元号。通过在 *HostAddress* 是时钟地址配置文件中的一条服务器语句配置参考时钟。密钥、版本和 ttl 选项不用于参考时钟支持。

参考时钟支持提供 **fudge** 命令，这个命令用特殊的方式配置参考时钟。这个命令采用以下格式:

fudge 127.127.*Type***.***Unit* [**time1** *Seconds*] [**time2** *Seconds*] [**stratum** *Integer*] [**refid** *Integer*] [**flag1 0** | **1**] [**flag2 0** | **1**] [**flag3 0** | **1**] [**flag4 0** | **1**]

time1 和 **time2** 选项都是定点的秒数并作为校正常量用于一些时钟驱动程序中。

stratum 选项是一个从 0 到 15 的数，用于将非标准操作层分配给时钟。由于 **xntpd** 守护程序对每个同级设备的层数加一，主服务器一般显示第一层。为了提供工程备份， 用 **stratum** 选项指定一个大于 0 的参考时钟层。除非特别注明，该选项可以应用于所有的时钟驱动程序。

refid 选项是一个长度从 1 到 4 的 ASCII 字符串，用于分配非标准的参考标识符给时钟。

二进制标志: **flag1**、**flag2**、**flag3** 和 **flag4** 是用于定制时钟驱动程序。这些值的具体解释以及它们是否都在用要看特定时钟驱动程序的需要。

出口状态

本命令返回以下出口值:

0 成功结束。
>0 发生错误。

安全性

访问控制: 必须具有 root 权限才能运行该命令。

审计事件: N/A

示例

1. 要启动 **xntpd** 守护程序，输入:

   ```
   startsrc -s xntpd
   ```

2. 要停止 **xntpd** 守护程序，输入:

   ```
   stopsrc -s xntpd
   ```

3. 要在运行 **xntpd** 守护程序时要使用认证密钥文件 `/etc/ntp.new.keys`，输入:

   ```
   /usr/sbin/xntpd -k /etc/ntp.new.keys
   ```

文件

/usr/sbin/xntpd	包含 **xntpd** 守护程序。
/etc/ntp.conf	包含缺省配置文件。
/etc/ntp.drift	包含缺省偏差文件。
/etc/ntp.keys	包含缺省的密钥文件。

相关信息

ntpq、**ntpdate**、**ntptrace** 和 **xntpdc** 命令。

---

xntpdc 命令

用途

启动网络计时协议守护程序 **xntpd** 的查询 / 控制程序。

语法

xntpdc [**-i**] [**-l**] [**-n**] [**-p**] [**-s**] [**-c** *SubCommand*] [*Host* ...]

描述

xntpdc 命令查询 **xntpd** 守护程序有关当前状态并请求对状态的更改。它以交互方式运行，或者通过使用命令行自变量运行。**xntpdc** 命令接口显示大量的状态信息和统计信息。几乎所有那些在启动时可用 **xntpd** 守护程序配置文件指定的配置选项，也都能够在运行时用 **xntpdc** 命令指定。

如果输入 **xntpdc** 命令，并带有一个或多个请求标志，则在每个所指定的主机上运行的 NTP 服务器（或缺省为本地主机）接收每个请求。如果不输入任何请求标志，**xntpdc** 命令试图从标准输入读取命令，并在运行于所指定的第一个主机上或者缺省情况下本地主机上的 NTP 服务器上运行它们。如果标准输入是终端，则它会提示要求输入子命令。

xntpdc 命令使用 NTP 方式 7 的包与 NTP 服务器通信，并能够在允许它的网络上查询任何兼容的服务器。

xntpdc 命令不试图重新发送请求，如果远程主机在适当的时间内没有响应，它将使请求超时。

指定 **-i** 或 **-n** 之外的标志会立刻将查询请求发送至指定的主机。否则，**xntpdc** 命令尝试从标准输入读取交互式格式的命令。

标志

-c *SubCommand*	指定交互式格式的命令。此标志将 *SubCommand* 添加到在指定的主机上运行的命令列表中。可以输入多个 **-c** 标志。
-i	指定交互方式。标准输出显示提示，标准输入读取命令。
-l	（小写的 L)显示服务器已知的同级设备的列表。本选项等同于 **listpeers** 子命令。
-n	以点分十进制格式（0.0.0.0）显示所有的主机地址，而不是规范的主机名。
-p	显示服务器已知的同级设备列表及其状态的摘要。本选项等同于 **peers** 子命令。
-s	显示服务器已知的同级设备列表及其状态的摘要，但格式和 **-p** 标志不同。本选项等同于 **dmpeers** 子命令。

参数

Host ...	指定主机。

出口状态

本命令返回以下出口值:

0 成功完成。
>0 发生错误。

安全性

访问控制: 必须是系统组的成员才能运行该命令。

审计事件: N/A

显示与指定的同级设备相关联的每个同级设备的统计计数器。

示例

1. 要为网络计时协议守护程序启动查询 / 控制程序，输入:

   ```
   xntpdc
   ```

2. 要显示主机 9.3.149.107 上地址为 127.127.1.0 的同级设备的统计计数器，请输入:

   ```
   xntpdc -c "pstats 127.127.1.0" 9.3.149.107
   ```

   显示类似于以下内容的输出:

   ```
   远程主机:     LOCAL(0)
   本地接口:     127.0.0.1
   上次接收时间:     49s
   到下次发送的时间:     15s
   可达性更改:     818s
   发送的包数:     13
   收到的包数    13
   错误的认证:     0
   无效的原点:     0
   副本:     0
   错误分散度:     4
   错误的引用时间:     0
   候选者顺序:     1
   ```

xntpdc 内部子命令

可以完全用 **xntpdc** 命令运行若干交互式格式的子命令，而这些命令并不发送 NTP 方式 7 的请求命令到服务器。下列子命令只能在运行 **xntpdc** 查询程序时使用。

交互式格式的子命令

交互式格式子命令由 0 到 4 个自变量跟随的关键字组成。只需输入整个关键字的足够的字符来唯一标识子命令。子命令输出到标准输出，但是可以通过向命令行附加大于符号（**>**），然后跟随一个文件名来重定向单个子命令的输出到一个文件。

? [*SubCommand*]
显示命令用法信息。当使用时，如果没有带 *SubCommand*，则显示所有的 **xntpdc** 命令关键字的列表。当与 *SubCommand* 一起使用时，显示命令的功能和用法信息。

help [*SubCommand*]
和 **?** [*Subcommand*] 子命令相同。

delay*Milliseconds*
指定添加到需要认证的请求中包含的时间戳记的时间间隔。使用此子命令可以在长延迟网络路径上、或时钟不同步的机器之间进行不可靠服务器的重新配置。如果输入此子命令没有带自变量，则它打印子命令的当前设置。

host *HostName*
指定查询被发送至的主机。*HostName* 可以是主机名，也可以是数字地址。如果输入此子命令没有带自变量，则它打印子命令的当前设置。

hostnamesyes | **no**
指定是显示主机名（**yes**）还是数字地址（**no**）。 缺省值为 **yes**，除非使用 **-n** 标志。如果输入此子命令没有带自变量，则它打印子命令的当前设置。

keyid*Number*
指定认证配置请求的服务器密钥数字。如果输入此子命令没有带自变量，则它打印子命令的当前设置。

passwd
提示输入 NTP 服务器认证密码来认证配置请求。

quit
退出 **xntpdc** 查询程序。

timeout*Milliseconds*
指定响应服务器查询的超时周期。缺省值是 8000 毫秒。如果输入此子命令没有带自变量，则它打印子命令的当前设置。

查询子命令

xntpdc 查询子命令导致向服务器发送包含请求的 NTP 方式 7 的包。这些子命令是只读的（它们不修改服务器配置状态）。

clkbug*ClockPeerAddress* [*Addr2*] [*Addr3*] [*Addr4*]
显示参考时钟驱动程序的调试信息。一些时钟驱动程序提供这种在多数情况下手头没有驱动程序源的副本就无法解码的信息。

clockbug*ClockPeerAddress* [*Addr2*] [*Addr3*] [*Addr4*]
显示有关同级时钟的信息。所获取的值提供了有关设置干扰因子的信息和其它时钟性能信息。

dmpeers
显示服务器正在维护状态的同级设备的列表以及状态摘要。除了最左列的字符外，其它都等同于 **peers** 子命令的输出。除了显示曾包含在时钟选择算法最后阶段的同级设备外，其它的只显示了字符。

在最左列可能的字符有:

. 表明这个同级设备曾经在故障告知检测中被释放。

+ 表明这个同级设备完成它。

***** 表示服务器目前同步的同级设备。

iostats
显示在输入／输出模块中的统计计数器。

kerninfo
显示内核周期固定的循环的操作参数。只有正在生成的主机内核已经进行特别修改使之适合精度计时函数后，这个信息才可用。

listpeers
显示服务器正在维护状态的同级设备的简要列表。这些包含所有配置好的同级设备关联，以及那些在服务器看来其层次可能是将来同步候选的同级设备。

loopinfo [**oneline** | **multiline**]
显示选中的循环过滤器变量值。此循环过滤器是 NTP 的一部分，它用来调整本地系统时钟。`offset` 是包处理代码传递给循环过滤器的最后偏移量。`frequency` 是在以每百万分率（ppm）表示的本地时钟频率错误。`poll adjust` 控制周期固定的循环抗干扰性（对更改的抵抗力）和它适应振荡器偏移的速度。`watchdog timer` 是自最后的样本偏移量传递给循环过滤器以来用掉的秒数。**oneline** 和 **multiline** 选项指定显示本信息的格式。**multiline** 选项是缺省值。

memstats
显示有关内存分配代码的统计计数器。

monlist
显示监视工具收集的和维护的流量计数。

peers

显示服务器正在维护状态的同级设备的列表以及状态摘要。摘要信息包含:

- 远程同级设备的地址,
- 参考标识（0.0.0.0 用于未知参考标识）,
- 远程同级设备的层次（层次 16 表明远程同级层未同步）,
- 轮询间隔（秒）,
- 可达性寄存器（八进制）, 和
- 同级设备的当前估计延迟、偏移量和分散度（秒）。

左页边距中的字符表示同级设备条目所处的方式是:

+ 主动均衡。

- 被动均衡。

= 服务器在客户机方式下被轮询。

^ 服务器在向这个地址广播。

~ 远程同级设备正在发送广播。

***** 标记服务器目前同步的同级设备。

主机字段内容可以是主机名、IP地址、带参数的参考时钟实现名或 REFCLK（*ImplementationNumber*, *Parameter*）。当使用 **hostnames no** 时, 只能显示 IP 地址。

pstats*PeerAddress* [*Addr2*] [*Addr3*] [*Addr4*]

显示与指定的同级设备相关联的每个同级设备的统计计数器。

reslist

显示服务器的约束条件列表, 这也许有助于理解约束条件是如何应用的。

sysinfo

显示与本地服务器相关的一系列系统状态变量。除了最后四行外其它所有的都在 NTP 版本 3 规范, RFC 1305 中有描述。系统标志显示各种不同系统标志, 其中一些能够通过 **enable** 和 **disable** 配置语句设置或清除。稳定性是在应用系统频率更正后剩余的残留频率错误。可以用它来进行维护和调试。在大多数体系结构中, 这个值会从最初的高达 500ppm 下降到一个从 0.01 到 0.1 ppm 之间额定的值。如果启动了守护程序后这个值有时仍然高出正常水平, 则可能是本地时钟出了问题, 或者是内核变量 *Tick* 的值不正确。广播延时显示了缺省广播延迟, 该值是用 **broadcastdelay** 配置语句设置的, 而认证延时显示缺省认证延迟, 它由 **authdelay** 配置语句设置。

sysstats

显示在协议模块中的统计计数器。

timerstats

显示在计时器 / 事件队列支持代码中的统计计数器。

运行时配置请求子命令

服务器通过使用配置好的 NTP 密钥来认证所有导致服务器状态更改的请求命令。如果没有配置密钥, 服务器就会禁用这项功能。必须使**xtnpdc** 命令知道密钥号和相应的密钥。可以用 **keyid** 和 **passwd** 子命令来设置密钥, 它们在终端提示输入密码作为加密密钥。第一次给出在服务器上要求认证的子命令时, **xtnpdc** 命令也可以自动提示要求输入密钥号和密码。认证不仅验证请求者是否有执行上述更改的许可权, 而且防止传输错误。

如同执行对认证代码的计算一样, 认证请求总是在数据包中包含一个时间戳记。服务器将这个时间戳记与它收到这个包的时间进行比较。

如果他们的差别大于 10 秒，服务器就会拒绝请求。这使得可截取 LAN 流量的某些人在服务器上执行简单的重放攻击变得更困难。这也使得从拓扑上的远程主机请求更改服务器配置变得更困难。当重新配置工具在本地主机上与服务器工作良好，以及完全能在相同 LAN 的时间同步主机之间工作时，它与更远的主机之间的工作变得相当困难。因此，如果选择合理的密码，小心密钥的分发和保护，并应用相应的源地址限制，运行时的重新配置工具应该提供完全级别上的安全性保护。

以下子命令都执行认证请求。

addpeer*PeerAddress* [*Keyid*] [*Version*] [**prefer**]
添加配置好的、以主动均衡方式在特定地址操作的同级设备关联。可以用这个子命令删除已经存在的相同的同级设备关联或者只是将现有关联转换为符合新配置。如果 *Keyid* 是一个非零整数，所有传输到远程服务器上的外发包都将附加一个使用此密钥加密的认证字段。如果不想指定认证，将 *Keyid* 输入为 0 或保留为空白。*Version* 的值应该是 1、2 或 3，3 为缺省值。**prefer** 选项表明在可能情况下首先用于时钟同步的首选同级设备。首选同级设备也决定 PPS 信号的有效性。如果这个首选同级设备适合于同步，同样 PPS 信号也适合。

addserver*PeerAddress* [*Keyid*] [*Version*] [**prefer**]
除了运行方式是客户机以外，其它的与 **addpeer** 子命令相同。

addtrap*Address* [*Port*] [*Interface*]
设置在用于以指定本地接口地址发送消息的指定地址和端口号的异步消息陷阱。如果不指定端口号，则设置为缺省值 18447。如果不指定接口地址，则值缺省为本地接口的源地址。

authinfo
显示与认证模块相关的信息，包含已知密钥以及执行加密和解密的计数。

broadcast*PeerAddress* [*Keyid*] [*Version*]
除了运行方式是广播以外，其它的与 **addpeer** 子命令相同。*PeerAddress* 是本地网络的广播地址，或者是分配给 NTP（224.0.0.1）的多点广播组地址。

clrtrap*Address* [*Port*] [*Interface*]
清除在用于以指定本地接口地址发送消息的指定地址和端口号的异步消息陷阱。如果不指定端口号，则设置为缺省值 18447。如果不指定接口地址，则值缺省为本地接口的源地址。

delrestrict*Address Mask* [**ntpport**]
从限制列表中删除匹配的条目。

disable*Option* ...
禁用各种服务器选项。不影响没有提到的选项。**enable** 子命令描述选项。

enable*Option* ...
启用各种服务器选项。不影响没有提到的选项。可以为 *Option* 指定以下一个或多个值:

auth
使服务器仅当同级设备已经使用可信的密钥和密钥标识正确认证时才与未配置的同级设备同步。这个选项的缺省值是禁用的（off）。

bclient
使服务器侦听来自广播或多点广播服务器的消息，在此之后自动为该服务器实例化一个关联。这个自变量的缺省值是禁用的（off）。

monitor
启用监视设备，缺省值为启用（on）。

pll
使服务器能够调整本地时钟，缺省值为启用（on）。如果不设置，本地时钟会以它的本征时间和频率偏移量自由运行。当本地时钟被一些别的设备或协议控制，NTP 只是用来提供与其它客户机的同步时，此选项十分有用。

stats
启用统计工具 filegen，缺省值为启用（on）。

fudge*PeerAddress* [*Time1*] [*Time2*] [*Stratum*] [*Refid*]
提供一种为参考时钟设置某些数据的方法。

Time1 和 *Time2* 都是定点的秒数并作为校正常量用于一些时钟驱动程序中。

stratum 是一个从 0 到 15 的数，用于将非标准操作层分配给时钟。

refid 是一个长度未 1 到 4 个字符串的 ASCII 字符串，用于分配非标准的参考标识符给时钟。

monitoryes | **no**
启用或禁用监视设备。**monitor no** 子命令后跟一个 **monitor yes** 子命令是重新设置包计数的一种好方法。

readkeys
清除认证密钥的当前设置并通过重新读取在 **xntpd** 配置文件中指定的密钥文件获取新的设置。这使得可以更改加密密钥而不用重新启动服务器。

reset*Module*
清除在服务器各个模块中的统计计数器。可以为 *Module* 指定以下一个或多个值: **io**、**sys**、**mem**、**timer**、**auth**、**allpeers**。

restrict*Address Mask Option ...* 将 *Option* 值添加到现有的限制列表条目中，或者用指定的 *Option* 将新的条目添加到列表。**mask** 选项按缺省值设置为 255.255.255.255，表明 *Address* 被当作一个独立主机的地址。可以为 *Option* 指定以下一个或多个值:

ignore 忽略来自匹配此条目的主机的所有包。既不响应查询也不响应服务器轮询。

limited 指定这些主机服从相同网络的客户机限制。在上下文中的网络指的是网络的 IP 类别（A 类、B 类、C 类等等）。只接受第一个在服务器上显示并在最后 **client_limit_period** 秒内曾处于活动状态的 **client_limit** 主机。拒绝来自相同网络的其它客户机的请求。只考虑时间请求包。专用、控制和广播包都不符合客户机限制，因此对客户机计数没有影响。**xntpd** 守护程序的监视能力保存客户机的历史记录。当使用这个选项时，监视功能保持为活动状态。**client_limit** 的缺省值是 3。**client_limit_period** 的缺省值是 3600 秒。

lowpriotrap
通过将主机与低优先级状态匹配来声明陷阱设置。服务器可以维护有限数量的陷阱（当前限制是 3），这些陷阱按“先来先服务”的基础分配，并拒绝对后来的陷阱请求者提供服务。这个参数通过允许后来的普通优先级陷阱的请求覆盖低优先级的请求来修改分配算法。

nomodify
忽略所有试图修改服务器状态（运行时重新配置）的 NTP 方式 6 和 7 的包。允许返回信息的查询。

nopeer 向轮询主机提供无状态时间服务，但不分配同级设备内存资源给这些主机。

noquery
忽略来自于源的所有 NTP 方式 6 和 7 的包（信息查询和配置请求）。不影响时间服务器。

noserve
忽略 NTP 方式 6 和 7 的包。它拒绝时间服务，但允许查询。

notrap 拒绝向匹配主机提供方式 6 的控制消息陷阱服务。陷阱服务是方式 6 控制消息协议的一个子系统，供远程事件记录程序使用。

notrust
在其它方面正常使用这些主机，只是不能将它们用作同步源。

ntpport
仅当包中的源端口是标准 NTP UDP 端口（123）时才匹配约束条目。

setprecision*Precision* 设置服务器通告的精度。*Precision* 应该是一个从 -4 到 -20 负整数。

traps 显示在服务器上设置的陷阱。

trustkey*Keyid ...* 将一个或多个密钥添加到可信的密钥列表。当启用认证时，使用可信的密钥来认证具有可信时间的同级设备。

unconfig *PeerAddress* [*Addr2*] [*Addr3*] [*Addr4*] 从指定的同级设备除去配置位。在很多情况下，删除同级设备关联。但是，在合适的情况下，如果远程同级设备希望继续使用这个方式，则关联可以保持为未配置方式。

unrestrict *Address Mask Option ...* 从 *Address* 和 *Mask* 显示的限制列表条目中除去指定的选项。**restrict** 子命令描述 *Option* 的值。

untrustkey*Keyid ...* 从可信的密钥列表除去一个或多个密钥。

文件

/usr/sbin/xntpdc 包含 **xntpdc** 命令。

相关信息

ntpq、**ntpdate** 和 **ntptrace** 命令。

xntpd 守护程序。

---

xpr 命令

用途

格式化窗口转储文件以便输出到打印机上。

语法

xpr [**-append** *FileName* [**-noff**] | **-output** *FileName*] [**-landscape** | **-portrait**] [**-compact**] [**-cutoff** *Level*] [**-density** *Dpi*] [**-gray** { **2** | **3** | **4** }] [**-header** *String*] [**-height** *Inches*] [**-left** *Inches*] [**-noposition**] [**-plane** *PlaneNumber*] [**-psfig**] [**-report**] [**-rv**] [**-scale** *Scale*] [**-split** *Number*] [**-top** *Inches*] [**-trailer** *String*] [**-width** *Inches*] [**-device** *Device*] [*ImageFile*]

描述

xpr 命令使用由 **xwd** 实用程序生成的窗口转储文件作为输入，并格式化转储文件以便输出到硬件支持的所有打印机上。如果不指定文件自变量，**xpr** 命令使用标准输入。缺省情况下，**xpr** 命令会在输出页上打印出尽可能大的窗口图像。

xpr 命令选项允许添加页头和页尾，指定页边距，调整尺寸和原点，以及将多个窗口转储附加到一个输出文件。输出位置将是标准输出，除非 **-output** 标志另有指定。

标志

-append *FileName*
指定以前由 **xpr** 命令生成的，窗口要附加的文件名。（这个标志在 PostScript 打印机上不支持。）

-compact
使用简单的行程编码来压缩有大量白色像素的窗口的表示方式。这个标志表示压缩白色区域但不压缩黑色区域， 所以它对反转视频窗口没有用。

（这个标志只支持 PostScript、LIPS II+ 和 LIPSIII 输出。）

-cutoff *Level*
更改颜色映射到的白色或黑色的亮度级别，以适应激光打印机的单色输出。*Level* 变量用完全亮度的百分比表示。也可以接受小数。

-device *Device*
指定打印文件的设备。**xpr** 命令支持以下打印机:

3812 或 pp
IBM PP3812

4207 Proprinter

5201 IBM Quietwriter 1 型号 2

5202 IBM Quietwriter 2

jprinter
IBM Japanese Printer（日文数据流）

ljet HP LaserJet 和 IBM Laser Printer

ps PostScript 打印机（这是缺省值）

lips2 Canon LaserShot LIPS II+ 方式

lips3 Canon LaserShot LIPS III 方式

-density *Dpi*
指定 HP 打印机使用的每英寸点数（dpi）密度。300dpi 是缺省值。允许的密度有 300、150、100 和 75 dpi。

-gray *Number*
指定彩色图像的灰度转换，而不是黑白图像的映射。*Number* 变量必须是下列值之一:

2 2 x 2 转换

3 3 x 3 转换

4 4 x 4 转换

这个转换可以分别以两倍、三倍或四倍的比例放大图像的有效宽度和高度。

注: 这个选项只对 PostScript 打印机有效。

-header *String*
指定在窗口上打印页眉字符串。

-height *Inches*
指定页面的最大高度。

ImageFile
包含图像的捕获位图。如果不指定 *ImageFile* 参数，**xpr** 命令使用标准输入。

-landscape
强制使窗口以横向方式打印。（在显示输出中宽度大于高度。）缺省情况下，窗口的打印会使它的最长的边与纸张的最长边相对应。

-left *Inches*
以英寸为单位指定左页边距。也可以接受小数。缺省情况下，此标志使窗口打印在页面中央。

-noff
当与 **-append** 标志一起指定时，窗口和上一个窗口显示在相同的页面上。（这个标志在 PostScript 打印机上不支持。）

-noposition
使激光打印机忽略页眉、页尾以及图像放置命令生成。

-output *FileName*
指定一个输出文件名。如果不指定此选项，**xpr** 命令使用标准输出。

-plane*PlaneNumber*
指定使用图像中的哪个层面。缺省情况下，使用整个图像并根据颜色亮度将值映射为黑色和白色。这个选项不支持激光喷墨打印机。

-portrait
强制使窗口以纵向方式打印。（在显示输出中高度大于宽度。）缺省情况下，窗口的打印会使它的最长的边与纸张的最长边相对应。

-psfig
禁止将 PostScript 图片转换到页面中央。

-report
将窗口 *ImageFile* 参数的统计信息打印到标准错误。

-rv
强制使窗口以反转视频打印。

-scale *Scale*
影响页面上窗口的大小。PostScript 打印机能够将窗口像素图的每一位转换成指定大小的网格。比如，每位可能转换成 3 x 3 的网格。要指定 3 x 3 的网格，输入 **-scale 3**。缺省情况下，窗口用适合页面指定方向的最大尺寸进行打印。如果不指定设备，纵横比可能不同。

-split *Number*
将窗口分割为几个页面。这对超大型窗口十分必要，否则会导致打印机过载并以模糊的方式打印页面。（这个标志在 PostScript 打印机和 HP Laserjet 上不支持。）

-top *Inches*
以英寸为单位指定窗口顶部页边距。也可以接受小数。缺省情况下，此标志使窗口打印在页面中央。

-trailer *String*
指定在窗口下方打印的页尾字符串。

-width *Inches* 指定页面的最大宽度。

注: 4207、5201 和 5202 打印机的图像必须以 XYPixmap 或 XYBitmap 格式通过 **xwd** 实用程序记录。XYPixmap 图像可以使用阈值算法转换成位图。对于 HP Laserjet 打印机，多层图像必须以 ZPixmap 格式记录。单层图像可以是 XYPixmap、XYBitmap 或 ZPixmap 格式。

相关信息

X 命令、**xwd** 命令、**xwud** 命令。

---

xpreview 命令

用途

在 X 显示器上显示 troff 文件。

语法

xpreview [**-BackingStore** *BackingStoreType*] [**-page** *Number*] [*ToolKitFlag* ...] { *File* | **-** }

描述

xpreview 命令是一个基于AIXwindows 2.1 和 Motif2.1 的应用程序，用来显示 **troff** 命令在AIXwindows显示器上的输出。必须为 X100 设备准备 **troff** 命令的输出文件。

此用户界面包含用于调用根菜单、图标化窗口和将窗口设置为全屏大小的标准AIXwindows界面控件。此界面也包含用于文本的可滚动显示区域的主窗口。使用“下一页”、“上一页”、“转至页”、“打印页面”、“打印文件”和“新建文件”等按钮来操作查看的文档。

鼠标按钮 3 激活一个弹出菜单来配置打印机功能。菜单中包含一个设置命令行的选项和另一个用于选择打印队列的选项。命令行对话框通过 **troff** 命令来引导命令行的输入。例如，

```
pic -Tibm3816 troff-input-file |tbl|troff -mm -Tibm3816
```

是一个可接受的命令行。打印机队列选项显示已配置的打印机队列的列表。如果不选择这个选项，**xpreview** 命令使用系统定义的缺省队列。

当预览输入文件时，“打印页面”和“打印文件”按钮要求命令行输入。注意：一旦选择了打印机队列，在查看会话的持续时间内它保持为选定状态，直到选中备用打印机队列。

欧洲语言环境中 devX100 设备的支持字体是:

- Times New Roman，普通字体、斜体和粗体
- Courier，普通字体和粗体
- Helvetica，普通字体和粗体
- Symbol

xpreview 命令支持以下字体大小：8、10、14、18、24、30 和 36。

xpreview 命令不显示由 **troff** 命令产生的文件，该命令是为本文档中没有描述的设备构造的。

要在某个设备上预览文件，**xpreview** 命令需要在以下目录中能找到字体:

- **/usr/lib/X11/fonts** 目录，用于除日语之外字体文件的格式化文件
- **/usr/lib/X11/fonts/JP**，用于日语字体文件

多字节支持

xpreview 命令支持多字节语言环境。而且要显示日文字符，必须安装日文的 16 点阵字体（部分日文 BSL 包）、24 点阵和 32 点阵字体（部分AIXwindows字体包）。要现实韩国语字符，必须安装韩国语字体（部分韩国语 BSL 包）。

日语支持当前包含以下字体集:

- 16 点阵: RomanKn12、Kanji12 和 IBM_JPN12
- 24 点阵: RomanKn17、Kanji17 和 IBM_JPN17
- 32 点阵: RomanKn23、Kanji23 和 IBM_JPN23、或 RomanKn23G、Kanji23G 和 IBM_JPN23G

韩国语支持当前包含以下字体集:

- 16 点阵: EnglHg16 和 Hangul16
- 24 点阵: EnglHg24 和 Hangul24

标志

xpreview 命令接受标准 **X** Toolkit 命令行标志和以下标志:

-
需要从标准输入中读取输入。

-help
表明应当打印允许的命令行标志的简单摘要。

-BackingStore *BackingStoreType*
-BackingStore 标志导致服务器保存窗口内容以便在视区中滚动时，窗口根据保存在服务器后备存储器中的内容进行绘制。重新显示绘制的窗口可能要花一秒钟左右。*BackingStoreType* 参数可以是以下值: **Always**、**WhenMapped** 或 **NotUseful**。

注:

1. 在 **-BackingStore** 标志和它的 *BackingStoreType* 参数之间输入一个空格。
2. 使用这个标志需要服务器启动时启用后备存储器。

-page *Number*
指定首先显示的文档页码。

ToolKitFlag
以下标准的 **X** Toolkit 标志通常与 **xpreview** 命令一起使用:

-bg *Color*
指定窗口背景使用的颜色。缺省值是白色。

-bg *Color*
指定窗口背景使用的颜色。缺省值是白色。

-fg *Color*
指定用于显示文本的颜色。缺省值为黑色。

-geometry *Geometry*
指定窗口的首选大小和位置。

-display *Host*:*Display*
指定要联系的 **X** 服务器。

-xrm *ResourceString*
指定要使用的资源字符串。

File 指定要打印的文件。

示例

1. 要通过 **troff** 命令将文件输出构建到适合和 **xpreview** 命令一起使用的文件中，请输入以下命令:

```
troff-TX100 troff-input | xpreview
pic -TX100 pic-troff-input | tbl | troff -man -TX100 | xpreview
```

2. 要通过 **troff** 命令将文件输出构建到适合和日语版本 **xpreview** 命令一起使用的文件中，请输入以下命令:

```
LANG=ja_JP
troff -TX100 troff-input | xpreview -
pic -TX100 pic-troff-input | tbl | troff -man -TX100 \
        | xpreview -
```

文件

/usr/lib/X11/app-defaults/XPreview	包含用户可配置的应用程序缺省文件。
/usr/lib/X11/Ja_JP/app-defaults/XPreview	包含用户可配置的日文（IMB-932）语言环境的应用程序缺省文件。
/usr/lib/X11/ja_JP/app-defaults/XPreview	包含用户可配置的日文（IBM-eucJP）语言环境的应用程序缺省文件。
/usr/lib/X11/ko_KR/app-defaults/XPreview	包含用户可配置的韩国语语言环境的应用程序缺省文件。
/usr/lib/X11/zh_TW/app-defaults/XPreview	包含用户可配置的繁体中文语言环境的应用程序缺省文件。
/usr/lib/font/devX100	包含 devX100 设备的 troff 字体。
/usr/lib/X11/fonts	包含 100dpi 设备的 X 字体。
/usr/lib/X11/fonts/JP	包含多字节字符的 X 字体。
/usr/lib/X11/fonts/JP	包含日文字符的 X 字体。

相关信息

cat 命令、**csplit** 命令、**diff** 命令、**lint** 命令、**lp** 命令、**lpr** 命令、**pg** 命令、**pr** 命令、**qprt** 命令，**sed** 命令、**sort** 命令、**tabs** 命令、**X** 命令、**xrdb** 命令。

eqn 命令、**grap** 命令、**pic** 命令、**tbl** 命令、**troff** 命令、**X** 命令、**xrdb** 命令。

AIX 5L Version 5.2 Technical Reference: Base Operating System and Extensions **nl_langinfo** 子例程。

---

xrdb 命令

用途

X 服务器资源数据库实用程序。

语法

xrdb [**-display** *Display*] [**-help**] [**-quiet**] [**-retain**] [**-cpp** *FileName* | **-nocpp**] [**-D** *Name*=*Value*] [**-I** *Directory*] [**-U** *Name*] [**-all** | **-global** | **-screen** | **-screens**] [**-n**] [**-edit** *FileName* | [**-backup** *String*] | **-merge** [*FileName*] | **-load** [*FileName*] | **-query** | **-remove** | **symbols**] **-override**]

描述

xrdb 命令在根窗口的屏幕 0 上获取或设置 RESOURCE_MANAGER 属性的内容，或在根窗口的所有或任何屏幕（或任何组合）的 SREEN_RESOURCES 属性。通常从 X 启动文件中运行这个程序。

大部分 X 客户机都用 RESOURCE_MANAGER 和 SCREEN_RESOURCES 属性来获取颜色、字体以及其它一些用于应用程序的用户首选项。将这些信息存储在服务器（其中的信息对所有用户都可用）而不是在磁盘上，这样解决了以前版本的 X 中需要在所有使用的机器上都需要维护 *defaults* 文件的问题。同时这样也支持动态更改缺省值而不需要编辑文件。

RESOURCE_MANAGER 属性指定应用于显示器所有屏幕的资源。每个屏幕的 SCREEN_RESOURCES 属性指定了该屏幕使用的附加（或基本的）资源。（当只有一个屏幕时，SCREEN_RESOURCES 通常没有用；所有资源都放在 RESOURCE_MANAGER 属性中。）

为保持兼容性，如果没有定义 RESOURCE_MANAGER 属性（或因为 **xrdb** 命令没有运行，或如果这个属性已被除去），资源管理器会在主目录里查找一个称为 **.Xdefaults** 的文件。

根据服务器使用的能力，文件名（或如果没有给定文件名，则为标准输入）会根据以下定义的符号有选择地经过 C 预处理器处理:

SERVERHOST=*Hostname*	指定连接的显示器的主机名部分。
SRVR_*name*	将 **SERVERHOST** 主机名字符串转换成合法标识符。例如 `my-dpy.lcs.mit.edu` 转换成 `SRVR_my_dpy_lcs_mit_edu`。
HOST=*Hostname*	指定连接的显示器的主机名部分。
DISPLAY_NUM=*num*	指定服务器主机上的显示器号。
CLIENTHOST=*Hostname*	指定运行 xrdb 的主机名。
CLNT_*name*	将 **CLIENTHOST** 主机名字符串转换成合法标识符。例如 `my-dpy.lcs.mit.edu` 转换成 `SRVR_my_dpy_lcs_mit_edu`。
WIDTH=*Number*	指定缺省屏幕的宽度（以像素值计算）。
HEIGHT=*Number*	指定缺省屏幕的高度（以像素值计算）。
X_RESOLUTION=*Number*	指定缺省屏幕的 x 分辨率（以像素／米计算）。
Y_RESOLUTION=*Number*	指定缺省屏幕的 y 分辨率（以像素／米计算）。
PLANES=*Number*	指定缺省值屏幕的根窗口的位平面个数。
RELEASE=*Number*	指定服务器的供应商发行版号码。这个号码的解释随 **VENDOR** 的变化而有所不同。
REVISION=*Number*	指定服务器支持的 X 协议小版本（当前为 0）。
VERSION=*Number*	指定服务器支持的 X 协议主版本号（应当始终为 11）。
VENDOR=*Vendor*	用于指定服务器供应商的字符串。
VNDR_*name*	将 **VENDOR** 名称字符串转换成合法标识符。例如，`MIT X Consortium` 转换成 `VNDR_MIT_X_Consortium`。
EXT_*name*	将每个扩展字符串转换成合法标识符。对服务器支持的每个协议扩展名都定义了一个符号。例如，`X3D-PEX` 转换成 `EXT_X3D_PEX`。
NUM_SCREENS=*num*	指定屏幕总数。
SCREEN_NUM=*num*	指定当前屏幕的号码。号码从 0（零）开始。
BITS_PER_RGB=*Number*	指定在 RGB 颜色规格中的有效位数。这是指从初级开始由硬件能够生成多少级以 2 为底以这个数为幂的灰度阴影。注意：这和 **PLANES** 没有关系。
CLASS=*VisualClass*	指定缺省屏幕的根窗口的可视类，这些类是以下值之一:
CLASS_*visualclass*=*visualid*	以支持 *#ifdef* 的方式指定根窗口的可视类。此值是该视觉的数字标识。
	DirectColor、**GrayScale**、**PseudoColor**、**StaticColor**、**StaticGray**、**TrueColor**

CLASS_*visualclass_depth=num*
对屏幕支持的每种视觉都定义了一个符号。这个符号包括视觉的类和深度；这个值是视觉的数字标识。（如果有多个视觉的类和深度相同，则使用向服务器报告的第一个视觉的数字标识。）

COLOR
只在 **CLASS** 是以下值之一时才定义：**StaticColor**、**PseudoColor**、**TrueColor** 或 **DirectColor**。

注释行以 ！（感叹号）开头并被忽略。

由于 **xrdb** 可以从标准输入中读取，可以用它直接从终端或从 shell 脚本更改属性内容。

标志

-all
表明操作是在显示器的每个屏幕上对不依赖屏幕的资源属性（RESOURCE_MANAGER）和特定于屏幕的属性（SCREEN_RESOURCES）执行的。例如，当和 **-query** 一起用时，所有属性的内容都会输出。对于 **-load** 和 **-merge**，对每个屏幕都处理一遍输入文件。在输出中公共使用的资源会被收集起来，并作为不依赖于屏幕的资源应用。剩下的资源会应用于每个单独的屏幕属性。这是操作的缺省方式。该选项是 X11R5 特有的。

-backup *String*
指定一个后缀追加到文件名。该选项与 **-edit** 一起使用来生成备份文件。**-edit** 是 **-backup** *String* 的先决条件。

-cpp *FileName*
指定要使用的 C 预处理器程序的路径名。尽管 **xrdb** 命令是设计为使用 CPP 的，但可以使用任何起过滤器作用并接受 **-D**、**-I** 以及 **-U** 标志的程序。

-D*Name*=*Value*
经过预处理器处理并定义符号在一些条件使用，比如 `#ifdef`。

-display *Display*
指定要使用的 X 服务器。它也指定 **-screen** 选项所用的屏幕，同时它指定为 **-global** 选项派生预处理器符号的屏幕。

-edit *FileName*
表明指定属性的内容应该在给定文件中编辑，并替换该处列出的全部值。这样可用将已对缺省值的更改保存回资源文件，并保留所有注释和预处理器行。

-global
表明该操作只应对不依赖屏幕的 RESOURCE_MANAGER 属性执行。该选项是 X11R5 特有的。

-help
打印允许标志的简短描述。

-I*Directory*
（大写的 i）通过预处理器处理并指定目录来搜索 `#include` 引用的文件。

-load
表明输入是作为指定属性的新值装入的，并替换旧的内容。这是缺省操作。

-merge
表明输入与指定属性的当前内容合并而不是替换当前内容。该选项对两个输入运行字典式分类合并，这也许不是想要的，但为了保持向后兼容性保留了此方式。

-n
表明对指定的属性的更改（当与 **-load** 或 **-merge**一起使用时）或对资源文件的更改（当与 **-edit** 一起使用时）应该显示在标准输出上，但不应该执行。该选项是 X11R5 特有的。

-nocpp
表明 **xrdb** 命令不应该在把输入文件装入属性之前使用预处理器运行它。

-override
表明将输入添加到指定属性的当前内容而不是替换当前内容。新的条目覆盖以前的条目。

-query
表明指定属性的当前内容应该打印到标准输出。注意：既然在输入资源的预处理器命令是输入文件的一部分，而不是属性的一部分，它们并不在该标志的输出中显示。

-quiet
表明不必显示条目重复的警告。该选项是 X11R5 特有的。

-remove
表明指定的属性应当从服务器上除去。

-retain
表明如果 **xrdb** 命令是第一个客户机，应当指示服务器不要复位。这在正常情况下是不必要的，因为**xdm** 和 **xinit** 命令始终担当第一个客户机的作用。该选项是 X11R5 特有的。

-screen
表明该操作只应对显示器的缺省屏幕的 SCREEN_RESOURCES 属性执行。该选项是 X11R5 特有的。

-screens
表明该操作只应对显示器的每个屏幕的 SCREEN_RESOURCES 属性执行。对于 **-load** 和 **-merge**，对每个屏幕都处理一遍输入文件。该选项是 X11R5 特有的。

-symbols
表明为预处理器定义的符号应当打印到标准输出。

-U*Name*
经过预处理器处理并除去该符号的所有定义。

示例

1. 要将文件装入数据库:

   ```
   xrdb -load myfile
   ```

2. 要获取刚装入的数据库内容并编辑或将它写入 newfile:

   ```
   xrdb -edit newfile
   ```

文件

xrdb 命令归纳了 **~/.Xdefaults** 文件。

---

xrx 命令

用途

RX 帮助器程序。

语法

xrx [**-***ToolKitOption* ...] *filename*

描述

帮助器程序可以与任何 web 浏览器一起使用，用来解释 RX MIME 类型格式的文档并启动远程应用程序。

xrx 读入用文件名指定的 RX 文档，从那里它获取应用程序要使用的服务列表。根据这个信息，**xrx** 设置不同的请求服务，包括如果 X 服务器支持 SECURITY 扩展则创建授权密钥。然后它通过一个相关的 CGI 脚本的 HTTP GET 请求，将相关数据，例如 X 显示器名称，传递到应用程序。Web 服务器接着执行 CGI 脚本以启动应用程序。客户机在连接到 X 服务器上的 Web 服务器主机上运行。

安装

需要配置 web 浏览器以使用**xrx** 打开 RX 文档。通常 **$HOME/.mailcap** 中有下列行就足够了:

```
application/x-rx; xrx %s
```

不过，可能需要参看 web 浏览器文档中关于配置帮助器应用程序的确切的指导信息。

一旦正确地完成配置，无论何时检索任何 MIME 类型 application/x-rx 的文件，浏览器就将激活帮助器。

标志

-*ToolKitOption*

这个**xrx** 帮助器接受所有标准 X Toolkit 命令行选项，例如:

xrm resourcestring

本选项指定要使用的资源字符串。在命令行也许有几个该选项的实例。

资源

xrx 程序的应用程序类名是 **Xrx**，它可以接受并解释以下应用程序资源名和类:

xrxHasFirewallProxy（类 **XrxHasFirewallProxy**）
指定 X 服务器防火墙是否正在运行且是否应当使用。缺省值为 “False”。

xrxInternalWebServers（类 **XrxInternalWebServers**）
不应使用 X 服务器防火墙代理的 Web 服务器（只有当 **xrxHasFirewallProxy** 是 “True” 有关系）。它的值是逗号分隔的掩码／值对列表，它们用于根据其地址来过滤内部 Web 服务器。 “掩码” 部分指定要考虑地址的哪些段而 “值” 部分则指定结果应该匹配的内容。例如，以下列表:

255.255.255.0/198.112.45.0,
255.255.255.0/198.112.46.0

匹配地址集: 198.112.45.* 和 198.112.46.*。更加精确地说，测试就是(address &mask) == value。

xrxFastWebServers（类 **XrxFastWebServers**）
不应使用 LBX 的 Web 服务器。资源值是一个地址掩码／值对列表，这在以上已经描述。

xrxTrustedWebServers（类 **XrxTrustedWebServers**）
远程服务器应当作为可信的客户机运行的 Web 服务器。缺省值是作为不可信的客户机运行远程应用程序。资源值是一个地址掩码／值对列表，这在以上已经描述。

环境

xrx 帮助器使用标准 X 环境变量，比如 DISPLAY，用以获取缺省的 X 服务器主机和显示器号码。如果 RX 文档请求 X-UI-LBX 服务而缺省 X 服务器没有通告 LBX 扩展，**xrx** 将寻找环境变量 XREALDISPLAY 来获取 X 服务器的另一地址并在该处寻找 LBX 扩展。当通过 **lbxproxy** 运行 Web 服务器时，如果希望远程应用程序能够通过因特网使用 LBX 服务，则需要将 XREALDISPLAY 设置为服务器的确切地址。

如果 RX 文档请求 XPRINT 服务，**xrx** 将寻找变量 XPRINTER、PDPRINTER、LPDEST、PRINTER 和 XPSERVERLIST 来获取要使用的打印机名和 X 打印服务器的地址。注意: 尽管变量设置允许指定多个服务器和打印机，但只使用第一个服务器和打印机对。最后，如果正在使用防火墙代理，**xrx** 将寻找 PROXY_MANAGER 来获取代理管理器的地址（请参阅**proxymngr**）。当没有指定时，它会使用 :6500 作为缺省值。

相关信息

libxrx 命令、**lbxproxy** 命令和 **proxymngr** 命令。

---

xsend 命令

用途

通过安全的通信通道发送保密邮件。

语法

xsend *User*

描述

xsend 命令发送只能由预定的收件人读取的消息。这个命令类似于 **mail** 命令，但用这个命令发送的邮件应该是保密的。

xsend 命令与 **enroll** 命令及 **xget** 命令一起使用来发送和接收保密邮件。**enroll** 命令设置用于接收保密邮件的密码。**xget** 命令使用这个密码接收邮件。

xsend 命令读取标准输入，直到输入 EOF（Ctrl-D）或者 .（句点）为止。然后它将一些头信息和这个文本一起进行加密并将其发送出去。在发送加密消息后，**xsend** 命令发送一个标准邮件消息到收件人以通知他们已接收保密邮件。

注: 保密邮件只能发送到本地用户。

示例

1. 要发送保密邮件，输入:

   ```
   xsend ron
   ```

   当已经用收件人的名字发出 **xsend** 命令时，使用邮件系统输入文本消息。当完成输入发送到用户 `ron` 的消息时，按 Enter 键，然后是 Ctrl-D 或者是一个 .（句点）来退出邮件编辑器并发送该消息。**xsend** 命令在消息发送前对其进行加密。

2. 要将文件发送到另一个用户，输入:

   ```
   xsend lance <proposal
   ```

   在本示例中，文件 `proposal` 被发送给用户 `lance`。

文件

/var/spool/secretmail/*.keys	包含 *User* 的加密密钥。
/var/spool/secretmail/*.[0-9]	包含 *User* 的加密邮件消息。
/usr/bin/xsend	包含命令可执行文件。

相关信息

bellmail 命令、**enroll** 命令、**mail** 命令、**xget** 命令。

*《AIX 5L V5.2 系统用户指南: 通信与网络》*中的『邮件概述』、『发送和接收保密邮件』。

---

xset 命令

用途

为 X-Windows 环境设置选项。

语法

xset [**-display** *Display*] [**b** [*Volume* [*Pitch* [*Duration*]]] | **-b** | **b on** | **b off**] [**bc** | **-bc**] **c** [*Volume*] | **-c** | **c on** | **c off**] [[**-** | **+**] **fp** [**-** | **+** | **=**] **Path** [**,***Path***,** [...]]] [**fp default**] [**fp rehash**] [[**-**] **led** [*Integer*]] [**led on** | **led off**] [**m** [*Accelerator*] [*Threshold*]]] [**m** [**ouse**] **default**] [**p** *Pixel Color*] [[**-**] **r**] [**r on** | **r off**] [**s** [*Length* [*Period*]]] [**s blank** | **s noblank**] [**s expose** | **s noexpose**] [**s on** | **s off**] [**s activate**] [**s reset**] [**s default**] [**q**]

描述

xset 命令定制 X-Windows 环境。

标志

-display *Host*:*Display*
指定要使用的 X 服务器。有关服务器的更多信息，请参阅 **X** 命令。

b 或 **b on**
打开响铃。这是缺省设置。

> **注**：不是所有硬件都可以更改响铃特征，但对可以更改响铃特性的硬件，所有的 **b** 标志变换和变量都是可用的。

b [*Volume* [*Pitch* [*Duration*]]]
指定响铃音量、音调和持续时间。这个标志最多可以接受三个数值。

Volume 如果只给定一个数字，则采用 *Volume*。根据当前硬件能力，响铃音量会以最大可能音量的百分比的形式设置该数值。

Pitch 第二个数字，以赫兹值计数，是响铃的声调。

Duration
第三个数字以毫秒为单位，指响铃时间长度。

-b 或 **b off**
关闭响铃。

bc 或 **-bc**
如果可能，在服务器上控制程序错误兼容性方式。前面的 **-**（破折号）禁用这个方式，否则，启用程序错误兼容性方式。要使 **bc** 标志有效，服务器必须支持 MIT-SUNDRY-NONSTANDARD 协议扩展。

新的应用程序开发应该在程序错误兼容性方式禁用环境下进行。

bc 标志是为 X11 R4（X11R4）以前版本的客户机提供的。一些 X11R4 以前版本的客户机在各种协议请求下传递非法值。当与 X11R4 服务器一起运行时，这些客户机异常结束或无法正确操作。

该标志将这些程序错误显式引入 X 服务器以便这些客户机能够运行。

c 或 **c on**
打开嘀嗒声功能。这是系统缺省值。

c *Volume*
一个从 0 到 100 的数字，指定在当前硬件能力下最大可能音量的百分比。

-c 或 **c off**
关闭嘀嗒声功能。

fp=*Path***,...**
将字体路径设置为 *Path* 参数中给定的目录。目录由服务器解释，而不是客户机，且依赖于服务器。服务器忽略不包含由 **mkfontdir** 命令创建的字体数据库的目录。所有 **fp** 标志支持的选项和变量都可用。

fp- 或 **-fp**
如果 **-**（破折号）在 **fp** 前，从当前字体路径结尾删除由 *Path* 参数指定的字体路径，或是如果 **-**（破折号）在**fp** 之后，则从字体路径开始删除。

fp+ 或 **+fp**
如果 **+**（加号）在**fp** 前，则添加由 *Path* 参数指定的字体路径到字体列表的底部；如果 **+**（加号）在 **fp** 后，则从字体路径结尾添加。

fp default
将字体路径复位为服务器缺省值。

fp rehash
使服务器重读当前字体路径下的字体数据库。通常只在运行 **mkfontdir** 重新创建字体数据库后添加字体到数据库才会使用。

led 或 **led on**
打开所有 LED。

-led *Integer*
关闭由 *Integer* 指定的 LED。有效值应该在 1 到 32 之间。

led *Integer*
打开由 *Integer* 指定的 LED。有效值应该在 1 到 32 之间。

-led 或 **led off**
关闭所有 LED。

注: 不是所有的硬件都会分配相同的 *Integer* 变量给相同的 LED 功能。

m
允许控制鼠标或其它定位设备的精度。如果没有指定 **default** 自变量或变量，将使用系统缺省值。这个标志接受以下可选的自变量和参数:

Acceleration
设置鼠标移动的放大系数。值可以指定为整数或小数。

Threshold
设置调用鼠标移动的最小像素值。该值指定为像素个数。

如果只给定一个参数，则该参数作为 *Acceleration* 参数解释。

default 使用系统缺省值。

p
控制像素颜色值。通过更改 BlackPixel 和 WhitePixel 的条目，可以更改一些服务器上的根背景色。尽管这些值总是 **0** 和 **1**，但它们一般不需要更改。

而且，服务器可以选择私下分配那些颜色，此时 **xset** 命令产生错误。**xset** 命令在映射条目是只读颜色时也会产生错误。

有效参数是:

Pixel 用十进制数指定颜色映射条目的数目。

Color 指定一种颜色。

r 或 **r on**
启用自动重复。

-r 或 **r off**
禁用自动重复。

s 或 **s default**
将屏幕保护程序参数设置为缺省屏幕保护程序特征。

s [*Length*[*Period*]]
指定激活屏幕保护程序前服务器保持不活动的时间长度。*Period* 指定为免感觉单调而必须更换背景模式的周期。*Length* 和 *Period* 值必须以秒为单位指定。如果只给定一个数字参数，则它作为 *Length* 参数解释。

s on 或 **s off**
分别打开或关闭屏幕保护程序的功能。

s activate
使屏保功能激活，即使该功能已经被关闭。

s reset
如果屏幕保护程序已经激活，那么使它停用。

s blank
将首选项设置为黑屏（如果硬件支持）而不是显示背景模式。

s noblank
将首选项设置为显示模式而不是黑屏。

s expose
将首选项设置为允许暴露窗口（服务器能够自由地废弃窗口内容）。

s noexpose
将首选项设置为禁用屏幕保护程序直到服务器可以重新生成屏幕而没有造成暴露事件。

q
报告当前设置信息。

注销时这些设置会复位为缺省值。

注: 不是所有的 X 实现都保证这些选项的有效性。

示例

1. 要设置响铃音量为中等，音调为 50 赫兹，响铃时间为 50 毫秒:

   ```
   xset b 50,50,50
   ```

2. 要将字体路径设置为 **/usr/lib/X11/fonts** 目录:

   ```
   xset fp= /usr/lib/x11/fonts
   ```

3. 要使服务器重读当前字体路径下的字体数据库:

   ```
   xset fp rehash
   ```

4. 要查看当前设置的信息:

   ```
   xset q
   ```

   该命令产生的输出类似于以下内容:

   ```
   Keyboard Control:
     auto repeat:  on    key click percent:  0    LED mask:  00000000
     auto repeating keys:  0000000000000000
                           0000000000000000
                           0000000000000000
                           0000000000000000
     bell percent:  50    bell pitch:  400    bell duration:  100

   Pointer Control:
     acceleration:  2 = 2 / 1    threshold:  4

   Screen Saver:
     prefer blanking:  no    allow exposures:  no
     timeout:  0    cycle:  0

   Colors:
     default colormap:  0x8006e    BlackPixel:  0    WhitePixel:  1

   Font Path:
      /usr/lib/X11/fonts/,/usr/lib/X11/fonts/75dpi/,/usr/lib/X11/fonts/100dpi/,/usr/
   lib/X11/fonts/oldx10/,/usr/lib/X11/fonts/oldx11/,/usr/lib/X11/fonts/bmug/,/usr/l
   ib/X11/fonts/info-mac/,/usr/lib/X11/fonts/JP/,/usr/lib/X11/fonts/misc/
   ```

相关信息

X 命令、**xmodmap** 命令、**xrdb** 命令、**xsetroot** 命令。

---

xsetroot 命令

用途

为 **X** 命令设置根窗口参数。

语法

xsetroot [**-bg** *Color*] [**-cursor** *CursorFile MaskFile*] [**-cursor_name** *CursorName*] [**-def**] [**-display** *Display*] [**-fg** *Color*] [**-help**] [**-name** *String*] [**-rv**] [**-bitmap** *FileName* | **-gray** | **-grey** | **-mod** *X Y* | **-solid** *Color*]

描述

xsetroot 命令允许在运行 X 的工作站显示器上定制背景（根）窗口的外观。通常，会试用 **xsetroot** 命令直到发现喜欢的个性化外观，接着将生成这种效果的 **xsetroot** 命令放入 X 启动文件中。如果没有指定选项或指定 **-def** 标志，窗口复位为缺省状态。**-def** 标志可以用其它标志指定,而只有未指定的特征才复位为缺省状态。

一次只能指定一种背景色（平铺）更改标志（**-bitmap**、**-solid**、**-gray**、**-grey** 或 **-mod**）。

标志

-bg *Color*	使用 *Color* 参数作为背景色。
-bitmap *FileName*	使用在文件中指定的位图来设置窗口模式。可以用 **bitmap** 程序来制作自己的位图文件（小图片）。整个背景由重复的平铺位图组成。
-cursor *CursorFile MaskFile*	当指针光标在任何窗口外部时将其更改成需要的形状。光标和掩码文件是 **bitmap** 程序可以制作的位图（小图片）。在习惯掩码的工作方式以前，您可能希望所有的掩码文件都为黑色。
-cursor_name *CursorName*	将指针光标更改为光标字体中的一种标准光标形状。
-def	将未指定的属性复位为缺省值。（将背景恢复为熟悉的灰色网格，将光标恢复为中空的 x 形状。）
-display *Display*	指定服务器连接。请参阅 **X** 命令。
-fg *Color*	使用 *Color* 参数作为前景色。前景色和背景色只有在使用 **-cursor**、**-bitmap** 或 **-mod** 标志时才有意义。
-gray	使整个背景更改为暗灰色。
-grey	使整个背景更改为灰色。
-help	打印一条使用消息并退出。
-mod *X Y*	在屏幕上使用像格子花纹一样的网格模式。*X* 和 *Y* 参数是范围在 1 到 16 之间的整数。零和负数都作为 1 对待。
-name *String*	将根窗口的名称设置为 *String* 参数。无缺省值。通常，分配给每个窗口一个名称，这样当窗口被图标化时，窗口管理器可以使用一个文本表示法。因为无法图标化背景，所以没有使用这个标志。
-rv	交换前景色和背景色。通常，前景色是黑色，背景色是白色。
-solid *Color*	将根窗口的背景设置为指定颜色。此标志只用于颜色服务程序。

相关信息

X 命令、**xset** 命令、**xrdb** 命令。

---

xss 命令

用途

提高无人照管工作站的安全性。

语法

xss [**-e** *CommandString*] [**-timeout** *Seconds*] [**-display** *DisplayPtr*] [**-v**] [**-fg** *Color*] [**-bg** *Color*] [**-geometry** *wxh+x+y*]

描述

xss 命令运用了新添加的麻省理工学院（MIT）屏幕保护程序扩展，来实现用户可控制的屏幕保护程序 / 屏幕锁定。这个命令是为提高无人照管工作站的安全性而设计的。

xss 命令在接收屏幕保护程序超时消息时，或者用户激活按钮时执行用户指定的命令字符串。当没有给定用户指定命令时，**xss** 命令缺省为 **xlock** 命令。

> **注**: **xss** 命令只使用新添加的 MIT 屏幕保护程序扩展。**xss** 命令在老式的 X 服务器或使用旧的 X 扩展库时不起作用。

标志

-e *CommandString*	设置当屏幕保护程序超时或用户激活按钮时执行 **xss** 命令。注意：如果 *CommandString* 参数值长于一个单词，必须用 ″ ″（双引号）引起来。
-timeout *Seconds*	设置在屏幕保护程序超时之前的用户无活动的秒数，并使 **xss** 命令运行 *CommandString* 参数。
-display *DisplayPtr*	设置到 X11 显示器的连接。
-v	打开详细方式。
-fg *Color*	设置按钮的前景色。
-bg *Color*	设置按钮的背景色。
-geometry *wxh+x+y*	指定客户机窗口的大小和位置。

示例

当远程运行并用 **-display** 标志运行 **xss** 命令时，请记住可能也必须使用命令的 **-display** 标志选项，该命令由 **xss** 命令执行。请参阅以下远程运行的示例:

1. 远程运行:

   ```
   xss -display myhost:0 -e "xlock -remote -display myhost:0"
   ```

2. 仅屏幕保护程序:

   ```
   xss -e "xlock -nolock"
   ```

3. 简单示例:

   ```
   xss -e xlock
   ```

---

xstr 命令

用途

从 C 程序中提取字符串以实现共享字符串。

语法

xstr [**-v**] [**-c**] [**-**] [*File*]

描述

xstr 命令维护一个名为 **strings** 的文件，大型程序各组件的字符串散列到该文件中。这些字符串替换为此数组的引用。这适合于实现共享常量字符串，如果它们也是只读属性则更有用。

命令:

```
xstr -c File
```

从 *File* 参数的 C 源文件中抽取字符串，用（**&xstr**[*number*]）格式的表达式来替换某些数字的字符串引用。xstr 数组的适当声明添加到文件首部。得到的 C 文本放入文件 **x.c** 中，然后进行编译。从文件中读出的字符串会追加到 **strings** 文件中（如果它们在该处尚不存在。）重复的字符串和作为现有字符串后缀的字符串不会造成文件 **strings** 的更改。

如果某字符串是文件中另一字符串的后缀，但是这个较短的字符串是先被 **xstr** 命令读入，那么两个字符串都存放于文件 **strings**。

在一个大型程序的所有组件都被编译后，声明公共 xstr 数组空间的文件 **xs.c** 可以用以下格式的命令创建:

```
xstr
```

接下来应编译 **xs.c** 文件并装入剩下的程序。如果可能，这个数组会被保存为只读（共享）以节省空间和数据交换的开销。

xstr 命令也可以用于单个文件。命令:

```
xstr File
```

像以前一样，创建文件 **x.c** 和 **xs.c**，但是没有使用或影响任何在相同目录下的 **strings** 文件。

如果任何宏定义产生字符串，或者如果存在条件代码包含字符串而事实上并不需要这些字符串，在经过 C 预处理器编译后运行 **xstr** 命令十分有用。

当给定 **-**（负号）标志时，**xstr** 命令读取标准输入，并且除非同时指定 **-c** 标志，否则不会更改 **strings** 文件。

在经过 C 预编译器编译后运行 **xstr** 命令的正确命令序列是:

```
cc -E name.c | xstr -c -
cc -c x.c
mv x.o name.o
```

除非添加新项目，**xstr** 命令并不改变文件 **strings**；因而除非真的需要，**make** 命令能够避免重新生成 **xs.o** 文件。

标志

-c 从指定文件中抽取字符串，并将其放在 **strings** 文件中。
-v 详细方式。指出字符串是何时找到的，或在 **strings** 文件中创建新字符串的时间。
- 读取标准输入。

示例

1. 从 *File.c* 参数表示的 C 源文件中抽取字符串，为某些字符串引用替换为（**&xstr**[*number*]）的格式的表达式:

   ```
   xstr -c File.c
   ```

   xstr 数组的适当声明添加到文件首部。得到的 C 文本放入文件 **x.c** 中，然后进行编译。
2. 要在 **xs.c** 文件中声明公共 xstr 数组空间:

   ```
   xstr
   ```

文件

strings 包含抽取字符串的文件。
x.c 处理过的 C 源程序。

xs.c	xstr 数组定义的 C 源程序。
/tmp/xs*	当 **xstr** 命令不更改 **strings** 文件*时的临时文件*。
/usr/ccs/bin/mkstr	包含可执行文件。
/usr/ccs/bin/mkstr	包含 Berkeley 环境中的可执行文件。

相关信息

mkstr 命令。

xterm 命令

用途

为 X Window System 提供终端仿真器。

注: **xterm** 命令是从麻省理工学院 (MIT) 的 X Window System 版本 11，发行版 6 移植过来的，无功能增强。**xterm** 命令不支持本地化或国际化。对于本地化的或国际化的终端仿真器，用户可以使用 **aixterm** 或 **dtterm** 命令。

语法

xterm [*-ToolkitOption* ...] [*-Option* ...]

描述

xterm 程序是 X Window System 的终端仿真器。它向不能直接使用视窗系统的程序提供 DEC VT102 和 Tektronix 4014 兼容终端。如果基础操作系统支持终端缩放功能，则 **xterm** 程序使用这些工具在窗口缩放时通知视窗系统中运行的程序。

VT102 和 Tektronix 4014 终端都有自己的窗口，因此可以同时在一种视窗系统中编辑文本而在另一种中查看图像。为保持正确的纵横比（高／宽），Tektronix 图像被限制为适合窗口大小且符合 4014 纵横比的最大框。该框位于窗口的左上区域。

虽然两个窗口可同时显示，但是其中一个会被认定为接收键盘输入和终端输出的*活动窗口*。该窗口是包含文本光标的窗口。活动窗口可以通过以下转义序列来选择: VT102 窗口中的 VT Options 菜单和 4014 窗口中的 Tek Options 菜单。

仿真

VT102 仿真相当完备，但不支持平滑滚动、VT52 方式、闪烁字符属性或是双倍宽和双倍大小字符集。与 **xterm** 命令一起使用的 **termcap** 文件条目包括 **xterm**、**vt102**、**vt100** 和 “ansi”，而且 **xterm** 命令自动按这个顺序在 **termcap** 文件中搜索这些条目，并设置 **TERM** 和 **TERMCAP** 环境变量。

很多专用的 **xterm** 功能可以在程序控制下通过一组与标准 VT102 转义序列不同的转义序列进行修改。

Tektronix 4014 仿真也相当优良。它支持 12 位的图像寻址，可以扩大到指窗口大小。它支持四种不同的字体大小和五种不同的行类型。不支持直写或散焦方式。

Tektronix 文本和图像命令由 **xterm** 命令进行内部记录，并可能通过发送 COPY 转义序列（或是通过 Tektronix 菜单，这会在以下各节描述）写入一个文件。该文件的名称将是 **COPY***yy*-*MM*-*dd*.*hh*:*mm*:*ss*，其中，*yy*、*MM*、*dd*、*hh*、*mm* 和 *ss* 是执行复制的年、月、日、小时、分钟和秒（该文件是在启动 **xterm** 命令的目录，或登录 **xterm** 的主目录中创建的)。

其它功能

xterm 命令在指针进入窗口（选中）时自动突出显示文本光标，在指针离开窗口时（未选中）对它取消突出显示。如果窗口是焦点窗口，无论指针定位在何处，文本光标都会突出显示。

在 VT102 方式中，可使用转义序列激活和停用备用屏幕缓冲区，备用屏幕缓冲区的大小与窗口的显示区域大小相同。当激活时，当前屏幕被保存并替换为备用屏幕。对超出窗口顶部的滚动行的保存被禁用，直至恢复通常的屏幕。

xterm 命令的 **termcap** 文件条目允许 **vi** 命令编辑器切换到备用屏幕进行编辑并在退出时恢复屏幕。

在 VT102 方式或 Tektronix 方式中，都有更改窗口名称的转义序列。

选项

xterm 终端仿真器接受所有标准 X Toolkit 命令行选项以及以下选项（如果选项以 a+ 开始而不是 a-，选项恢复为缺省值）：

-help	使 **xterm** 命令打印出描述其选项的消息。
-132	通常，在 80 和 132 列间切换的 VT102 DECCOLM 转义序列被忽略。该选项使 DECCOLM 转义序列可以成为被识别的，而且 xterm 窗口将相应地调整大小。
-ah	表明 **xterm** 命令应该总是突出显示文本光标。缺省情况下，当无焦点或指针离开该窗口时，**xterm** 命令将显示一个中空的文本光标。
+ah	表明 **xterm** 命令应该根据焦点突出显示文本光标。
-b *Number*	以像素为单位指定内边框大小（字符外边框和窗口边框之间的距离）。缺省值是 2。
-cc *CharacterClassRange*:*Value*[**,**...]	设置用于根据文字进行选择的给定范围所表明的类。
-cn	表明换行符不应该在行方式选择中被剪切下来。
+cn	表明换行符应该在行方式选择中被剪切下来。
-cr *Color*	指定文本光标使用的颜色。缺省值是使用与文本前景色相同的颜色。
-cu	表明 **xterm** 命令应该在回避 **more** 程序中的错误，该错误导致不能正确显示如下的行：行宽度与窗口宽度完全相同并跟随一个由制表符开始的行（前导 tab 不显示）。该选项之所以如此命名是因为它起初就被认为是 **curses** 函数光标运动软件包中的一个程序错误。
+cu	表明 **xterm** 不应当回避前面提到的 **more** 程序的功能错误。
-e *Program* [*Arguments*]	指定在 xterm 窗口中运行的程序（及其命令行自变量）。如果 -**T** 或 -**n** 选项都没有在命令行中给定，它也设置窗口标题和图标名作为运行程序的基本名称。 **注：** 这必须是命令行的最后一个选项。
-fb *Font*	指定在显示粗体文本时要使用的字体。该字体和普通字体高度和宽度相同。如果只指定了一种普通字体或是粗体字体，它会作为普通字体使用并通过叠印该字体来生成粗体字体。缺省值是叠印普通字体。
-i	打开**useInsertMode** 资源。
+i	关闭 **useInsertMode** 资源。

-j
表明 **xterm** 命令应该进行跳跃式的滚动。通常情况下，文本一次只滚动一行；该选项允许 **xterm** 命令一次移动多行以便不致于落后太远。强烈推荐使用该选项，因为它使 **xterm** 命令在扫描大量的文本时要快得多。VT100 中启用和禁用平滑滚动的转义序列，就和 VT Options 菜单一样，可以用来打开或关闭该功能。

+j
表明 **xterm** 命令不应该进行跳跃式的滚动。

-ls
表明在 xterm 视窗中启动的 shell 是登录 shell（换句话说，*ArgumentVector* 参数的第一个字符是破折号，用来向 shell 表明它会读取用户的 **.login** 或 **.profile** 文件）。

+ls
表明启动的 shell 不应是登录 shell（换句话说，它是一个普通的子外壳程序）。

-mb
表明 **xterm** 命令在用户输入到靠近行右端时应该响页边距铃提醒。该选项可以用 VT Options 菜单打开或关闭。

+mb
表明不应响页边距铃提醒。

-mc *Milliseconds*
指定多击选择之间的最大时间。

-ms *Color*
指定指针光标使用的颜色。缺省值是使用前景色。

-nb *Number*
指定页边缘铃在离行的右端结束处还有多少个字符数时会响铃（如果启动了页边缘铃）。缺省值是 10。

-rw
表明应当允许逆向回绕。这允许光标从一行的最左列回退到前一行的最右边列。这对编辑长的 shell 命令行很有用，鼓励使用该选项。该选项可以用 VT Options 菜单打开或关闭。

+rw
表明不允许逆向回绕。

-aw
表明应当允许自动回绕。这允许光标在一行最右边的位置而且在输入文本时自动环绕到下一行的开始。

+aw
表明不允许自动回绕。

-s
表明 **xterm** 命令可以异步滚动，这表明在滚动时不必完全保持为最新状态。这允许 **xterm** 命令在网络延迟严重时运行更快，并在通过大型因特网或很多网关运行时十分有用。

+s
表明 **xterm** 命令应该同步滚动。

-sb
表明应该保存一些滚动到窗口顶部以上的行，而且也应该显示滚动条以便可以查看这些行。该选项可以用 VT Options 菜单打开或关闭。

+sb
表明不应显示滚动条。

-sf
表明应该为功能键生成 Sun Function Key 转义码。

+sf
表明应该为功能键生成标准的转义码。

-si
表明到窗口的输出不应自动将屏幕重新定位到滚动区域的底部。该选项可以用 VT Options 菜单打开或关闭。

+si
表明到窗口的输出应使窗口滚动至底部。

-sk
表明在使用滚动条查看文本中的先前行时，如果按下一个键会导致窗口自动重新定位到滚动区域底部的通常位置。

+sk
表明在使用滚动条时按下一个键不会导致窗口重新定位。

-sl *Number*
指定要保存的、滚动到屏幕顶部以上的行数。缺省值是 64。

-t
表明 **xterm** 命令应该以 Tektronix 方式启动，而不是 VT102 方式。可使用 Options 菜单在这两个窗口之间进行切换。

+t
表明 **xterm** 命令应以 VT102 方式启动。

-tm *String*
指定以应与相关功能绑定的字符开头的一系列终端设置关键字，类似于 **stty** 程序。允许的关键字包括：**intr**、**quit**、**erase**、**kill**、**eof**、**eol**、**swtch**、**start**、**stop**、**brk**、**susp**、**dsusp**、**rprnt**、**flush**、**weras** 和 **lnext**。控制字符可以指定为 ^*Character*（例如 ^c 或 ^u），并且 ^? 可用来表示 Delete。

-tn *Name*
指定在 **TERM** 环境变量中设置的终端类型名称。该终端类型必须存在于 **termcap** 数据库中而且应当包含 **li#** 和 **co#** 条目。

-ut 表明 **xterm** 命令不应将记录写入 **/etc/utmp** 系统日志文件。

+ut 表明 **xterm** 命令应该将记录写入 **/etc/utmp** 系统日志文件。

-vb 表明更倾向于使用可视铃，而不是可听的铃。每当接收到 Ctrl+G 序列信号时，窗口会闪烁，而不是响起终端铃声。

+vb 表明不使用可视铃。

-wf 表明 **xterm** 命令在第一次启动子进程时等待窗口映射完毕，以确保初始终端大小设置和环境变量是正确的。捕获随后终端大小的更改是应用程序的职责。

+wf 表明 **xterm** 命令在启动子进程前不等待。

-C 表明窗口接收控制台输出。并非所有系统都支持本选项。要获取控制台输出，必须是控制台设备的所有者，而且必须对它有读和写的许可权。如果在控制台屏幕的 **xdm** 下运行 X-Windows，可能需要会话启动和重新设置程序来更改控制台设备的所有权以便该选项能够起作用。

-S*ccn* 指定在从属方式中使用伪终端名称的最后两个字母和继承的文件描述符号码。该选项是解析的 ``%c%c%d''。这允许将 **xterm** 命令用作现有程序的输入和输出通道，且有时在专用的应用程序中使用。

以下命令行自变量是为兼容旧版本而提供的。由于 X Toolki 提供完成同样任务的标准选项，在下个发行版中也许不支持这些自变量。

%*geom* 指定 Tektronix 视窗的首选大小和位置。这是对指定 ***tekGeometry** 资源的简写。

#*geom* 指定图标窗口的首选位置。这是对指定 ***iconGeometry** 资源的简写。

-T *String* 指定 **xterm** 程序窗口的标题。它等同于 **-title**。

-n *String* 指定 **xterm** 程序窗口的图标名。这是对指定 ***iconName** 资源的简写。注意：这与 Toolkit 选项 **-name** 不同（参阅以下内容）。缺省图标名是应用程序名。

-r 表明反转视频应当通过调换前景和背景颜色来进行模拟。它等同于 **-rv**。

-w *Number* 指定环绕窗口的边框的宽度（以像素计算）。它等同于 **-borderwidth** 或 **-bw**。

以下标准 X Toolkit 命令行参数通常与 **xterm** 命令一起使用：

-bg *Color* 指定窗口背景使用的颜色。缺省值是白色。

-bd *Color* 指定窗口边框使用的颜色。缺省值为黑色。

-bw *Number* 指定环绕窗口边框的宽度（以像素计算）。

-fg *Color* 指定用于显示文本的颜色。缺省值为黑色。

-fn *Font* 指定用于显示普通文本的字体。缺省值是 fixed。

-name *Name* 指定获取资源的应用程序名，而不是缺省的可执行文件名。*Name* 参数不应包含 .（点）或是 *（星号）字符。

-title *String* 指定窗口标题字符串，该标题会在用户选择时由视窗管理器显示出来。缺省标题是在 **-e** 选项后指定的命令行（如果有的话）；否则，就是应用程序名。

-rv 表明反转视频应当通过调换前景和背景颜色来进行模拟。

-geometry *Geometry* 指定 VT102 视窗的首选位置和大小；请参阅 **X** 命令。

-display *Display* 指定要联系的 X 服务器；请参阅 **X** 命令。

-xrm *ResourceString* 指定要使用的资源字符串。这对设置没有单独命令行选项的资源特别有用。

-iconic 表明 **xterm** 命令应该要求视窗管理器作为一个图标启动，而不是作为普通窗口启动。

资源

本程序能接受并解释所有核心 X Toolkit 资源名、类以及:

iconGeometry（类 **IconGeometry**）
指定在应用程序图标化时的首选位置和大小。没有必要让所有视窗管理器遵守本设置。

termName（类 **TermName**）
指定要在 **TERM** 环境变量中设置的终端类型名称。

title（类 **Title**）
指定在显示本应用程序时窗口管理器使用的字符串。

ttyModes（类**TttyModes**）
指定一个字符串，该字符串包含终端设置的关键字和它们可能会绑定的字符。允许的关键字包括: **intr**、**quit**、**erase**、**kill**、**eof**、**eol**、**swtch**、**start**、**stop**、**brk**、**susp**、**dsusp**、**rprnt**、**flush**、**weras** 和 **lnext**。控制字符可以指定为 ^*Character*（例如 ^c 或 ^u），并且 ^? 可用来表示 Delete。这对重设缺省终端设置而不用在每次 **xterm** 窗口启动时都运行 **stty** 程序十分有用。

useInsertMode（类 **useInsertMode**）
通过添加相应的条目到 **TERMCAP** 环境变量强制使用插入方式。如果系统条件冒损坏，该选项十分有用。缺省值是 **false**。

utmpInhibit（类 **UtmpInhibit**）
指定 **xterm** 是否应该尝试在 **/etc/utmp** 中记录用户终端。

sunFunctionKeys（类 **SunFunctionKeys**）
指定是否应当为功能键（而不是标准转义序列）生成 Sun Function Key 转义码。

waitForMap（类 **WaitForMap**）
指定 **xterm** 命令在启动子进程前是否应该等待初始化视窗映射完毕。缺省值为 False。

以下资源是作为 **vt100** 小窗口（类 **VT100**）的一部分指定的:

allowSendEvents（类 **AllowSendEvents**）
指定是否应当解释或废弃合成键和按键事件（使用 X 协议 **SendEvent** 请求生成这些事件）。缺省值是 False，表示应该废弃这些键和事件。注意: 允许这些事件会造成很大的安全性漏洞。

alwaysHighlight（类 **AlwaysHighlight**）
指定 **xterm** 是否应该始终突出显示文本光标。缺省情况下，无论是指针移动到窗口外还是窗口丧失输入焦点，都显示一个中空的文本光标。

appcursorDefault（类 **AppcursorDefault**）
如果值为 True，光标键最初为应用程序方式。缺省值为 False。

appkeypadDefault（类 **AppkeypadDefault**）
如果值为 True，小键盘键最初为应用程序方式。缺省值为 False。

autoWrap（类**AutoWrap**）
指定是否应当启用自动回绕。缺省值为 True。

bellSuppressTime（类 **BellSuppressTime**）
指定发送铃声命令后的毫秒数，在这期间其它的铃声都将会被禁止。缺省值是 200。如果设置为非零值，其它铃声将被禁止直到服务器报告第一个铃声的处理已经完成为止；该功能对可视铃特别有用。

boldFont（类 **BoldFont**）
指定要使用而不是进行叠印的粗体字体的名称。

c132（类 **C132**）
指定是否应当使用 VT102 DECCOLM 转义序列。缺省值为 False。

charClass（类**CharClass**）
指定以下格式的字符类绑定的以逗号分隔的列表：[*low*-]*high*:*value*。它们用来确定在剪切和粘贴时应该对哪些组字符同等对待。请参阅第 187 页的『字符类』。

curses（类 **Curses**）
指定是否应该回避 **curses** 功能中最后一列程序错误。缺省值为 False。

cutNewline（类 **cutNewline**）
如果为 **false**，三次单击选择一行时不包括行尾的换行符。如果为 **true**，则同时选中换行符。缺省值是 **true**。

cutToBeginningofLines（类 **CutToBeginningOfLine**）
如果为 **false**，三次单击选择一行时仅选中从当前单词向前的部分。如果为 **true**，则选择整行内容。缺省值是 **true**。

background（类 **Background**）
指定窗口背景使用的颜色。缺省值是白色。

foreground（类 **Foreground**）
指定用于显示窗口中文本的颜色。设置类名而非实例名是使通常以文本颜色显示的内容更改颜色的一种简单途径。缺省值为黑色。

cursorColor（类 **Foreground**）
指定文本光标使用的颜色。缺省值为黑色。

eightBitInput（类 **EightBitInput**）
如果值为 True，在八位输入打开的情况下，键盘的元字符输入作为单个字符出现。如果为 False，元字符转换为一个 2 字符的序列，字符本身之前插入 **ESC**。缺省值为 True。

eightBitOutput（类 **EightBitOutput**）
指定从主机发送的 8 位字符是否应该按原样接受或是在打印时予以删节。缺省值为 True。

font（类 **Font**）
指定常规字体的名称。缺省值是 fixed。

font1（类 **Font1**）
指定第一个备用字体的名称。

font2（类 **Font2**）
指定第二个备用字体的名称。

font3（类 **Font3**）
指定第三个备用字体的名称。

font4（类 **Font4**）
指定第四个备用字体的名称。

font5（类 **Font5**）
指定第五个备用字体的名称。

font6（类 **Font6**）
指定第六个备用字体的名称。

geometry（类 **Geometry**）
指定 VT102 视窗的首选位置和大小。

hpLowerleftBugCompat（类 **hpLowerleftBugCompat**）
指定是否应当回避 **xdb** 中的程序错误，它忽略条件冒并会一直发送 ESC F 来移动到左下角。**true** 导致 **xterm** 将 ESC F 解释为移动到屏幕左下角的请求。缺省值是 **false**。

internalBorder（类 **BorderWidth**）
指定字符与窗口边框之间的像素值。缺省值是 2。

jumpScroll（类 **JumpScroll**）
指定是否应当使用跳跃滚动。缺省值为 True。

loginShell（类 **LoginShell**）
指定在窗口中运行的 shell 是否应当作为登录 shell 启动。缺省值为 False。

marginBell（类 **MarginBell**）
指定在用户输入靠近右页边距时是否应当响起铃声。缺省值为 False。

multiClickTime（类 **MultiClickTime**）
指定多击选择事件之间的最大时间（以毫秒计算）。缺省值是 250 毫秒。

multiScroll（类 **MultiScroll**）
指定是否应当进行异步滚动。缺省值为 False。

nMarginBell（类 **Column**）
指定从右页边距开始的字符数，在该处页边距铃会响铃（如果该选项已启用）。

pointerColor （类 **Foreground**）
指定指针的前景色。缺省值是 **XtDefaultForeground**。

pointerColorBackground（类 **Background**）
指定指针的背景色。缺省值是 **XtDefaultBackground**。

pointerShape（类 **Cursor**）
指定指针形状的名称。缺省值是 **xterm**。

resizeGravity（类 **ResizeGravity**）
在调整窗口大小到更高或更短时会影响其行为。**NorthWest** 指定屏幕文本的顶行保持固定。如果窗口变低，请从底部删除行；如果窗口变高，请在底部添加空白行。

这与 MIT 版本 X11R4 的行为相兼容。**SouthWest**（缺省值）指定屏幕文本的底行保持固定。如果窗口变高，其它的已保存的行会向下滚动到屏幕上；如果视窗变低，文本行会滚出屏幕顶部，顶部已保存的行会被删除。

reverseVideo（类 **ReverseVideo**）
指定是否应当模拟反转视频。缺省值为 False。

reverseWrap（类 **ReverseWrap**）
指定是否应当启用逆向回绕。缺省值为 False。

saveLines（类 **SaveLines**）
指定在滚动条打开时要保存的、超出屏幕顶端的行数。缺省值是 64。

scrollBar（类 **ScrollBar**）
指定是否应当显示滚动条。缺省值为 False。

scrollTtyOutput（类 **ScrollCond**）
指定到终端输出是否应当使滚动条自动滚动到滚动区域底部。缺省值为 True。

scrollKey（类 **ScrollCond**）
指定到按键操作是否应当使滚动条自动滚动到滚动区域底部。缺省值为 False。

scrollLines（类 **ScrollLines**）
指定 **scroll-back** 和 **scroll-forw** 操作应该作为缺省值使用的行数。缺省值是 1。

signalInhibit（类 **SignalInhibit**）
指定在 Main Options 菜单中发送信号到 **xterm** 的条目是否应该被禁止。缺省值为 False。

tekGeometry（类 **Geometry**）
指定 Tektronix 视窗的首选位置和大小。

tekInhibit（类 **TekInhibit**）
指定输入 Tektronix 方式的转义序列是否应当被忽略。缺省值为 False。

tekSmall（类 **TekSmall**）
指定如果没有给定明确的几何图形，Tektronix 方式下的窗口是否应该以最小大小启动。在小屏幕显示器上运行 **xterm** 命令时这十分有用。缺省值为 False。

tekStartup（类 **TekStartup**）
指定 **xterm** 是否应该以 Tektronix 方式启动。缺省值为 False。

titeInhibit（类 **TiteInhibit**）
指定 **xterm** 是否应当从 **TERMCAP** 字符串除去 **ti** 和 **te** 条件冒文件条目（在很多面向屏幕的程序启动时，用来在备用屏幕之间切换）。如果设置此选项，**xterm** 命令也会忽略切换到备用屏幕的转义序列。

translations（类 **Translations**）
指定菜单、选项、已编程字符串等等的键和按钮绑定。请参阅“操作”。

visualBell（类 **VisualBell**）
指定当接收到 Ctrl+C 组合键信号时是否应当用可视铃（闪烁）来替代音效铃。缺省值为 False。

以下资源是作为 **tek4014** 小窗口（类 **Tek4014**）的一部分来指定的:

width（类 **Width**）
指定 Tektronix 窗口的宽度（以像素值计算）。

height（类 **Height**）
指定 Tektronix 窗口的高度（以像素值计算）。

fontLarge（类 **Font**）
指定在 Tektronix 视窗系统中使用的大字体。

font2（类 **Font**）
指定在 Tektronix 视窗系统中使用的 2 号字体。

font3（类 **Font**）
指定在 Tektronix 视窗系统中使用的 3 号字体。

fontSmall（类 **Font**）
指定在 Tektronix 视窗系统中使用的小字体。

initialFont（类 **InitialFont**）
指定四种 Tektronix 字体中最初使用哪种字体。值和 **set-tek-text** 操作的值相同。缺省值为 large。

ginTerminator（类 **GinTerminator**）
指定在 GIN 报告或状态报告后应当跟随哪些字符。有可能是，“none”，它发送非终止字符；CRonly，它发送 CR；和 CR&EOT，它同时发送 CR 和 EOT。缺省值为 none。

可能为各种不同菜单指定的资源在 **Athena SimpleMenu** 小窗口的文档中都有描述。以下是每个菜单中各条目的类和名称的列表。

mainMenu 包含以下条目:

securekbd（类 **SmeBSB**）
调用 **secure()** 操作。

allowsends（类 **SmeBSB**）
调用 **allow-send-events(toggle)** 操作。

redraw（类 **SmeBSB**）
调用 **redraw()** 操作。

line1（类 **SmeLine**）
这是分隔符。

suspend（类 **SmeBSB**） 在支持作业控制的系统中调用 **send-signal(tstp)** 操作。
continue（类 **SmeBSB**） 在支持作业控制的系统中调用 **send-signal(cont)** 操作。
interrupt（类 **SmeBSB**） 调用 **send-signal(int)** 操作。
hangup（类 **SmeBSB**） 调用 **send-signal(hup)** 操作。
terminate（类 **SmeBSB**） 调用 **send-signal(term)** 操作。
kill（类 **SmeBSB**） 调用 **send-signal(kill)** 操作。
line2（类 **SmeLine**） 这是分隔符。
quit（类 **SmeBSB**） 调用 **quit()** 操作。

vtMenu 包含以下条目:

scrollbar（类 **SmeBSB**） 调用 **set-scrollbar(toggle)** 操作。
jumpscroll（类 **SmeBSB**） 调用 **set-jumpscroll(toggle)** 操作。
reversevideo（类 **SmeBSB**） 调用 **set-reverse-video(toggle)** 操作。
autowrap（类 **SmeBSB**） 调用 **set-autowrap(toggle)** 操作。
reversewrap（类 **SmeBSB**） 调用 **set-reversewrap(toggle)** 操作。
autolinefeed（类 **SmeBSB**） 调用 **set-autolinefeed(toggle)** 操作。
appcursor（类 **SmeBSB**） 调用 **set-appcursor(toggle)** 操作。
appkeypad（类 **SmeBSB**） 调用 **set-appkeypad(toggle)** 操作。
scrollkey（类 **SmeBSB**） 调用 **set-scroll-on-key(toggle)** 操作。
scrollttyoutput（类 **SmeBSB**） 调用 **set-scroll-on-tty-output(toggle)** 操作。
allow132（类 **SmeBSB**） 调用 **set-allow132(toggle)** 操作。
cursesemul（类 **SmeBSB**） 调用 **set-cursesemul(toggle)** 操作。
visualbell（类 **SmeBSB**） 调用 **set-visualbell(toggle)** 操作。
marginbell（类 **SmeBSB**） 调用 **set-marginbell(toggle)** 操作。
altscreen（类 **SmeBSB**） 该条目当前被禁用。
line1（类 **SmeLine**） 这是分隔符。
softreset（类 **SmeBSB**） 调用 **soft-reset()** 操作。
hardreset（类 **SmeBSB**） 调用 **hard-reset()** 操作。
clearsavedlines（类 **SmeBSB**） 调用 **clear-saved-lines()** 操作。
line2（类 **SmeLine**） 这是分隔符。
tekshow（类 **SmeBSB**） 调用 **set-visibility(tek,toggle)** 操作。
tekmode（类 **SmeBSB**） 调用 **set-terminal-type(tek)** 操作。
vthide（类 **SmeBSB**） 调用 **set-visibility(vt,off)** 操作。

fontMenu 包含以下条目:

fontdefault（类 **SmeBSB**） 调用 **set-vt-font(d)** 操作。
font1（类 **SmeBSB**） 调用 **set-vt-font(1)** 操作。
font2（类 **SmeBSB**） 调用 **set-vt-font(2)** 操作。
font3（类 **SmeBSB**） 调用 **set-vt-font(3)** 操作。
font4（类 **SmeBSB**） 调用 **set-vt-font(4)** 操作。
font5（类 **SmeBSB**） 调用 **set-vt-font(5)** 操作。
font6（类 **SmeBSB**） 调用 **set-vt-font(6)** 操作。
fontescape（类 **SmeBSB**） 调用 **set-vt-font(e)** 操作。
fontsel（类 **SmeBSB**） 调用 **set-vt-font(s)** 操作。

tekMenu 包含以下条目:

tektextlarge（类 **SmeBSB**） 调用 **set-tek-text(l)** 操作。
tektext2（类 **SmeBSB**） 调用 **set-tek-text(2)** 操作。
tektext3（类 **SmeBSB**） 调用 **set-tek-text(3)** 操作。
tektextsmall（类 **SmeBSB**） 调用 **set-tek-text(s)** 操作。

line1（类 **SmeLine**） 这是分隔符。
tekpage（类 **SmeBSB**） 调用 **tek-page()** 操作。
tekreset（类 **SmeBSB**） 调用 **tek-reset()** 操作。
tekcopy（类 **SmeBSB**） 调用 **tek-copy()** 操作。
line2（类 **SmeLine**） 这是分隔符。
vtshow（类 **SmeBSB**） 调用 **set-visibility(vt,toggle)** 操作。
vtmode（类 **SmeBSB**） 调用 **set-terminal-type(vt)** 操作。
tekhide（类 **SmeBSB**） 调用 **set-visibility(tek,toggle)** 操作。

当为 **Athena Scrollbar** 小窗口指定资源时以下资源十分有用:

thickness（类 **Thickness**） 指定滚动条的宽度（以像素值计算）。
background（类 **Background**） 指定滚动条背景使用的颜色。
foreground（类 **Foreground**） 指定滚动条前景使用的颜色。滚动条的*滚动块*是一个简单的棋盘模式，其前景色和背景色可以使用备用像素。

指针用法

创建 VT102 窗口后，使用 **xterm** 命令可以选择文本并将其复制到相同或不同的窗口。

在不带修改量的情况下使用指针按钮，以及与 Shift 键一起使用时，会调用选择功能。对键和按钮的功能分配可以通过对资源数据库的更改来实现。

指针按钮 1（通常为左按钮）用于将文本保存到剪切缓冲区。将光标移动到文本开头，然后按住按钮并将光标拖动到区域结束处并释放按钮。选中的文本被突出显示并被保存在全局剪切缓冲区中，在按钮释放时进行 PRIMARY 选择（主选择）。

双击按文字进行选择，连击三次按行进行选择，四次连击返回到字符。多击是由释放按钮到按下按钮的时间量决定的，因此可以在选择中更改选择单位。如果键或按钮绑定指定要进行的 X 选择，**xterm** 命令将会使选中的文本保持为突出显示，只要它是选择主体。

指针按钮 2（通常为中间按钮）从 PRIMARY 选择（如果有，否则从剪切缓冲区）中“输入”（粘贴）文本，将文本作为键盘输入插入。

指针按钮 3（通常为右按钮）扩展当前选择。如果按下时靠近选择区的右边比左边近，它扩展或收缩选择区的右边界。如果收缩选择范围穿过了选择区的左边界， **xterm** 命令认定确实指的是左边界并恢复原选择，然后扩展或收缩选择区的左边界。

反之亦然：如果按下时靠近左边边界比右边界近，它扩展／收缩选择区的左边界。如果收缩选择范围超出了选择区的右边界，**xterm** 命令认定确实指的是右边界并恢复原选择，然后扩展/收缩选择区的左边界。扩展上次选择或扩展执行的选择单元方式启动；可以多击按钮来循环使用选择方式。

例如，在剪切和粘贴一块块文本而不包含后续换行符时，可以从不同窗口的多个位置获取文本，然后组成 shell 的一个命令，或从程序获取输出并将其插入最喜欢的编辑器。因为剪切缓冲区是在不同应用程序之间全局共享的，所以可以把它看成是内容已知“文件”。终端仿真器和其它文本程序会把它看成是一个文本文件；换句话说，文本由换行符定界。

滚动区域显示相对于实际保存的文本数量，当前显示在窗口中（突出显示）的位置和文本数量。当更多文本保存（直到最大值）时，突出显示区域的大小会减少。

在滚动区域用指针单击按钮 1 将邻近行移动到显示窗口的顶端。

单击按钮 3 将显示窗口的顶行向下移动到指针位置。

单击按钮 2 将显示移动到与滚动条指针位置相对应的已保存文本中。

和 VT102 视窗不同，Tektronix 视窗中不允许文本复制。它确实允许使用 Tektronix GIN 方式，而且在该方式下光标会从箭头变为十字形。按任意键将发送该键和十字光标的当前坐标。按按钮 1、2、3 会分别返回字母 l、m、r。

如果在指针按钮按下时按下 Shift 键，则发送相应的大写字母。为将指针按钮和键分开，设置了字符的高位（但这一位经常被删去，除非终端方式是 RAW；请参阅 **tty** 命令获取详细信息）。

菜单

xterm 命令有四个菜单，名称分别为 mainMenu、vtMenu、fontMenu 和 tekMenu。每一个菜单在正确的按键和按钮组合下展开。大部分菜单都通过水平线分隔而划分为两个部分。菜单顶部包含可以改变的各种方式。在当前活动的方式的旁边显示复选标记。选择其中一个方式会切换其状态。菜单底部列出命令条目；选择其中之一将执行显示的功能。

xterm 菜单在窗口中的控制键和指针按钮 1 按下时打开。mainMenu 包含适用于 VT102 和 Tektronix 视窗的项。在不安全的环境中输入密码或是别的敏感数据时请使用 **Secure Keyboard** 方式。

菜单命令部分中值得注意的条目是：**Continue**、**Suspend**、**Interrupt**、**Hangup**、**Terminate** 和 **Kill**，它们分别发送 **SIGCONT**、**SIGTSTP**、**SIGINT**、**SIGHUP**、**SIGTERM** 和 **SIGKILL** 信号到在 **xterm** 下运行的进程的进程组中（通常就是 shell）。如果用户偶然按下 Ctrl+Z 暂挂了进程，**Continue** 功能会非常有用。

vtMenu 在 VT102 仿真中设置不同方式，并在 VT102 窗口中按下控制键和指针按钮 2 时被打开。在该菜单的命令部分，软复位条目会重新设置滚动区域。当某些程序使滚动区域保持错误设置时（通常是在使用 VMS 或 TOPS-20 时有问题），软复位十分方便。

在 **xterm** 命令处理完命令行选项后，完全复位条目会立即清除屏幕，将制表符复位为每 8 列，并将终端方式（比如换行和平滑滚动）重新设置为它们的初始状态。

fontMenu 设置在 VT102 视窗中使用的字体。除了由资源设置的缺省字体和若干备用字体，菜单还提供“设置字体”转义序列上次指定的字体（请参阅“控制序列”）和当前选择作为字体名（如果有 PRIMARY 选项）。

tekMenu 在 Tektronix 仿真中设置不同方式，在 Tektronix 窗口中按下控制键和指针按钮 2 时可将其打开。当前字体大小在菜单的 Modes 部分检入。命令部分的 **PAGE** 条目会清除 Tektronix 窗口。

安全性

X 视窗环境在它们的安全性级别方面有所不同。MIT 服务器，在 **xdm** 下运行，可以使用 *magic cookie* 授权模式，该模式可以对很多人提供合理的安全等级。如果服务器只使用基于主机的机制来控制对服务器的访问（参阅 **xhost** 命令），而且如果对一个主机启用访问且也允许其他用户在相同的主机上运行客户机，那么很可能有人可以运行使用 X 协议的基本服务窥探到您的活动，并有可能捕获到从键盘输入的所有内容的副本。

这在要输入密码或其它敏感数据时要特别注意。该问题的最佳解决方案是用一种更好的授权机制而不是基于主机的控制，但在 **xterm** 命令中有一种保护键盘输入的简单机制。

xterm 菜单包含 **Secure Keyboard** 选项，启用时，能够确保所有的键盘输入仅定向到 **xterm** 命令（使用 **GrabKeyboard** 协议请求）。当应用程序提示输入密码（或其它敏感数据）时，可以用菜单启用 **Secure Keyboard**，输入数据，然后再次通过使用菜单禁用 **Secure Keyboard**。

一次只能有一个 X 客户机可以保护键盘，因此当尝试启用 **Secure Keyboard** 时，它可能会失败。在这种情况下，会响铃提示。如果 **Secure Keyboard** 成功启动，前景和背景颜色会交换（就像在“方式”菜单中选择了 **反转视频** 条目一样）；当退出安全方式时，颜色将再次交换。如果颜色*没有*交换，这就*十分*有可能正在受到欺骗。

如果正在运行的应用程序在询问密码前显示提示符，最安全的方法就是：在提示符显示*之前*进入安全方式，而且确保提示符显示正确（以新的颜色），以将受欺骗的可能性降到最低。也可以再次打开菜单并确保该条目旁边有复选标记显示。

如果在 **Secure Keyboard** 方式中 xterm 窗口已图标化（或是已取消映射），或者如果启动一个再生视窗管理器（它在窗口周围摆放标题栏或其它装饰品），则 **Secure Keyboard** 方式会自动禁用。（这是 X 协议的一个不容易攻破的功能。）当发生上述情况时，前景和背景颜色将会交换而且会响铃警告。

字符类

快速连击鼠标中间按钮两次会导致所有同类字符（比如字母、空白、标点）都被选中。因为不同的人对那些应该被选中的内容有不同的偏好（比如，文件名是否应该全被选中还是只是单独的子名），缺省映射可以通过使用 **charClass**（类 **CharClass**）资源来覆盖。

该资源是一系列以逗号分隔的 *range*:*value* 对。*range* 是在 0 到 127 之间单个数字或 *low-high*（表示从低到高），对应于要设置字符的 ASCII 代码。*value* 是任意的，虽然缺省表使用在集合中出现的第一个字符的字符数。

缺省表是:

```
static int charClass[128] = {
/* NUL  SOH  STX  ETX  EOT  ENQ  ACK  BEL */
    32,   1,   1,   1,   1,   1,   1,   1,
/*  BS   HT   NL   VT   NP   CR   SO   SI */
     1,  32,   1,   1,   1,   1,   1,   1,
/* DLE  DC1  DC2  DC3  DC4  NAK  SYN  ETB */
     1,   1,   1,   1,   1,   1,   1,   1,
/* CAN   EM  SUB  ESC   FS   GS   RS   US */
     1,   1,   1,   1,   1,   1,   1,   1,
/*  SP    !    "    #    $    %    &    ' */
    32,  33,  34,  35,  36,  37,  38,  39,
/*   (    )    *    +    ,    -    .    / */
    40,  41,  42,  43,  44,  45,  46,  47,
/*   0    1    2    3    4    5    6    7 */
    48,  48,  48,  48,  48,  48,  48,  48,
/*   8    9    :    ;    <    =    >    ? */
    48,  48,  58,  59,  60,  61,  62,  63,
/*   @    A    B    C    D    E    F    G */
    64,  48,  48,  48,  48,  48,  48,  48,
/*   H    I    J    K    L    M    N    O */
    48,  48,  48,  48,  48,  48,  48,  48,
/*   P    Q    R    S    T    U    V    W */
    48,  48,  48,  48,  48,  48,  48,  48,
/*   X    Y    Z    [    \    ]    ^    _ */
    48,  48,  48,  91,  92,  93,  94,  48,
/*   `    a    b    c    d    e    f    g */
    96,  48,  48,  48,  48,  48,  48,  48,
/*   h    i    j    k    l    m    n    o */
    48,  48,  48,  48,  48,  48,  48,  48,
/*   p    q    r    s    t    u    v    w */
    48,  48,  48,  48,  48,  48,  48,  48,
/*   x    y    z    {    |    }    ~  DEL */
    48,  48,  48, 123, 124, 125, 126,   1};
```

比如，字符串 33:48,37:48,45-47:48,64:48 表示感叹号、百分号、破折号、句点、斜杠和 &字符都应该与字符和数字相同对待。这对剪切和粘贴电子邮件地址和文件名很有用。

操作

可以通过更改 **vt100** 或 **tek4014** 小窗口的转换将键（或组合键）重新绑定到任意输入字符串。更改事件的转换而不是键和按钮事件并非是预期的操作，这会导致不可预测的行为。以下操作是为在 vt100 或 tek4014 转换资源中使用而提供的:

bell([*Percent*]**)**	将基本音量提高／降低指定的百分比来提示键盘铃声。
ignore()	忽略该事件但检查特殊指针位置转义序列。
insert()	插入与按下的键相关联的字符或字符串。
insert-seven-bit()	等同于 **insert()**。
insert-eight-bit()	插入一个与按下的键相关联的 8 位（元）版本的字符或字符串。确切的操作依靠于 **eightBitInput** 资源的值。
insert-selection(*SourceName* [, ...]**)**	插入在选择区或剪切缓冲区找到的由 *SourceName* 参数指示的字符串。按给定的顺序检查源（区分大小写）直到找到一个目标。通常使用的选择区包括 PRIMARY、SECONDARY 和 CLIPBOARD。剪切缓冲区通常命名为从 CUT_BUFFER0 到 CUT_BUFFER7 的序列。
keymap(*Name***)**	动态定义一个新的转换表，其资源名是 *Name* 且后缀是 *Keymap*（区分大小写）。名称 None 将恢复原始的转换表。
popup-menu(*MenuName***)**	显示指定的弹出菜单。有效的名称（区分大小写）包括 mainMenu、vtMenu、fontMenu 和 tekMenu。
secure()	切换 **Secure Keyboard** 方式（在名为“安全性”的一节中有该方式的描述），而且是从 mainMenu 中 **securekbd** 条目中调用的。
select-start()	在当前指针位置开始文本选择。请参阅标题为“指针用法”的一节以获取如何进行选择的详细信息。
select-extend()	跟踪指针并扩展该选择。仅将其绑定到 Motion 事件。
select-end(*DestName* [, ...]**)**	将当前选择的文本放入 *DestName* 指定的所有选择区或剪切缓冲区。
select-cursor-start()	类似于 **select-start**，除了它是从当前文本光标位置开始选择。
select-cursor-end(*DestName* [, ...]**)**	类似于 **select-end**，除了它应与 **select-cursor-start** 一起使用。
set-vt-font(*d/1/2/3/4/5/6/e/s* [,*NormalFont* [, *BoldFont*]]**)**	设置当前 VT102 视窗使用的字体。第一个自变量为单个字符，它指定要使用的字体: *d* 或 *D* 表示缺省值字体（在 **xterm** 命令启动时最初使用的字体）， *1* 到 *6* 表示由 *font1* 到 *font6* 资源指定的字体， *e* 或 *E* 表示通过转义码设置的普通和粗体字体（或是分别指定为由第二个和第三个操作自变量）， *s* 或 *S* 表示由第二个操作自变量指定的字体选择（由程序选择，比如 **xfontsel** 程序）。
start-extend()	类似于 **select-start**，除了选择区扩展到当前指针位置。
start-cursor-extend()	类似于 **select-extend**，除了选择区扩展到当前文本光标位置。

string(*String***)**
插入指定的文本字符串，就好像它是被输入的一样。如果字符串包含空白或非字母数字字符，则必须使用引号。如果字符串自变量以字符“0x”开头，则它会解释为一个十六进制字符常量。

scroll-back(*Count* [,*Units*])
回滚文本窗口，以便先前滚动出屏幕顶部的文本可以显示出来。*Count* 自变量表示要滚动的 *Units*（可能是 *page*、*halfpage*、*pixel* 或 *line*）的数量。

scroll-forw(*Count* [,*Units*])
滚动类似于 **scroll-back**，除了它是向另一方向滚动。

allow-send-events(*On/Off/Toggle***)**
设置或切换 **allowSendEvents** 资源而且也可以通过 mainMenu 中的 **allowsends** 条目调用。

redraw()
重绘窗口，也可以通过 mainMenu 中的 **redraw** 条目调用。

send-signal(*SigName***)**
发送名为 *SigName* 的信号到**xterm** 子进程（由 **-e** 命令行选项指定的 shell 或程序），并可以通过 mainMenu 中的 **suspend**、**continue**、**interrupt**、**hangup**、**terminate** 和 **kill** 等条目调用。允许的信号名为（不区分大小写）：

tstp （如果操作系统支持），

suspend （等同于 **tstp**),

cont （如果操作系统支持），

int,

hup,

term,

quit,

alrm,

alarm（等同于 **alrm**），和

kill。

quit()
发送 **SIGHUP** 到子程序并退出。它也可以通过 mainMenu 中的 **quit** 条目调用。

set-scrollbar(*On/Off/Toggle***)**
切换 **scrollbar** 资源，也可以通过 vtMenu 中的 **scrollbar** 条目调用。

set-jumpscroll(*On/Off/Toggle***)**
切换 **jumpscroll** 资源，也可以通过 vtMenu 的 **jumpscroll** 条目调用。

set-reverse-video(*On/Off/Toggle***)**
切换 **reverseVideo** 资源，也可以通过 vtMenu 中的 **reversevideo** 条目调用。

set-autowrap(*On/Off/Toggle***)**
切换长输入行的自动换行，并可通过 vtMenu 中的 **autowrap** 条目调用。

set-reversewrap(*On/Off/Toggle***)**
切换 **reverseWrap** 资源，也可以通过 vtMenu 的 **reversewrap** 条目调用。

set-autolinefeed(*On/Off/Toggle***)**
切换换行的自动插入方式，并可以通过 vtMenu 中的 **autolinefeed** 条目调用。

set-appcursor(*On/Off/Toggle***)**
切换 Application Cursor Key 方式的处理，并可以通过 vtMenu 中的 **appcursor** 条目调用。

set-appkeypad(*On/Off/Toggle***)**
切换 Application Keypad 方式的处理，并可以通过 vtMenu 中的 **appkeypad** 条目调用。

set-scroll-on-key(*On/Off/Toggle***)**
切换 **scrollKey** 资源，也可以从 vtMenu 中的 **scrollkey** 条目调用。

set-scroll-on-tty-output(*On/Off/Toggle***)**
切换 **scrollTtyOutput** 资源，也可以从 vtMenu 中的 **scrollttyoutput** 条目调用。

set-allow132(*On/Off/Toggle***)**
切换 **c132** 资源，也可以从 vtMenu 中的 **allow132** 条目调用。

set-cursesemul(*On/Off/Toggle*) 切换 **curses** 资源，也可以从 vtMenu 中的 **cursesemul** 条目调用。

set-visual-bell(*On/Off/Toggle*) 切换 **visualBell** 资源，也可以通过 vtMenu 中的 **visualbell** 条目调用。

set-marginbell(*On/Off/Toggle*) 切换 **marginBell** 资源，而且也可以通过 vtMenu 中的 **marginbel** 条目调用。

set-altscreen(*On/Off/Toggle*) 在备用屏幕和当前屏幕之间切换。

soft-reset() 重新设置滚动区域，而且也可以从 vtMenu 中的 **softreset** 条目调用。

hard-reset() 重新设置滚动区域、标记、窗口大小和光标键并清除屏幕。它也可以从 vtMenu 中的 **hardreset** 条目调用。

clear-saved-lines() 执行 **hard-reset**（参看前面的条目），而且也清除超出屏幕顶部的已保存行的历史记录。它也可以从 vtMenu 中的 **clearsavedlines** 条目调用。

set-terminal-type(*Type*) 按照 *Type* 字符串，直接输出到 vt 或 tek 窗口。也可以从 vtMenu 中的 **tekmode** 条目和 tekMenu 中的 **vtmode** 条目调用。

set-visibility(*vt/tek*, *On/Off/Toggle*) 控制 vt 或 tek 窗口是否可见。它也可以从 vtMenu 中的 **tekshow** 和 **vthide** 条目和 tekMenu 中的 **vtshow** 和 **tekhide** 条目调用。

set-tek-text(*large/2/3/small*) 根据自变量，将 Tektronix 视窗中使用的字体设置为 **tektextlarge**、**tektext2**、**tektext3** 和 **tektextsmall** 资源的值。它也可通过 tekMenu 中与资源同名的条目调用。

tek-page() 清除 Tektronix 窗口，而且也可以通过 tekMenu 中的 **tekpage** 条目调用。

tek-reset() 复位 Tektronix 窗口，而且也可以通过 tekMenu 中的 **tekreset** 条目调用。

tek-copy() 将用于生成当前窗口内容的转义码复制到当前目录中一个名称以 **COPY** 开头的文件。它也可以从 tekMenu 中的 **tekcopy** 条目调用。

visual-bell() 快速刷新窗口。

Tektronix 视窗也具有以下操作:

gin-press(*l/L/m/M/r/R*) 发送指示的图像输入代码。

在 VT102 视窗中缺省绑定是:

```
Shift <KeyPress> Prior:     scroll-back(1,halfpage) \n\
Shift <KeyPress> Next:      scroll-forw(1,halfpage) \n\
Shift <KeyPress> Select:    select-cursor-start \
                            select-cursor-end(PRIMARY,
                            CUT_BUFFER0) \n\
Shift <KeyPress> Insert:    insert-selection(PRIMARY,
                            CUT_BUFFER0) \n\
~Meta<KeyPress>:            insert-seven-bit \n\
Meta<KeyPress>:             insert-eight-bit \n\
!Ctrl <Btn1Down>:           popup-menu(mainMenu) \n\
!Lock Ctrl <Btn1Down>:      popup-menu(mainMenu) \n\
~Meta <Btn1Down>:           select-start \n\
~Meta <Btn1Motion>:         select-extend \n\
!Ctrl <Btn2Down>:           popup-menu(vtMenu) \n\
!Lock Ctrl <Btn2Down>:      popup-menu(vtMenu) \n\
~Ctrl ~Meta <Btn2Down>:     ignore \n\
~Ctrl ~Meta <Btn2Up>:       insert-selection(PRIMARY,
                            CUT_BUFFER0) \n\
```

```
!Ctrl <Btn3Down>:           popup-menu(fontMenu) \n\
!Lock Ctrl <Btn3Down>:      popup-menu(fontMenu) \n\
~Ctrl ~Meta <Btn3Down>:     start-extend \n\
~Meta <Btn3Motion>:         select-extend \n\
<BtnUp>:                    select-end(PRIMARY, CUT_BUFFER0) \n\
<BtnDown>:                  bell(0)
```

在 Tektronix 视窗中缺省绑定是:

```
~Meta<KeyPress>:            insert-seven-bit \n\
Meta<KeyPress>:             insert-eight-bit \n\
!Ctrl <Btn1Down>:           popup-menu(mainMenu) \n\
!Lock Ctrl <Btn1Down>:      popup-menu(mainMenu) \n\
!Ctrl <Btn2Down>:           popup-menu(tekMenu) \n\
!Lock Ctrl <Btn2Down>:      popup-menu(tekMenu) \n\
Shift ~Meta<Btn1Down>:      gin-press(L) \n\
~Meta<Btn1Down>:            gin-press(l) \n\
Shift ~Meta<Btn2Down>:      gin-press(M) \n\
~Meta<Btn2Down>:            gin-press(m) \n\
Shift ~Meta<Btn3Down>:      gin-press(R) \n\
~Meta<Btn3Down>:            gin-press(r)
```

以下是如何运用 **键映射** 操作来添加特殊键来输入通常输入的工作:

```
*VT100.Translations:          #override <Key>F13: keymap(dbx)
*VT100.dbxKeymap.translations:
 \

    <Key>F14:    keymap(None) \n\
    <Key>F17:    string("next") string(0x0d) \n\
    <Key>F18:    string("step") string(0x0d) \n\
    <Key>F19:    string("continue") string(0x0d) \n\
    <Key>F20:    string("print ")
                 insert-selection(PRIMARY,CUT_BUFFER0)
```

环境

xterm 命令为创建的大小窗口设置适当的 **TERM** 和 **TERMCAP** 环境变量。它也使用和设置 **DISPLAY** 环境变量来指定使用哪个位图显示终端。**WINDOWID** 环境变量设置为 xterm 窗口的 X-Windows 标识号。

程序错误

有一些系统上不能执行大容量的粘贴。这不是 **xterm** 命令的程序错误；这是那些系统的伪终端驱动程序的程序错误。 **xterm** 命令以 pty 能接受数据的速度将大容量粘贴反馈给 pty，但有些 pty 驱动程序并没有返回能判断写操作是否已经成功的足够的信息。

在 **xterm** 命令启动后很多选项不能重新设置。

只支持固定宽度、字符单元字体。

控制序列

本节列出 **xterm** 命令可用的控制序列。

定义

以下信息显示如何解释本节中的组合键。

c	文字符号 *c*。
C	单个（需要的）字符。
P_s	单个（通常是可选的）数字参数，由一个或多个数字组成。

P_m 由任意多个单个数字参数组成的多数字参数，由 **;**（分号）字符分隔。

P_t 由可打印的字符组成的文本参数。

VT100 方式

这些控制序列中大部分是标准 VT102 控制序列，但这里也有一些来自稍后 DEC VT 终端的序列。不支持的主要的 VT102 功能特征是平滑滚动、双倍大小字符、闪烁字符和 VT52 方式。

有一些附加的控制序列提供 xterm- 从属函数，例如滚动条或窗口大小。其中功能由 DEC 或 ISO 6429 指定，分配给它的代码在圆括号里给出。指定字符集的转义码由 ISO 2022 指定；请参阅有关文档获取字符集的论述。

控制序列	描述
BEL	响铃（Ctrl+G）
BS	回退（Ctrl+H）
TAB	水平制表符（HT）（Ctrl+I）
LF	换行或新起一行（NL）（Ctrl+J）
VT	垂直制表符（Ctrl+K），与 LF 相同
FF	换页或新起一页（NP）（Ctrl+L），与 LF 相同
CR	回车符（Ctrl+M）
SO	移出（Ctrl+N）–> 切换至备用字符集：调用 G1 字符集。
SI	移入（Ctrl+O）–> 切换至标准字符集：调用 G0 字符集（缺省值）。
ESC # 8	DEC 屏幕测试（DCECALN）
ESC (*C*	指定 G0 字符集（ISO 2022） ***C* = 0** DEC 特殊字符和线图集 ***C* = A** 联合王国（UK） ***C* = B** 美国（USASCII）
ESC) *C*	指定 G1 字符集（ISO 2022） ***C* = 0** DEC 特殊字符和线图集 ***C* = A** 联合王国（UK） ***C* = B** 美国（USASCII）
ESC * *C*	指定 G2 字符集（ISO 2022） ***C* = 0** DEC 特殊字符和线图集 ***C* = A** 联合王国（UK） ***C* = B** 美国（USASCII）
ESC + *C*	指定 G3 字符集（ISO 2022） ***C* = 0** DEC 特殊字符和线图集 ***C* = A** 联合王国（UK） ***C* = B** 美国（USASCII）
ESC 7	保存光标（DECSC）
ESC 8	恢复光标（DECRC）
ESC =	应用程序小键盘（DECPAM）

控制序列	描述
ESC >	正常小键盘（DECNM）
ESC D	索引（IND）
ESC E	下一行（NEL）
ESC H	制表符设置（HTS）
ESC M	逆向索引（RI）
ESC N	G2 字符集的单精度 2 位选择（SS2）：只影响下一个字符。
ESC P	G3 字符集的单精度 2 位选择（SS2）：只影响下一个字符。
ESC O P_t **ESC **	设备控制字符串（DCS）。xterm 没有实现 DCS 功能；P_t 被忽略。P_t 不需要是可打印字符。
ESC Z	返回终端标识符（DECID）。**ESC [** ***c*** (DA) 的废弃的格式
ESC [P_s **@**	插入多个字符中的 P_s（空白）字符（缺省值=1）（ICH）
ESC [P_s **A**	光标向上 P_s 倍数（缺省值=1）（CUU）
ESC [P_s **B**	光标向下 P_s 倍数（缺省值=1）（CUD）
ESC [P_s **C**	光标向前 P_s 倍数（缺省值=1）（CUF）
ESC [P_s **D**	光标向后 P_s 倍数（缺省值=1）（CUB）
ESC [P_s **;** P_s **H**	光标位置 [行;列]（缺省值=1）（CUP）
ESC [P_s **J**	擦除显示器（ED） P_s **= 0** 向下清除（缺省值） P_s **= 1** 向上清除 P_s **= 2** 全部清除
ESC [P_s **K**	擦除行（EL） P_s **= 0** 清除到右边（缺省值） P_s **= 1** 清除到左边 P_s **= 2** 全部清除
ESC [P_s **L**	插入 P_s 行（缺省值=1）（IL）
ESC [P_s **M**	删除 P_s 行（缺省值=1）（DL）
ESC [P_s **P**	删除 P_s 字符（缺省值=1）（DCH）
ESC [P_s **;** P_s **;** P_s **;** P_s **;** P_s **T**	初始化高亮度鼠标轨迹。参数为 [*Func*;*Startx*;Starty;*FirstRow*;*LastRow*]。请参阅第 197 页的『鼠标跟踪』。
ESC [P_s **c**	SendDevice 属性（DA）删除 P_s 字符（缺省值=1）（DCH） P_s **=0 或被省略** 从终端请求属性 **ESC [? 1 ; 2 c** （“我是有高级视频选项的 VT100。”）
ESC [P_s **;** P_s **f**	水平和垂直位置 [行;列]（缺省值是 [1;1]）（HVP）
ESC [P_s **g**	制表符清除（TBC） P_s **= 0** 清除当前列（缺省值） P_s **= 3** 全部清除

控制序列	描述
ESC [P_m **h**	设置方式（SM） P_s **= 4** 插入方式（IRM） P_s **= 2 0** 自动换行（LNM）
ESC [P_m **l**	复位方式（RM） P_s **= 4** 替换方式（IRM） P_s **= 2 0** 正常换行（LNM）
ESC [P_m **m**	字符属性（SGR） P_s **= 0** 正常（缺省值） P_s **= 1** 粗体 P_s **= 4** 下划线 P_s **= 5** 闪烁（显示为粗体） P_s **= 7** 逆相
ESC [P_s **n**	设备状态报告（DSR） P_s **= 5** 状态报告 **ESC [0 n** (″OK″) P_s **= 6** 报告光标位置（CPR） [row;column] as **ESC [r ;** *c* **R** P_s **= 2 0** 自动换行（LNM）
ESC [P_s **;** P_s **r**	设置滚动区域 [顶部;底部]（缺省值=视窗整个大小）（DECSTBM）
ESC [P_s **x**	请求终端参数（DECREQTPARM）

控制序列	描述
ESC [? P_m h	DE 专用方式（DECSET） **P_s = 1** 应用程序光标键（DECCKM） **P_s = 2** 对字符集 G0–G3 指定 USASCII。（在 VT102 中，这选择 VT52 方式（DECANM），而该方式 **xterm** 并不支持。） **P_s = 3** 132 列方式（DECCOLM） **P_s = 4** 平滑（慢）滚动（DECSCLM） **P_s = 5** 反转视频（DECSCNM） **P_s = 6** 初始方式（DECOM） **P_s = 7** 回绕方式（DECAWM） **P_s = 8** 自动重复键（DECARM） **P_s = 9** 设置按钮按下时的鼠标 X 和 Y 坐标。请参阅第 197 页的『鼠标跟踪』。 **P_s = 3 8** 进入 Tektronix 方式（DECTEK） **P_s = 4 0** 允许 80 <—> 132 方式 **P_s = 4 1** **curses** 功能修订 **P_s = 4 4** 打开页边距铃声提醒 **P_s = 4 5** 逆向回绕方式 **P_s = 4 7** 使用备用屏幕缓冲区（直到被 **titeInhibit** 资源禁用） **P_s = 1 0 0 0** 设置按钮按下和释放时的鼠标 X 和 Y 坐标。请参阅第 197 页的『鼠标跟踪』。 **P_s = 1 0 0 1** 使用高亮度的鼠标跟踪。

控制序列	描述
ESC [? P_m l	DEC 专用方式复位（DECRST） **P_s = 1** 正常光标键（DECCKM） **P_s = 3** 80 列方式（DECCOLM） **P_s = 4** 快速跳动滚动（DECSCLM） **P_s = 5** 正常视频（DECSCNM） **P_s = 6** 正常光标方式（DECOM） **P_s = 7** 无回绕方式（DECAWM） **P_s = 8** 无自动重复键（DECARM） **P_s = 9** 按钮按下时不发送鼠标的 X 和 Y 坐标。 **P_s = 4 0** 禁止 80 <—> 132 方式 **P_s = 4 1** 无 **curses** 功能修订 **P_s = 4 4** 关闭页边距铃声提醒 **P_s = 4 5** 无逆向回绕方式 **P_s = 4 7** 使用正常屏幕缓冲区 **P_s = 1 0 0 0** 在按钮按下和释放时不发送鼠标 X 和 Y 坐标。 **P_s = 1 0 0 1** 不使用高亮度的鼠标跟踪。xxx
ESC [? P_m r	恢复 DEC 专用方式值。该先前保存的 **P_s** 值被恢复。**P_s** 值和 DECSET 相同。
ESC [? P_m s	保存 DEC 专用方式值。**P_s** 值和 DECSET 相同。
ESC]? P_s ; P_t BEL	设置文本参数 **P_s = 0** 将图标名和窗口标题更改为 **P_t** **P_s = 1** 将图标名更改为 **P_t** **P_s = 2** 将窗口标题更改为 **P_t** **P_s = 5 0** 将字体设置为 **P_t**
**ESC P_t ESC **	专用消息（PM）。**xterm** 未实现 PM 功能；**P_t** 不需要是可打印字符。
**ESC _ P_t ESC **	应用程序命令（APC）。专用消息（PM）。**xterm** 未实现 APC 功能；**P_t** 被忽略。**P_t** 不需要是可打印字符。
ESC c	完全复位（RIS）
ESC n	选择 G2 字符集（LS2）
ESC o	选择 G3 字符集（LS3）

控制序列	描述
ESC I	把 G3 字符集作为 GR（LS3R）调用。对 **xterm** 没有可见的效果。
ESC }	把 G2 字符集作为 GR（LS2R）调用。对 **xterm** 没有可见的效果。
ESC	把 G1 字符集作为 GR（LS1R）调用。对 **xterm** 没有可见的效果。

XTERM 描述限制

在 AIX V4中的 DEC.TI 文件中的 xterm 终端描述可通过使用 SGR 属性来提供下划线方式。该 SMUL 和 RMUL 属性当前并没有在 AIX V4的 XTERM 终端描述中定义。使用名为 SGR 更为通用的功能。

```
tput sgr x y
```

其中 *x* 是 1 或 0，分别用于打开或关闭突出显示方式，*y* 是 1 或 0，分别用于打开或关闭下划线方式。请参阅文章《**terminfo** 文件格式》以获得有关 SGR 功能的更多信息。

```
tput sgr 0 1    关闭突出显示; 打开下划线
tput sgr 0 0    关闭突出显示; 关闭下划线
tput sgr 1 1    打开突出显示; 打开下划线
tput sgr 1 0    打开突出显示; 关闭下划线
```

鼠标跟踪

VT 小窗口可以设置为在按钮按下时发送鼠标位置和其它信息。该方式典型情况下用于编辑器和其它希望使用鼠标的全屏方式应用程序。

有三种互斥的方式，每种都通过 DECSET（或 DECRST）转义序列中不同的参数来启用（或禁用）。**xterm** 命令生成的所有鼠标跟踪转义序列参数编码成单个字符的数字参数，如: *value*+040。该屏幕坐标系基于 1。

例如 **!** 是 1。该屏幕坐标系基于 1。

X10 兼容性方式在按钮按下时发送一个转义序列，对位置和按下的鼠标按钮进行编码。该方式是通过指定参数 DECSET 为 9 启用的。在按钮按下时，**xterm** 命令发送以下 “6 个字符”。*C*b 是按钮 1。*C*x 和 *C*y 是按钮按下时鼠标的 *x* 和 *y* 坐标。

ESC [M $C_bC_xC_y$

正常轨迹方式发送在按钮按下和松开的时候都会发送一个转义序列。修饰符信息也被发送。通过指定参数 DECSET 为 1000 可以启用该方式。在按钮按下或释放时，**xterm** 命令发送以下 “组合键”：

ESC [M $C_bC_xC_y$

C_b 的低两位编码按钮信息: 0=MB1 按下，1=MB2 按下，2=MB3 按下， 3=释放。高位编码在按钮按下时哪个修改量关闭并将它们加在一起。4=Shift，8=Meta，16=Control。C_x 和 C_y 是鼠标事件的 *x* 和 *y* 坐标。左上角是（1，1）。

鼠标高亮度跟踪向程序通知按钮按下，从程序中得到了一定范围的行，并用在该范围内突出显示鼠标覆盖的区域直到鼠标释放，然后向程序发送释放时的坐标。该方式是通过指定参数 DECSET 为 1001 启用的。

警告: 该方式的使用需要合作程序，否则它将挂起 **xterm** 命令。在按钮按下时，生成和正常跟踪相同的信息; **xterm** 命令然后等待程序发送鼠标跟踪信息。*所有 X 事件都被忽略，直到从 pty 接收到以下正确的转义序列*:

ESC [P_s **;** P_s **;** P_s **;** P_s **; T**

参数为 *Func*、*Startx*、*Starty*、*FirstRow* 和*LastRow*。*Func* 参数是非零值，用来初始化高亮度跟踪，0（零）则表示放弃。*Startx* 和 *Starty* 参数给出突出显示区域的开始 x 和 y 位置。结束位置跟踪鼠标轨迹，但不会在行 *FirstRow* 之上而且始终 *LastRow* 行之上。（屏幕顶端是行 1。）当按钮释放时，**xterm** 命令以两种方法之一报告结束位置：如果开始和结束坐标都是有效的文本位置，**xterm** 命令按以下方式报告“结束位置”：

ESC [t C_xC_y

如果任一个坐标超过了行尾，**xterm** 命令按如下方式报告“结束位置”：

ESC [T $C_xC_yC_xC_yC_xC_y$

参数为 *Startx*、*Starty*、*Endx*、*Endy*、*Mousex* 和 *Mousey*。*Startx*、*Starty*、*Endx* 和 *Endy* 参数给出该区域的开始和结束字符位置。*Mousex* 和 *Mousey* 参数给出鼠标弹起时的位置，它不超过一个字符。

Tektronix 4014 方式

大多数序列是标准 Tektronix 4014 控制序列。未提供的主要功能是直写和散焦方式。本文档没有描述在各种不同 Tektronix 绘图方式中用的命令，但描述了开关方式的命令。

相关信息

aixterm 命令、**resize** 命令、**tset** 命令、**vi** 或 **vedit** 命令。

---

xwd 命令

用途

转储增强 X-Windows视窗的图像。

语法

xwd [**-add** *Value*] [**-frame**] [**-display** *Display*] [**-help**] [**-nobdrs**] [**-xy**] [**-out** *File*] [**-root** | **-id** *id* | **-name** *Name*] [**-icmap**] [**-screen**]

描述

xwd 命令是增强 X-Windows 的窗口转储实用程序。**xwd** 命令允许将视窗映像存储为特殊的已格式化的转储文件。然后，该文件可以被其它各种可执行重显、打印、编辑、格式化、归档和图像处理等功能的 X 实用程序读取。通过在期望的窗口中单击鼠标来选择目标窗口。键盘铃会在转储开始时响一次，转储结束后响二次。

标志

-add *Value*	指定添加到每个像素的有签名的值。该选项是 X11R5 特有的。
-frame	该选项表明在手工选择窗口时，视窗管理器框架也应该包含在内。
-display *Display*	指定服务器连接。
-help	打印使用命令的语法摘要。

-nobdrs 指定窗口转储不包含构成 X-Windows 边框的像素。如果想将视窗内容包含在文档中作为说明时，这十分有用。**-nobdrs** 标志的结果取决于那个视窗管理器在运行。很多视窗管理器都从客户机中除去所有边框。例如，**XGetWindowAttributes** 函数在客户机启动时返回一个值为 0 的 `border_width` 字段而不管真正的边框宽度为多少。因此，任何在屏幕上可见的边沿都属于视窗管理器；客户机对其则一无所知。在这种情况下，**-nobdrs** 标志没有作用。

-out *File* 在命令行上指定输出文件。缺省值是输出到标准输出。

-root 表明在窗口转储时根视窗应该被选中，而不需要请求用户用指针选择窗口。该选项是 X11R5 特有的。

-id *id* 表明在窗口转储时具有指定资源标识符的窗口应该被选中，而不需要请求用户用指针选择视窗。该选项是 X11R5 特有的。

-name *Name* 表明在窗口转储时具有指定 WM_NAME 属性的窗口应该被选中，而不需要请求用户用指针选择窗口。该选项是 X11R5 特有的。

-icmap 强制使用屏幕的第一个已安装的颜色表来获取 RGB 值。缺省情况下，使用选中窗口的颜色表。该选项是 X11R5 特有的。

-screen 表明用以获取图像的 GetImage 请求应该在根视窗上完成，而不是直接在指定的视窗上。用这种方式可以获取与指定视窗重叠的其它视窗，而且，更为重要的是可以捕获那些作为独立视窗而显示在指定视窗之上的菜单和弹出菜单。该选项是 X11R5 特有的。

-xy 选择 xy 格式转储而不是使用缺省的 z 格式。该选项只适用彩色显示器。

文件

XWDFile.h X-Windows 转储文件格式定义文件。

相关信息

xwud 命令。

---

xwud 命令

用途

检索和显示增强的 X-Windows 的转储图像。

语法

xwud [**-in** *FileName*] [**-noclick**] [**-geometry** *Geometry*] [**-display** *Display*] [**-new**] [**-std** *MapType*] [**-raw**] [**-vis visual_type** | **visual_id**] [**-help**] [**-rv**] [**-plane** *Number*] [**-fg** *Color*] [**-bg** *Color*]

描述

xwud 检索增强的 X-Windows 的转储图像。它通过在一个视窗中显示由 **xwd** 命令先前产生的特殊格式的转储文件来实现上述功能。转储文件格式由 **XWDFile.h** 文件决定。

可以用标志指定彩色显示器、视窗大小及位置、输入字段和可视类或标识。也可以选择要显示的图像单一层面。

标志

-bg *Color*
指定在显示一幅位图图像（或图像的单一层面）时图像中显示的 **0**（零）位的颜色。

-display *Display*
指定要连接的服务器；请参阅 **X** 命令。

-fg *Color*
指定在显示一幅位图图像（或图像的单一层面）时图像中显示的 **1** 位的颜色。

-geometry *Geometry*
指定视窗的大小和位置。典型用法是只要指定位置，让大小缺省设置为图像的实际大小。

-help
打印允许选项的简短描述。

-in *FileName*
在命令行指定输入文件。如果没有指定输入文件，就采用标准输入。

-new
为显示图像创建一个新的颜色映射。如果图像特征匹配显示器的特征，这个标志可以在屏幕上更快地显示图像，但这是以使用新的颜色映射为代价的（在很多终端上这会导致其它视窗也显示为彩色）。

-noclick
在单击视窗中的一个按钮时阻止应用程序结束。通过输入字符 q 或 Q，或是用 Ctrl-C 组合键来结束应用程序。

-plane *Number*
选择要显示的图像单一层面。平面用数字标识，0（零）是最低有效位。用该标志决定哪个层面传递到 **xpr** 命令打印。

-raw
用当前屏幕上存在的任何颜色显示转储图像。这个标志在回存图像到图像原来所在的相同屏幕，而该源视窗还在屏幕上时很有用。这样可以更快地在屏幕上获取图像。

-rv
如果位图图像（或图像的单一层面）显示时交换前景色和背景色。这个标志在显示区分像素值为 0 和 1（从其在显示器上的显示逆向获得）的颜色的位图图像时很有用。

-std *MapType*
使用指定的标准颜色映射显示图像。通过将类型转换为大写字母，并加前缀 **RGB_** 和添加后缀 **_MAP**，可以获取映射类型。典型的映射类型为 **best**、**default** 和 **gray**。请参阅 **/usr/lpp/X11/Xamples/clients/xstdcmap** 有关创建标准颜色映射的信息。

-vis visual_type | visual_id
指定特定的可视类型或可视标识。缺省值选取**最好** 的一个，或者您指定 **default**，这和根视窗的颜色映象图的类相同。

也可以指定一个特定的类：**StaticGray**、**GrayScale**、**StaticColor**、**PseudoColor**、**DirectColor**、**TrueColor**。指定 **Match** 使用和源图像一样的类。

指定一个十六进制的数（以 0x 为前缀）或是一个十进制数作为确切的可视标识（对服务器是特定的）。该字符串不区分大小写。

环境变量

DISPLAY 获取缺省显示器。

示例

要在转储视窗检索一个特定的文件，输入：

```
xwud -in FileName
```

相关信息

X 命令、**xpr** 命令、**xwd** 命令。

---

yacc 命令

用途

从与上下文无关的语法规范所组成的输入生成 LALR(1) 语法分析程序。

语法

yacc [**-b** *Prefix*] [**-C**] [**-d**] [**-l**] [**-Nn***Number*] [**-Nm***Number*] [**-Nr***Number*] [**-p** *Prefix*] [**-s**] [**-t**] [**-v**] [**-y** *Path*] *Grammar*

描述

yacc 命令将与上下文无关的语法规范转换成一组表格，执行 LALR(1) 分析算法的简单自动机会用到该组表格。语法可能是有歧义的；此时，使用指定的优先规则来解决语法的二义性。

必须用 C 语言编译器编译输出文件 **y.tab.c**，以生成 **yyparse** 函数。这个函数必须由 **yylex** 词法分析器，**main** 子例程以及 **yyerror** 错误处理子例程（必须提供这些子例程）装入。**lex** 命令对创建可供 **yyparse** 子例程使用的词法分析器很有用。简化版本的 **main** 和 **yyerror** 子例程通过 **yacc** 库 **liby.a** 可以得到。而且，**yacc** 能够用来生成 C++ 输出。

可以用**-DYACC_MSG** 选项编译 **yacc** 生成的 C 文件（**y.tab.c**）来包含使用消息工具必须的代码。当在编译时使用该选项，由 **yyparse** 子例程和 **YYBACKUP** 宏生成的错误消息会从 **yacc_user.cat** 目录中抽取。

这允许在非英语语言环境中以不同于英语的语言接收错误消息。如果找不到目录或打不开目录，**yyparse** 和 **YYBACKUP** 子例程显示缺省英文消息。

yacc 命令受 **LANG**、**LC_ALL**、**LC_CTYPE** 和 **LC_MESSAGES** 环境变量影响。

标志

-b *Prefix* 使用 *Prefix* 取代 **y** 作为所有输出文件名的前缀。代码文件 **y.tab.c**，头文件 **y.tab.h**（在 **-d** 指定时创建）和描述文件 **y.output**（在 **-v** 指定时创建）分别更改为 *Prefix*.**tab.c**、*Prefix*.**tab.h** 和 *Prefix*.**output**。

-C 为和 C++ 编译器一起使用生成 **y.tab.C** 文件取代 **y.tab.c** 文件。要使用输入输出的 I/O 流库，定义宏 **_CPP_IOSTREAMS**。

-d 产生文件 **y.tab.h**。这包含 **#define** 语句，该语句将 **yacc** 分配的标记代码与标记名关联起来。这允许除了 **y.tab.c** 的源文件通过包含头文件来访问标记代码。

-l 在 **y.tab.c** 中不包含任何 **#line** 构造。只在语法和关联操作全都调试过后才这个标记。

-Nn*Number* 将标记名和非终端名数组的大小更改为 *Number*。缺省值是 8000。有效值应该大于 8000。

-Nm*Number* 将内存状态数组大小更改为 *Number*。缺省值是 40000。有效值应该大于 40000。

-Nr*Number* 更改内部缓冲区大小以处理大型语法。缺省值是 2000。有效值应该大于 2000。

-p *Prefix* 使用 *Prefix* 取代 **yy** 作为所有由 **yacc** 命令创建的文件外部名的前缀。受影响的外部名包括：**yychar**、**yylval**、**yydebug**、**yyparse( )**、**yylex( )** 和 **yyerror( )**。（以前，**-p** 被用来指定备用解析器；现在 **-y***Path* 可以用来指定备用解析器。）

-s 将 **yyparse** 函数分解成几个更小的函数。由于该函数的大小多少和它的文法成比例，这样有可能 **yyparse** 函数变得太大无法编译、优化或有效地执行。

-t 编译运行时的调试代码。缺省方式下，在编译 **y.tab.c** 时不包含这些代码。然而，运行时调试代码是由预处理器宏 **YYDEBUG** 控制的。如果 **YYDEBUG** 有非零值，C 编译器（**cc**）包含调试代码，而不管是否使用 **-t** 标志。如果不想将调试代码包含在编译器中，**YYDEBUG** 应该值为 0。在不编译这个代码的情况下，**yyparse** 子例程将会有更快的执行速度。

-t 标志会导致对调试代码的编译，但实际上它不打开调试方式。要获取调试输出，**yydebug** 变量必须通过添加 C 语言声明 `int yydebug=1` 到 **yacc** 语法文件的声明部分来设置，或者设置通过 **dbx** 的 **yydebug** 进行上述设置。

-v 准备文件 **y.output**。它包含可读的语法分析表描述和由于文法二义性而生成的冲突报告。

-y *Path* 使用由 *Path* 而不是缺省的 **/usr/lib/yaccpar** 文件来指定的解析器原型。（以前，**-p** 用来指定备用解析器。）

出口状态

本命令返回以下出口值:

0 成功完成。
>0 发生错误。

示例

1. 以下命令:

   ```
   yacc grammar.y
   ```

   从 **grammar.y** 文件提取 **yacc** 规则，并将输出置于 **y.tab.c**。

2. 以下命令:

   ```
   yacc  -d grammar.y
   ```

   与示例 1 的作用相同，但它也产生 **y.tab.h** 文件，对于 **grammar.y** 文件中定义的每个标记，该文件包含 C 风格的 **#define** 语句。

文件

y.output	包含可读的语法分析表描述和由于语法二义性而生成的冲突报告。
y.tab.c	包含输出文件。
y.tab.h	包含标记名的定义。
yacc.tmp	临时文件。
yacc.debug	临时文件。
yacc.acts	临时文件。
/usr/ccs/lib/yaccpar	包含 C 程序的解析器原型。
/usr/ccs/lib/liby.a	包含运行时程序库。

相关信息

lex 命令。

AIX 5L Version 5.2 General Programming Concepts: Writing and Debugging Programs 中的 Creating an Input Language with the lex and yacc Commands。

AIX 5L Version 5.2 General Programming Concepts: Writing and Debugging Programs 中的 >Example program for the lex and yacc programs。

yes 命令

用途

重复输出肯定的响应。

语法

yes [*charstring*]

描述

yes 命令重复输出肯定的响应。使用 **yes** 命令作为到在它完成指定操作前要求肯定响应的另一个命令的管道输入。例如，**yes** 命令在删除某个目录中的多个文件时很有用。Ctl-C 组合键可以终止连续的肯定响应。

> **注**: 当前语言环境是由 **LC_MESSAGES** 环境变量或 *charstring* 参数决定的（如果指定的话）。 *charstring* 参数可以是任何单一的字符或字符流。如果您在发出 **yes** 命令后输入 *charstring* 参数，则直到键入 Ctl-C 组合键才会在屏幕上显示 *charstring* 参数。

示例

要将单词 `first` 显示到屏幕上，输入:

```
yes first
```

除非输入 Ctl-C 组合键来结束，本语句会显示单词 first。

文件

/usr/bin/yes	包含 **yes** 命令。

相关信息

环境文件。

*《AIX 5L V5.2 系统用户指南: 操作系统与设备》*中的『shell 概述』。

ypbind 守护程序

用途

使客户机进程能绑定或连接到 NIS 服务器。

语法

/usr/lib/netsvc/yp/ypbind [**-s -ypset -ypsetme**]

描述

ypbind 守护程序将网络信息服务（NIS）客户机上的进程绑定或连接到 NIS 服务器上的服务。运行在每个 NIS 客户机上的该守护程序可以用以下系统资源控制器（SRC）命令启动或停止:

```
startsrc -s ypbind
stopsrc -s ypbind
```

当客户机从网络信息服务（NIS）映射请求信息时，**ypbind** 守护程序在网络上进行广播以寻找服务器。当服务器响应时，它给守护程序提供主机因特网地址和端口号。这是提供客户机正在寻找的信息的主机。**ypbind** 守护程序将此地址信息存储于 **/var/yp/binding** 目录，且文件名使用 **domainname.version**。然后，下一次客户机想访问 NIS 映射时，客户机的 **ypbind** 守护程序会参考 **domainname.version** 文件中的地址。

ypbind 守护程序可以同时维持到几个域及其服务器 **-ypsetme** 的绑定。缺省域是由 **domainname** 命令在启动时间指定的域。

注:

1. 如果域出现不能绑定的情况（通常是在服务器崩溃或过载的情况下），**ypbind** 守护程序会再次广播查找另一个服务器。
2. 要强制客户机绑定到某一个特定服务器上，请使用 **ypset** 命令。
3. 要找出客户机绑定到哪个服务器，请使用 **ypwhich** 命令。
4. 如果 **/var/yp/binding/domainname/ypservers** 文件存在， **ypbind** 将在广播前尝试联系哪个文件中列出的服务器。该文件应该包含一个服务器 IP 地址的列表，每行一个。
5. 缺省情况下，NIS 客户机将无限期地等待 NIS 服务器，在此期间，不可能登录到客户机系统。然而，限制等待的时间长度是可能的。如果 **YPBIND_MAXWAIT** 环境变量是在 **ypbind** 守护程序启动前设置的（通常是在**/etc/environment** 中），这个值（以秒计）将限制 NIS 客户机等待 NIS 服务器的时间数量。如果超出了该限制，客户机表现好像 NIS 不可用而且继续使用本地文件。这将允许本地登录（例如 root 用户）。

标志

-s	在安全方式下用特权通信端口运行 **ypbind** 守护程序。
-ypset	指明本地主机接受从本地主机或远程主机来的 **ypset** 命令。
-ypsetme	表明本地主机只接受从本地主机来的 **ypset** 命令。如果和**-ypset** 标志一起指定，则覆盖 -ypset。

注:

1. 如果 **-ypset** 或 **-ypsetme** 标志两者都没有指定，本地主机拒绝所有主机的全部 **ypset** 命令。这是最安全的方式，因为 NIS 服务器不可能更改。
2. 如果 **-ypset** 或 **-ypsetme** 标志两者都没有指定，本地主机拒绝所有主机的全部 **ypset** 命令。这是最安全的方式，因为 NIS 服务器不可能更改。然而，如果在网络上不存在 NIS 服务器直接连接到客户机上，**-ypsetme** 标志必须使用而且 NIS 服务器应该用 **ypset** 命令指定。

文件

/var/yp/binding 目录	包含 NIS 服务器的因特网地址和端口号。
/var/yp/binding/domainname/ypservers	包含服务器因特网地址列表，每行一个，并会在广播前尝试联系这些服务器。
domainname.version	包含当前 NIS 服务器地址和端口号的二进制文件。

相关信息

domainname 命令、**makedbm** 命令、**mkclient** 命令，**mkmaster** 命令、**mkslave** 命令、**ypcat** 命令、**ypinit** 命令、**ypmatch** 命令、**yppoll** 命令、**yppush** 命令、**ypset** 命令、**ypwhich** 命令、**ypxfr** 命令。

AIX 5L Version 5.2 System Management Concepts: Operating System and Devices 中的 System Resource Controller Overview。

*《AIX 5L V5.2 系统管理指南: 通信与网络》*中的『系统管理的网络文件系统（NFS）概述』。

AIX 5L Version 5.2 Network Information Services (NIS and NIS+) Guide 中的 Network Information Service (NIS) Overviewfor System Management。

NIS 参考。

---

ypcat 命令

用途

打印出网络信息服务 (NIS)映射。

语法

要显示网络信息服务数据库

/usr/bin/ypcat [**-k**] [**-t**] [**-d** *域名*] *MapName*

要显示别名转换表

/usr/bin/ypcat -x

描述

该 **ypcat** 命令打印出您用 *MapName* 参数指定的网络信息服务（NIS）映射。您可以指定一个映射名或是映射的别名。因为**ypcat** 命令使用 NIS 服务，您不用指定服务器。

标志

-k	显示映射中键值为空或是不为映射值一部分的关键字。（任何从在 **/etc** 目录下有二进制版本的文件派生出来的映射都不属于这个类。）
-t	表明由 *MapName* 参数指定的名字不是别名。该标志导致 **ypcat** 命令忽略别名转换表只搜索 *MapName* 参数指定的映射。
-d *域名*	搜索指定映射的指定域。
-x	显示别名转换表。该表列出了该命令知道的映射别名并表示出与每个别名关联的（由 *MapName* 参数指定的）映射名。

示例

1. 察看全网络的密码映射，**passwd.byname**，输入:

   ```
   ypcat passwd
   ```

   在该示例中，`passwd` 是 **passwd.byname** 映射的别名。
2. 要定位一个映射，输入:

```
ypcat -t passwd
```

在该示例中，**ypcat** 命令忽略任何别名为 passwd 的映射，并搜索全名为 passwd 的映射。

3. 要显示另一个域的映射图，输入:

```
ypcat -d polaris passwd
```

在本示例中，**ypcat** 命令在名为 polaris 的域中定位名为 passwd 的映射。

4. 要显示映射图别名转换表，输入:

```
ypcat -x
```

在该示例中，**ypcat** 命令显示映射别名列表和它们相关的映射名。

相关信息

domainname 命令、**ypmatch** 命令。

ypserv 守护程序。

*《AIX 5L V5.2 系统管理指南: 通信与网络》*中的『系统管理的网络文件系统（NFS）概述』。

AIX 5L Version 5.2 Network Information Services (NIS and NIS+) Guide 中的 Network Information Service (NIS) Overviewfor System Management。

NIS 参考。

---

ypinit 命令

用途

在网络信息服务（NIS）服务器上设置 NIS 映射。

语法

在 NIS 主控服务器上设置 NIS

/usr/sbin/ypinit [**-o**] [**-n**] [**-q**] **-m** [*SlaveName* ...]

在 NIS 从属服务器上设置 NIS

/usr/sbin/ypinit -s *MasterName*

描述

ypinit 命令在网络信息服务（NIS）主控服务器或 NIS 从属服务器上设置 NIS 映射。只有拥有 root 用户权限的用户才能使用 **ypinit** 命令。

缺省情况下，**ypinit** 命令使用 ASCII 文件系统作为所创建映射的输入文件。

标志

-m [*SlaveName...*]
表明本地主机将作为 NIS 主控服务器。如果使用 **-q** 标志，则 **-m** 标志后可以跟有将设置为 NIS 从属服务器的机器名。

-n
表明即使发现错误 **ypinit** 命令也不会停止。

-o	允许当前 NIS 域的任何现有映射都将被覆盖。
-q	表明 **ypinit** 命令直接从命令行获取自变量而不提示输入。
-s *MasterName*	从用 *MasterName* 参数指定的服务器工作站中复制 NIS 映射。

示例

1. 要设置作为所有 NIS 映射的主控服务器的 NIS 主控服务器，在命令行输入以下命令:

   ```
   ypinit -m
   ```

   这个命令调用 **make** 过程，执行 **/var/yp/Makefile** 文件中的指令。
2. 要设置 NIS 从属服务器，输入:

   ```
   ypinit -s zorro
   ```

   在该示例中，**ypinit** 命令将 NIS 映射从名为 `zorro` 的 NIS 服务器复制到所在的工作站，并使工作站作为 NIS 从属服务器。
3. 要设置 NIS 主控服务器而不用提示输入，输入:

   ```
   ypinit -o -n -q -m slave
   ```

   注: 如果系统以前被配置为 NIS 主控服务器，必须确保在执行 **ypinit** 前目录 **/var/yp/binding** 已经被除去。如果旧的信息存储在 **/var/yp/binding**，在配置 NIS 主控服务器时可能会导致错误。

文件

/etc/bootparams	列出可以用于无盘客户机引导的客户机。
/etc/passwd	包含已允许登录到机器的每个用户的项。
/etc/group	包含已允许登录到机器的每个用户组的项。
/etc/hosts	包含在网络上的每个主机的一个条目。
/var/yp/Makefile	包含制作 NIS 映射的规则。
/etc/networks	包含在 DARPA 网际网上的每个网络的名称。
/etc/netmasks	列出用于实现 IP 标准子网络的网络掩码。
/etc/netid	包含机器、主机和组的标识信息。
/etc/rpc	包含 RPC 程序的映射信息。
/etc/services	包含在因特网上可用的每个服务器的一个条目。
/etc/protocols	定义在本地主机上使用的因特网协议。
/etc/netgroup	包含网络上每个用户组的信息。
/etc/ethers	包含因特网网络上主机的以太网地址。
/etc/publickey	包含用于 NIS 映射的公共或保密密钥。

相关信息

chmaster 命令、**chslave** 命令、**lsmaster** 命令、**makedbm** 命令、**mkmaster** 命令、**mkslave** 命令、**yppush** 命令、**ypxfr** 命令。

ypserv 守护程序。

AIX 5L Version 5.2 Network Information Services (NIS and NIS+) Guide 中的 Network Information Service (NIS) Overviewfor System Management。

NIS 参考。

ypmatch 命令

用途

显示网络信息服务（NIS）映射中给定键的值。

语法

显示 NIS 映射的键值

/usr/bin/ypmatch [**-d** *Domain*] [**-k**] [**-t**] *Key... MapName*

显示 NIS 映射的别名表

/usr/bin/ypmatch -x

描述

ypmatch 命令显示网络信息服务（NIS）映射中与一个或多个键关联的值。使用 *MapName* 参数指定想搜索的映射的名称或别名。

当在 *Key* 参数中指定多个参数时，系统搜索所有键的同一个映射。因为模式匹配不可用，系统会以每个键的大写格式和长度来做精确匹配。如果没有查找到指定键的匹配值，则显示一条诊断消息。

标志

-d *Domain*	指定一个域而不是缺省域。
-k	在打印键的值前打印冒号后跟随的键值。只有在键值不重复时，或者如果您指定了太多的键而使得输出可能会很混淆时才有用。
-t	禁止将别名转换为映射名。
-x	显示映射的别名表。这会列出这个命令知道的别名（*MapName* 参数指定），并显示与每个别名相关联的映射名。

示例

要显示与一个特定键相关的值，输入:

```
ypmatch -d ibm -k host1 hosts
```

在该示例中，**ypmatch** 命令显示 `host1` 键的值，该键值来自 `ibm` 域中的 `hosts` 映射。

相关信息

ypcat 命令。

AIX 5L Version 5.2 Network Information Services (NIS and NIS+) Guide 中的 Network Information Service (NIS) Overviewfor System Management。

NIS 参考。

yppasswd 命令

用途

更改网络信息服务（NIS）中的网络密码。

语法

yppasswd [**-f** [*Name*] | **-s** [*Name* [*ShellProg*]]]

描述

yppasswd 命令更改（或安装）网络密码，并将它与在 *UserName* 参数中指定的名称关联起来。要创建或更改密码，必须是想更改的密码的所有者。网络信息服务（NIS）密码可以与在自己机器上的密码不同。NIS 服务器上的 root 用户可以在不知道另一用户原密码的情况下更改其密码。要这样做，Root 用户可在用户的原始密码处输入他们的密码。不过在 NIS 客户机的 root 用户没有这项特权。

当在命令行输入 **yppasswd** 命令时，系统会提示输入旧密码。完成后，系统提示输入新密码。如果用大小写字符混合的字符串，输入的密码最少可以是四个字符。否则，密码最少必须为六个字符或更长。当然如果坚持认为足够的话，规则也可以放松。

如果输入的旧密码不正确，必须在系统给出错误消息前输入新密码。系统之所以需要这两个密码是因为 **update** 协议同时将它们发送到服务器。服务器捕获错误并通知输入了错误的旧密码。

要验证新密码，系统提示再输入一次。为使新密码生效，**yppasswdd** 守护程序必须运行在 NIS 服务器上。

注：**yppasswd** 命令不能建立密码规则，密码规则由 **passwd** 建立。

标志

-f [*Name*]	更改在 NIS 映射中用户 *Name* 通用信息。通用信息是存储于 **/etc/passwd** 文件中的一般信息。
-s [*Name* [*ShellProg*]]	更改在 NIS 映射中用户 *Name* 的登录 shell。

示例

1. 要更改用户的 NIS 密码，输入：

   ```
   yppasswd Joe
   ```

   本示例演示如何为名为 Joe 的用户更改 NIS 密码。系统提示输入 Joe 的旧密码然后是他的新密码。
2. 要将名为 Joe 的用户的登录 shell 更改为 **/bin/ksh**，如果 **yppasswdd** 守护程序没有用 **-noshell** 标志启动，输入：

   ```
   yppasswd  -s Joe /bin/ksh
   ```
3. 要更改名为 Joe 的用户在 **passwd** 文件中的通用信息，如果 **yppasswdd** 守护程序没有用 **-nogecos** 标志启动，输入：

   ```
   yppasswd  -f Joe
   Old NIS password:
   Joe’s current gecos:
   ```

```
John Doe Test User Id
Change (yes) or (no)? >y
To?>Joe User Test User Id
```

相关信息

yppasswdd 守护程序。

AIX 5L Version 5.2 Network Information Services (NIS and NIS+) Guide 中的 Network Information Service (NIS) Overviewfor System Management。

NIS 参考。

---

yppasswdd 守护程序

用途

从 **yppasswd** 命令接收和执行请求。

语法

rpc.yppasswdd *FileName* [**-nogecos**] [**-nopw**] [**-noshell**] [**-r** | **-m** [*Argument...*]]

描述

yppasswdd 守护程序是从 **yppasswd** 命令中接收和执行新密码请求的服务器。这些请求需要守护程序去验证用户旧密码并更改它。该守护程序更改以 *FileName* 参数指定的文件中密码，该参数与 **/etc/passwd** 文件格式相同。

要使从远程机器上更新网络信息服务（NIS）密码映射成为可能，**yppasswdd** 守护程序必须运行在包含 NIS 密码映射的主控服务器上。

注: **yppasswdd** 守护程序缺省情况下是不运行的，也不能像其它远程过程调用（RPC）守护程序一样从 **inetd** 守护程序启动。

yppasswdd 守护程序可以用以下 系统资源控制器（SRC）命令启动和停止:

```
startsrc -s yppasswdd
stopsrc -s yppasswdd
```

标志

-m 使用 **/var/yp** 目录下的制作文件运行 **make** 命令。这将新的或更改过的密码添加到 NIS 密码映射中。任何后跟 **-m** 标志的自变量都传给 **make** 命令。

-nogecos 表明服务器将不从 **yppasswd** 命令中接受通用信息的更改。

-nopw 表明服务器将不接受从 **yppasswdd** 命令中接受对密码的更改。

-noshell 表明服务器将不从 **yppasswd** 命令中接受对用户 shell 的更改。

-r 直接用新的或更改过的密码在主控服务器以及从属服务器上更新 **/var/yp/***domainname***/passwd.byname** 和 **/var/yp/***domainname***/passwd.byuid** 数据库文件。该选项比 **-m** 标志快，因为 **make** 命令没有运行。**-r** 标志在数据库文件很大（有几千个条目或更多）时很有用。

注: 系统资源控制器（SRC）使用以缺省方式指定的 **-m** 标志启动 **yppasswdd** 守护程序。使用 **chssys** 命令更改缺省值为 **-r** 标志。

示例

要立即传播更新后的密码，根据以下格式调用 **yppasswdd** 守护程序:

```
startsrc -s yppasswdd
```

文件

/etc/inetd.conf	定义 **inetd** 守护程序如何处理因特网服务请求。
/var/yp/Makefile	包含制作 NIS 映射的规则。
/etc/rc.nfs	包含 NFS 和 NIS 守护程序的启动脚本。
/etc/security/passwd	存储密码信息。

相关信息

chssys 命令、**domainname** 命令、**make** 命令、**passwd** 命令、**startsrc** 命令、**yppasswd** 命令。

inetd 守护程序。

/etc/security/passwd 文件。

AIX 5L Version 5.2 System Management Concepts: Operating System and Devices 中的System Resource Controller Overview。

*《AIX 5L V5.2 系统管理指南: 通信与网络》*中的『系统管理的网络文件系统（NFS）概述』。

AIX 5L Version 5.2 Communications Programming Concepts 中的 Remote Procedure Call (RPC) Overview for Programming。

AIX 5L Version 5.2 Network Information Services (NIS and NIS+) Guide 中的 Network Information Service (NIS) Overviewfor System Management。

NIS 参考。

---

yppoll 命令

用途

显示当前服务器上使用的网络信息服务（NIS）映射的顺序号（标识号）。

语法

/usr/sbin/yppoll [**-h** *Host*] [**-d** *Domain*] *MapName*

描述

yppoll 命令使用 **ypserv** 守护程序显示以 *MapName* 参数指定的映射的顺序号。顺序号是映射的标识号，该号码由系统分配。每当映射被更新时，这个号码就会更改。在想确定服务器使用的是特定映射的最新版本时，请使用 **yppoll** 命令。

yppoll 命令可以在安装了网络信息服务（NIS）协议的版本 1 或版本 2 的系统上运行。不过要注意的是，每个版本的协议都有它自己的诊断消息设置。

注: 指定 *MapName* 时，必须确保输入映射的全名。**yppoll** 命令不能识别映射的别名。

标志

-h *Host* 使您可以指定一个服务器而不是使用缺省值服务器。要找出命令缺省使用的服务器，请使用 **ypwhich** 命令。

-d *Domain* 使您可以指定一个域而不是缺省域。要找出命令缺省使用的域，请使用 **domainname** 命令。

示例

1. 要察看在特定主机上的映射，输入:

   ```
   /usr/sbin/yppoll -h thor netgroups.byuser
   ```

   在本示例中，**yppoll** 命令显示位于名为 `thor` 的主机上的`netgroups.byuser` 映射的顺序号。
2. 要察看域上的映射，请输入:

   ```
   /usr/sbin/yppoll -d atlantis hosts.byname
   ```

   在该示例中，**ypoll** 命令显示位于 `atlantis` 域的 `hosts.byname` 映射的顺序号。

相关信息

domainname 命令、**ypwhich** 命令。

ypserv 守护程序。

AIX 5L Version 5.2 Network Information Services (NIS and NIS+) Guide 中的 Network Information Service (NIS) Overviewfor System Management。

NIS 参考。

---

yppush 命令

用途

提示网络信息服务（NIS）从属服务器复制已更新的 NIS 映射。

语法

/usr/sbin/yppush [**-v**] [**-d** *Domain*] *MapName*

描述

从 **/usr/etc/yp** 目录发出的 **yppush** 命令，提示网络信息服务 (NIS) 从属服务器复制已更新的 NIS 映射。*MapName* 变量指定要传输到主控服务器的从属服务器上的映射。要获取需要提示的服务器列表，**yppush** 命令读取 *Domain* 参数或当前缺省域指定的 **ypservers** 映射。在被提示后，每个从属服务器使用 **ypxfr** 命令复制并将映射传回自己的数据库。

可以使用系统管理接口工具（SMIT）运行该命令。要使用 SMIT，输入:

```
smit yppush
```

注: 如果系统使用版本 1 的 NIS 协议，**ypxfr** 命令就不是传输代理程序。

标志

-d *域*
指定一个域而不是缺省域。已指定域的映射必须存在。

-v
在服务器被呼叫时显示消息并显示每个服务器的响应条消息，如果您是用的版本 2 的协议的话。如果该标志被省略，命令只显示错误消息。
注：NIS 协议的版本 1 不显示消息。如果您的系统使用版本 1，使用**yppoll** 命令验证传输是否发生过。

示例

1. 要复制另一个域中的映射到从属服务器，输入：

   ```
   /usr/sbin/yppush -d atlantis netgroup
   ```

   在该示例中，**yppush** 命令从 `atlantis` 域复制 `netgroup` 映射。
2. 要在 **yppush** 命令调用每个从属服务器时显示命令的进行状态，输入：

   ```
   /usr/sbin/yppush -v -d atlantis netgroup
   ```

   在该示例中，**yppush** 命令从 `atlantis` 域将 `netgroup` 映射复制到每个网络从属服务器时显示进行状态消息。

文件

/var/yp/*DomainName***/ypservers.{dir, pag}**
列出 **yppush** 命令提示复制更新的 NIS 映射的服务器。

相关信息

yppoll 命令、**ypxfr** 命令。

ypserv 守护程序。

AIX 5L Version 5.2 System Management Concepts: Operating System and Devices 中的 System Management Interface Tool。

*《AIX 5L V5.2 系统管理指南：通信与网络》*中的『系统管理的网络文件系统（NFS）概述』和 *AIX 5L Version 5.2 Network Information Services (NIS and NIS+) Guide* 中的 NIS Maps。

AIX 5L Version 5.2 Network Information Services (NIS and NIS+) Guide 中的 Network Information Service (NIS) Overviewfor System Management。

NIS 参考。

---

ypserv 守护程序

用途

在本地网络信息服务（NIS）映射中查找信息。

语法

/usr/lib/netsvc/yp/ypserv

描述

ypserv 守护程序在其本地网络信息服务（NIS）映射中查找信息。**ypserv** 守护程序执行的操作是为实施者（通过 NIS 协议规范）和程序员（通过 **/usr/include/rpcsvc/yp_prot.h** 头文件）定义的。与 **ypserv** 守护程序的通信是通过远程过程调用（RPC）的方式实现的。

ypserv 守护程序只在服务器上运行。**ypserv** 守护程序通过以下系统资源控制器（SRC）命令启动和停止:

```
startsrc -s ypserv
stopsrc -s ypserv
```

ypserv 守护程序对 NIS 域内指定的映射执行以下操作:

Match	取一个键并返回相关值。
Get_first	返回映射中的首个键值对。
Get_next	列举映射中的下一个键值对。
Get_all	将整个 NIS 映射传输给请求者以作为单个 RPC 请求的响应。
Get_order_number	提供映射的信息而不是映射条目。顺序号实际上作为键值对存在于映射中，但服务器不会通过正常的查询功能返回该值。不过，如果用 **makedbm** 命令来检查映射则可以看到该键值对。
Get_master_name	提供映射的信息而不是映射条目。主控服务器实际上作为键值对存在于映射中，但服务器不会通过正常的查询功能返回该值。不过，如果用 **makedbm** 命令来检查映射则可以看到该键值对。

如果 **ypserv** 守护程序开始运行时该文件存在，则日志信息写入 **/var/yp/ypserv.log** 文件。

如果 **/var/yp/securenets** 文件存在，**ypservr** 命令只响应在该文件中指定的 ip 范围内的主机。

文件

/etc/rc.nfs	包含对 NFS 和 NIS 守护程序的启动脚本。
/var/yp/ypserv.log	包含 **ypserv** 守护程序的日志。

相关信息

chmaster 命令、**chslave** 命令、**domainname** 命令、**makedbm** 命令、**mkmaster** 命令、**mkslave** 命令、**ypcat** 命令、**ypinit** 命令、**ypmatch** 命令、**yppoll** 命令、**yppush** 命令、**ypset** 命令、**ypwhich** 命令、**ypxfr** 命令。

AIX 5L Version 5.2 System Management Concepts: Operating System and Devices 中的 System Resource Controller Overview。

*《AIX 5L V5.2 系统管理指南: 通信与网络》*中的『系统管理的网络文件系统（NFS）概述』和 *AIX 5L Version 5.2 Network Information Services (NIS and NIS+) Guide* 中的 NIS Maps。

AIX 5L Version 5.2 Communications Programming Concepts 中的 Remote Procedure Call Overview for Programming。

*《AIX 5L V5.2 系统管理指南: 通信与网络》*中的『如何配置 NIS』。

AIX 5L Version 5.2 Network Information Services (NIS and NIS+) Guide 中的 Network Information Service (NIS) Overviewfor System Management。

NIS 参考。

ypset 命令

用途

引导客户机和特定的服务器连接。

语法

/usr/sbin/ypset [**-V1**] [**-d** *Domain*] [**-h** *Host*] *Server*

描述

ypset 命令指导 **ypbind** 客户机上的守护程序连接到服务器上的 **ypserv** 守护程序。**ypbind** 守护程序会连接到 *Server* 参数指定的服务器，获取网络信息服务（NIS）服务用于 *Domain* 参数指定的域。**ypbind** 守护程序从服务器上的 **ypserv** 守护程序获取 NIS 服务。

在绑定设置后，直到客户机进程（例如 **ypcat** 命令或 **ypwhich** 命令）尝试获取对该域的绑定才会测试该绑定。如果绑定尝试失败（指定的服务器已经当机或没有运行 **ypserv** 守护程序），**ypbind** 守护程序会为相同的域进行另一次绑定尝试。

在 *Server* 参数中指定名称或因特网协议（IP）地址。如果指定名称，**ypset** 命令尝试通过使用 NIS 服务将名称解析给 IP 地址。只有在机器对请求的域拥有当前有效的绑定时这才会起作用。大多数情况下，应该用 IP 地址指定服务器。

在本地网络的多个主机提供 NIS 服务的情况下，**ypbind** 守护程序可以重新绑定到另一主机。如果服务器当机或没有运行 **ypserv** 守护程序，**ypbind** 守护程序将客户机重新绑定到另一服务器。在这种情况下，网络信息服务可以在可用的 NIS 服务器之间平衡负载。

使用 **ypset** 命令，如果网络:

- 不支持广播。
- 支持广播但没有 NIS 服务器。
- 访问只存在于特定 NIS 服务器上的映射。

标志

-d *Domain*	指定一个域而不是缺省域。
-h *Host*	为指定的主机上的而不是本地主机上的 **ypbind** 守护程序指定绑定。该主机可以用名称或 IP 地址指定。
-V1	绑定（旧）版本 1 的 NIS 协议的指定服务器。

示例

要设置服务器使其绑定到特定域中的主机上，请输入:

```
ypset -d ibm -h venus mars
```

在该示例中，**ypset** 命令导致名为 `venus` 的主机绑定到名为 `mars` 的服务器。

相关信息

domainname 命令、**ypcat** 命令、**ypwhich** 命令。

ypbind 守护程序、**ypserv** 守护程序。

*《AIX 5L V5.2 系统管理指南: 通信与网络》*中的『系统管理的网络文件系统（NFS）概述』。

AIX 5L Version 5.2 Network Information Services (NIS and NIS+) Guide 中的 How to Configure NIS。

AIX 5L Version 5.2 Network Information Services (NIS and NIS+) Guide 中的 Network Information Service (NIS) Overviewfor System Management。

NIS 参考。

---

ypupdated 守护程序

用途

更新网络信息服务（NIS）映射中的信息。

语法

/usr/lib/netsvc/yp/rpc.ypupdated [**-i** | **-s**]

描述

ypupdated 守护程序更新网络信息服务（NIS）映射中的信息。但是，在更新信息前，该守护程序会查询 **/var/yp** 目录中的 **updaters** 文件，决定哪个 NIS 映射应该被更新以及应当如何更改。

缺省情况下，**ypupdated** 守护程序需要可用的最安全的认证方法，或是 DES（安全）或是 UNIX（不安全）。

ypupdated 守护程序通过以下系统资源控制器（SRC）命令启动和停止:

```
startsrc -s ypupdated
stopsrc -s ypupdated
```

标志

-s 只接受用安全的远程过程调用（RPC）机制（AUTH_DES 认证）认证的调用。这会禁用 NIS 映射计划性的更新，除非网络支持这些调用。

-i 接受非安全的 AUTH_UNIX 凭证的 RPC 调用。这允许所有网络中的计划性 NIS 映射更新。

示例

要从命令行启动 **ypupdated** 守护程序，输入:

```
startsrc -s ypupdated
```

文件

/var/yp/updaters 用于更新 NIS 映射的制作文件。

相关信息

startsrc 命令。

keyserv 守护程序。

*《AIX 5L V5.2 系统管理指南: 操作系统与设备》*中的『系统资源控制器概述』。

AIX 5L Version 5.2 Communications Programming Concepts 中的 Remote Procedure Call Overview for Programming。

AIX 5L Version 5.2 Network Information Services (NIS and NIS+) Guide 中的 Network Information Service (NIS) Overviewfor System Management。

NIS 参考。

---

ypwhich 命令

用途

标识网络信息服务（NIS）服务器或作为给定映射的主控服务器的服务器。

语法

标识 NIS 服务器

/usr/bin/ypwhich [**-d** *Domain*] [**-V1** | **-V2**] [*HostName*]

标识映射的主控 NIS 服务器

/usr/bin/ypwhich [**-t**] [**-d** *Domain*] [**-m** [*MapName*]]

显示映射别名表

/usr/bin/ypwhich -x

描述

ypwhich 命令标识哪个服务器提供网络信息服务（NIS）服务或哪个服务器是映射的主控服务器，这取决于 **ypwhich** 命令如何调用。如果调用时没有自变量，该命令为本地机器显示 NIS 服务器名称。如果指定了主机名 *HostName*，系统就会向该主机查询以查找它使用的主控服务器。

标志

-d *Domain*	使用指定域而不是缺省域。
-V1	表明哪个服务器在为旧版本 1 的 NIS 协议客户机进程提供服务。
-V2	表明哪个服务器在为当前版本 2 的 NIS 协议客户机进程提供服务。如果没有指定任何版本，**ypwhich** 命令尝试定位提供版本 2 服务的服务器。如果当前没有版本 2 的服务器被绑定，那么 **ypwhich** 会命令尝试定位提供版本 1 服务的服务器。因为服务器和客户机都是反向兼容的，用户几乎很少需要关心当前在使用哪个版本。
-t	禁止别名转换，如果有一个映射名与别名相同，这十分有用。
-m *MapName*	查找映射的主控 NIS 服务器。不能用 **-m** 标志指定主机。*MapName* 变量可以是一个映射名或映射别名。当该映射名被省略时，**-m** 标志产生可用映射的列表。
-x	显示映射的别名表。这会列出这个命令知道的别名（*MapName*），并显示与每个别名相关联的映射名。

示例

1. 要查找某个映射的主控服务器，输入:

   ```
   ypwhich -m passwd
   ```

   在该示例中，**ypwhich** 命令显示 passwd 映射的服务器名。
2. 要查找名为 passwd 的映射而不是别名为 passwd 的映射，输入:

   ```
   ypwhich -t -m passwd
   ```

   在该示例中，**ypwhich** 命令显示全名为 passwd 映射的服务器名。
3. 要查找哪个服务器为运行旧版本 1 的 NIS 协议的客户机提供服务，输入:

   ```
   ypwhich -V1
   ```
4. 要显示映射别名表，输入:

   ```
   ypwhich -x
   ```

相关信息

ypset 命令。

ypserv 守护程序。

*《AIX 5L V5.2 系统管理指南: 通信与网络》*中的『系统管理的网络文件系统（NFS）概述』。

AIX 5L Version 5.2 Network Information Services (NIS and NIS+) Guide 中的 Network Information Service (NIS) Overviewfor System Management。

NIS 参考。

---

ypxfr 命令

用途

将网络信息服务（NIS）映射从 NIS 服务器转移到本地主机。

语法

/usr/sbin/ypxfr [**-f**] [**-c**] [**-d** *Domain*] [**-h** *Host*] [**-s** *Domain*] [**-C** *TID Program IPAddress Port*] [**-S**] *MapName*

描述

ypxfr 命令将网络信息服务（NIS）映射从 NIS 服务器转移到本地主机，如下所述:

1. 在客户机的 **/var/yp/***Domain* 目录中（该目录必须存在）创建一个临时映射。
2. 从服务器上获取映射条目并填入客户机上的映射，一次一个。
3. 获取并装入映射参数（顺序号和服务器）。
4. 删除任何旧版本的映射。
5. 将临时映射移到实际映射名。

如果 **/var/yp/securenets** 文件存在， **ypxfr** 命令只响应在该文件中列出的主机。

MapName 变量指定将从 NIS 服务器上转移的映射名。

如果交互地运行，**ypxfr** 命令将输出发送到用户终端。如果进行没有控制终端的调用，**ypxfr** 命令将它的输出追加到 **/var/yp/ypxfr.log** 文件（如果该文件已经存在）。该文件记录每次传输尝试及其结果。**ypxfr** 命令通常是从 root 用户的 **crontab** 文件或通过 **ypserv** 守护程序调用的。

要保持服务器间信息的一致性，使用 **ypxfr** 命令定期更新 NIS 数据库中的所有映射。不过，要注意有些映射比另外一些更改更频繁，因此需要更频繁地更新。例如，一些不频繁更改的映射，比如每几个月更改一次，应该至少每个月更新一次。经常更改的映射，比如一天几次，应该每小时检查更新。例如，**services.byname** 映射也许几个月都不会更改一次，而 **hosts.byname** 映射会一天更改几次。

要自动执行周期性更新，请使用 **crontab** 条目。要一次更新几个映射，可以在 shell 脚本中将几个命令组合起来用。shell 脚本的示例可以在 **/usr/etc/yp** 目录的以下文件中找到：**ypxfr_1perday**、**ypxfr_2perday**、**ypxfr_1perhour**。

可以使用系统管理接口工具（SMIT）运行该命令。要使用 SMIT，输入：

```
smit ypxfr
```

标志

-C *TID Program IPAddress Port*
告诉 **ypxfr** 命令在何处查找 **yppush** 命令。**ypserv** 守护程序调用 **ypxfr** 命令来将 **yppush** 命令回调到主机。使用参数表明以下信息：

TID 指定 **yppush** 命令的事务标识。

Program
指定与 **yppush** 命令关联的程序号。

IPAddress
指定 **yppush** 命令驻留的端口的因特网协议地址。

Port 指定 **yppush** 命令监听的端口。

注：该选项仅供 **ypserv** 守护程序使用。

-c
阻止发送请求 `Clear Current Map` 到本地 **ypserv** 守护程序。如果运行 **ypxfr** 命令时本地没有运行 **ypserv** 守护程序，请使用此标志。否则，**ypxfr** 命令显示错误消息，且转移失败。

-d *Domain*
指定一个域而不是缺省域。指定域的映射必须存在。

-f
强制转移，即使主控服务器上的版本没有本地版本新。

-h *Host*
从指定主机获取映射，而不管映射对主控服务器的报告。如果不指定主机，**ypxfr** 命令询问 NIS 服务以获取主控服务器的名称，并尝试从该处获取映射。*Host* 变量可以包含名称或格式为 `a.b.c.d` 的因特网地址。

-S
要求 **ypserv** 服务器，使用特权 IP 端口从该服务器获取要传输的映射。因为通常只允许 root 用户进程使用特权端口，该功能使得在传输时需要考虑安全性事项。如果被传输的映射是保密的映射，**ypxfr** 命令将映射的许可权设置为 `0600`。

-s *Domain*
指定一个源域，从该源域转移在域之间应当相同的映射（例如 **services.byname** 映射）。

示例

要从另一域中主机获取映射，输入：

```
/usr/sbin/ypxfr -d ibm -h venus passwd.byname
```

在该示例中，**ypxfr** 命令从 `ibm` 域中名为 `venus` 的主机获取 `passwd.byname` 映射。

文件

/var/yp/ypxfr.log	包含日志文件。
/usr/sbin/ypxfr_1perday	包含每天运行一次转移、和 **cron** 守护程序一起使用的脚本。
/usr/sbin/ypxfr_2perday	包含每天运行两次转移的脚本。
/usr/sbin/ypxfr_1perhour	包含对易变映射的每小时一次转移的脚本。

相关信息

crontab 命令、**yppush** 命令。

cron 守护程序、**ypserv** 守护程序。

AIX 5L Version 5.2 System Management Concepts: Operating System and Devices 中的 System Management Interface Tool。

*《AIX 5L V5.2 系统管理指南: 通信与网络》*中的『系统管理的网络文件系统（NFS）概述』和*AIX 5L Version 5.2 Network Information Services (NIS and NIS+) Guide*中的NIS 映射。

AIX 5L Version 5.2 Network Information Services (NIS and NIS+) Guide 中的 Network Information Service (NIS) Overviewfor System Management。

NIS 参考。

---

zcat 命令

用途

将压缩文件扩展到标准输出。

语法

zcat [**-n**] [**-V**] [*File* ...]

描述

zcat 命令允许用户扩展并查看压缩文件而不用将该文件解压。**zcat** 命令并不重命名扩展的文件，也不除去 **.Z** 的扩展名。**zcat** 命令将扩展输出写入标准输出。

标志

-n 从压缩文件省略压缩文件头。
-V 将当前版本和编译选项写入标准错误。

参数

File ... 指定要扩展的压缩文件。

返回值

如果有以下任意事件发生，**zcat** 命令以状态 1 退出:

- 输入文件不是由 **compress** 命令产生的。
- 不能读取输出文件或不能写入输出文件。

如果没有错误发生，出口状态为 0。

出口状态

0 成功完成。
>0 发生错误。

示例

要查看 `foo.Z` 文件而不将其解压，输入:

```
zcat foo.Z
```

`foo.Z` 文件的解压内容被写入标准输出。文件不重命名。

相关信息

compress 命令、**pack** 命令、**uncompress** 命令、**unpack** 命令。

*《AIX 5L V5.2 系统用户指南: 操作系统与设备》*中的『命令概述』。

---

zdump 命令

用途

转储时区信息。

语法

zdump [**-v**] [**-c** *CutOffYear*] *TimeZoneName* ...

描述

zdump 命令打印在命令行上指定的每个时区的当前时间。

标志

-c *CutOffYear* 在接近 *CutOffYear* 年的开始时停止详细输出。

-v 对每个在命令行列出的 TimeZoneName，打印以下信息:

- 当前时间。
- 最低可能时间值的时间。
- 最低可能时间值后一天的时间。
- 每次更改计算本地时间规则时的确切时间和一秒前的时间。
- 最高可能时间值的时间
- 最高可能时间值前一天的是时间。

如果给定时间是夏令时，每行以 `isdst=1` 结束，否则用 `isdst=0` 结束。

参数

TimeZoneName 表示显示时区信息的时区名。

出口状态

0 命令成功完成。

>0 发生错误。

示例

1. 要报告新加坡的时区信息，输入:

   ```
   zdump -v Singapore
   ```

2. 要报告直到 2035 年为止的土耳其的详细时区信息，输入:

   ```
   zdump -v -c 2035 Turkey
   ```

文件

/usr/sbin/zdump 包含 System V **zdump** 命令。
/usr/share/lib/zoneinfo 包含标准时区目录。

相关信息

zic 命令。

---

zic 命令

用途

提供时区编译器。

语法

zic [**-v**] [**-d** *Directory*] [**-l** *LocalTime*] [**-y** *YearIsType*] [*FileName* ...]

描述

zic 命令处理命令行中指定的文件中的文本并创建在此输入中指定的时间转换二进制文件。如果文件名指定为 -（连字号），则时区信息从标准输入中处理。

指定文件中的输入行由字段组成。字段分隔符是任意数目的空白字符。输入文件中的井号（#）指明从该井号出现一直延伸到这行的结束都为注释。如果用作字段的一部分，空白字符和井号都可用双引号（"）引起来。空行将被忽略。

规则行格式为:

```
Rule  NAME  FROM   TO   TYPE   IN   ON   AT   SAVE   LETTER/S
```

组成规则行的字段显示如下:

NAME 为应用此规则的规则集提供一个随机名称。

FROM 提供该规则应用的第一年。指定最小数（min）是用可表示的时间值表明最小年份。指定最大数（max）用可表示的时间值表明最大年份。

TO 提供该规则应用的最后一年。这通过有效的时间值或通过指定最小数和最大数表明。指定 `only` 用于重复 **FROM** 字段的值。

TYPE

提供规则应用的年份类型。

TYPE 字段有以下值:

'-' 该规则应用于所有在 **FROM** 和 **TO** 之间的年份。

uspres
该规则应用于美国总统的选举年。

nonpres
该规则应用于非美国总统的选举年。

如果 **TYPE** 字段有一个有别于上面提到的值， **zic** 命令运行 **/usr/sbin/yearistype year type** 命令来检查年份类型。

yearistype 命令接受两个参数；年份和年份类型。出口状态 0 就意味着年份是给定类型。否则，1 是作为出口状态返回。

IN 表示规则生效的月份。月份名也许是缩写。

ON 说明规则生效的天。可识别的格式包括:

- **lastFri** 代表该月的上个星期五。
- **lastMon** 代表该月的上个星期一。
- 代表该月的天数的数字。例如，5 代表该月的第 5 天。
- **lastSun** 代表该月的上个星期天。
- lastMon 代表该月的上个星期一。
- Sun>=8 代表 8 号或 8 号以后的第一个星期天。
- Sun<=25 代表在 25 号或 25 号前的最后一个星期天。

该周的天数的名称可以用缩写或全称。注意：在 **ON** 字段中不能有空格。

AT 表示规则生效当天的时间。可识别的格式包括:

- 代表用小时表示的时间的数字。比如，2 表示 2 小时。

- 2:00 表示 2 小时 0 分。
- 15:00 表示 24 小时格式时间的下午 3 点钟。
- 1:28:14 表示用小时、分钟、秒格式的 1 点 28 分 14 秒。

如果给定时间是本地 *wall-clock* 时间，则任何这些格式都用字母 w 引导；或如果给定时间是本地*标准*时间，则以 s 开头。如果缺少 w 或 s，就采用 wall-clock 时间。

在多于两种类型的本地时间的区域，需要在最早转换时间规则的 **AT** 字段使用本地的标准时间，以确保在结果时区二进制文件中存储的最早转换时间的准确性。

SAVE 代表在规则生效时添加到本地标准时间的时间总数。该字段的格式与 **AT** 字段一样。w 和 s 后缀在该字段中无效。

LETTER/S
提供时区缩写的*变量部分*，该时区缩写在规则生效时使用。当该字段包含 -（连字符）时，变量为空。S 字符用来表示 EST，而 D 字符用来表示 EDT。

区域行有以下格式:

```
Zone   NAME   GMTOFF   RULES/SAVE   FORMAT   [UNTIL]
```

构成区域行的字段为:

NAME 表明时区名称。这是用来创建该时区的时间转换信息文件的名称。

GMTOFF
表明添加到 GMT 以获取这个区域的标准时间的时间总数。该字段的格式与规则行 **AT** 和 **SAVE** 字段一样。如果时间必须从 GMT 中减去，可用负号作为该字段的开头。

RULES/SAVE
表明用于时区或添加到本地标准时间的时间总数的规则名称。如果该字段值是 -（连字符），那么该时区始终使用标准时间。

FORMAT
表明该时区中的时区缩写格式。%s 字符用来显示时区缩写的变量部分添加到哪个地方。

UNTIL 表明一个位置的 GMT 偏移量或规则更改的时间。要指定年、月、日和当天时间。如果这些都已指定，会从给定的 GMT 偏移量和规则更改中生成到指定时间为止的时区信息。

下一行必须是*连续*行。该连续行将前一行的 **UNTIL** 字段中指定的时间开始的信息放在前一行使用的文件中。除了 Zone 字符串和名字被省略，该行与区域行格式相同。正如区域行一样，连续行可以包含一个 **UNTIL** 字段，表明下一行是进一步的连续。

链接行有以下格式:

```
Link    LINK-FROM    LINK-TO
```

LINK-FROM 字段应该在区域行作为 **NAME** 字段出现；**LINK-TO** 字段用作该时区的备用名。

除连续行以外，输入行可以按任何顺序出现。

标志

-d *Directory*
在 *Directory* 目录而不是 **/usr/share/lib/zoneinfo/** 标准目录中创建时间转换信息文件。

-l *TimeZone*
使用 *TimeZone* 时区作为本地时间。**zic** 命令执行时就好像该文件包含与以下类似的链接:

```
Link   timezone  localtime
```

-v
如果在数据文件中显示的年份超出了系统时间值可表示的范围 （从 1970 年 1 月 1 日上午 GMT 0:00:00 到 2038 年 1 月 19 号上午 GMT 3:14:07），则提供一则消息。

-y *YearIsType*
在检查年份类型时使用给定的 **yearistype** 命令而不是 **/usr/sbin/yearistype** 命令。

参数

FileName
包含输入行的文件，该行指定要创建的时间转换信息文件。如果 *FileName* 是 **-**（连字符），那么读取标准输入。

示例

1. 规则行可以有以下格式:

```
Rule  USA  1970    max    -    Sep   Sun<=14 3:00   0     S
```

2. 时区行可以有以下格式:

```
Zone    Turkey          3:00    Turkey          EET%s
```

3. 链接行可以有以下格式:

```
Link    MET     CET
```

4. 要编译包含输入时区信息的 **timezone.infile** 文件并将该二进制文件放入标准时区 **/usr/share/lib/zoneinfo/** 目录，输入:

```
zic timezone.infile
```

5. 要编译包含输入时区信息的 **timezone.infile** 文件并将该二进制文件放入由 **-d** 选项指定的目录，输入:

```
 zic -d tzdir timezone.infile
```

6. 要在年份范围不正确时在编译时区输入文件时报告警告消息，输入:

```
zic -v timezone.infile
```

7. 要编译包含输入时区信息 **timezone.infile** 的文件并使用由 **-y** 标志指定的 **yearistype** 文件来检查年份类型，输入:

```
zic -y year timezone.infile
```

出口状态

0 命令成功完成。

>0 发生错误。

文件

/usr/sbin/yearistype
包含用来检查年份类型的 **yearistype** 命令。

/usr/sbin/zic
包含 System V **zic** 命令。

/usr/share/lib/zoneinfo
由**zic** 命令创建的文件使用的标准目录。

相关信息

zdump 命令。

附录 A. 对大于 2 吉字节的文件的命令支持

AIX 为大于 2 吉字节的文件提供支持，以便用户能在单个文件中存储大量数据。许多命令（但不是所有）支持使用大于 2 吉字节的文件。此外，有些命令在支持大型文件时存在限制。

不支持大于 2 吉字节的文件的命令

在许多情况下，不支持大型文件的命令不能使用任意大小的文件开始，诸如 **date**、**echo**、**nice**、**kill** 命令及其它命令。

此支持也不扩展到特定的系统控制文件，如 **/etc/passwd**、**/etc/inittab**、在 **/etc/security** 中的文件、系统记帐文件等。因此，仅使用这些系统文件的命令，如对管理员用户和系统安全的命令（**mkuser**、**su**）、系统记帐命令（**acctcom**、**prdaily**）以及常用系统控制命令（**init**、**penable**）都没有大型文件支持。

其它命令不支持大文件，是因为它们处理指定格式的文件，而格式定义为小于或等于 2 吉字节。这包括定义目标文件和可执行文件的 XCOFF 文件格式。定义 XCOFF 的文件头没有用来定义支持此大型文件的字段，并且系统不能装入这样大小的可执行文件。使用这些文件的命令，如 **ld**、**as**、**m4**、**strip** 等，都没有大型文件支持。

pack、**unpack** 和 **pcat** 命令的头格式没有足够的字符来存储超过 2 吉字节的文件。

其它文件格式也阻止文件大小超过 2 吉字节。这些包括一些格式上受工业标准限制的归档实用程序，如 **cpio**、**pax** 和 **tar**（可以用 **backup** 归档大型文件）命令，和限制 **ar** 命令的对象文件归档格式。

打印假脱机子系统在前端可以用来支持对大于 2 吉字节的文件的提交、操作和取消。但是，缺省的打印机后端，**piobe** 命令，不支持这样大小的文件。这表明大于 2 吉字节的打印作业可以发送到远程打印机或能处理这些大型文件的打印服务器，或者备用用户或供应商提供的能处理综合大型文件的备用后端。

注: 一项超过 2 吉字节的打印作业工作可能要花费好几天的时间来完成。

最后，有些命令中使用的用户文件按道理不应超出 2 吉字节。例如，尽管目录可以包含大型文件，目录文件本身不可以超过 2 吉字节。因此，诸如 **mkdir** 和 **rmdir** 这样的一些命令不支持大型目录。不需大文件支持的其它示例有使用 **wall** 命令来广播极大型文件的内容到所有终端，或者使用 **nroff** 命令在单个文件中处理超过 2 吉字节的写入文本。

支持大于 2 吉字节的文件命令

以下命令都能支持大于 2 吉字节的文件。在本列表中未列出的命令不支持大型文件。提供对大型文件有限支持的命令用星号（*）标记，对其限制的解释在列表后指出。

aclget	**auditcat ***	**auditconv ***	**auditselect ***
awk *	**backup**	**bdiff**	**bsh ***
cancel *	**cat**	**chgrp**	**chmod**
chown	**cksum**	**cmp**	**comm**
compress	**cp**	**csh ***	**csplit**
ctags	**cut**	**dd**	**devnm**
du	**egrep**	**enq ***	**expand**
fgrep	**file**	**find ***	**fold**

© Copyright IBM Corp. 1997, 2003

grep	**head**	**iconv**	**install**
join	**ksh** *	**link**	**ln**
lp *	**lpd** *	**lpq** *	**lpr** *
lprm *	**lpstat** *	**ls**	**make** *
move	**mv**	**nawk** *	**newform**
nl	**nohup**	**od**	**paste**
patch	**pr**	**proto**	**qcan** *
qchk *	**qdaemon** *	**qpri** *	**qprt** *
qstatus *	**rdist** *	**rdump**	**rembak** *
restore	**rev**	**rm**	**rrestore**
Rsh *	**sed**	**sort**	**split**
strings	**sum**	**tab**	**tail**
tee	**test**	**touch**	**tr**
trbsd	**tsh** *	**uncompress**	**unexpand**
uniq	**unlink**	**untab**	**update**
wc	**whereis**	**which**	**zcat**

限制

打印机命令仅在打印机前端支持大于 2 吉字节的文件。缺省的打印机后端，**piobe** 命令，不支持这样大小的文件。这对以下命令适用:

cancel	**lpq**	**qcan**	**qprt**
enq	**lpr**	**qchk**	**qstatus**
lp	**lprm**	**qdaemon**	**rembak**
lpd	**lpstat**	**qpri**	

shell 支持大于 2 吉字节的文件的 I/O 重定向。在 shell 中，不提供对大于 2 吉字节文件的其它支持。这适用于 **bsh**、**csh**、**ksh**、**Rsh** 和 **tsh** 命令。

> **注:** **sh** 命令是到 **ksh** 命令的链接。

awk 和 **nawk** 命令能处理大于 2 吉字节的数据文件。但是，**awk** 和 **nawk** 脚本本身不能这么大。

find 命令能处理大于 2 吉字节的文件，但是不允许使用 *Number* 大于 2 吉字节的 **-size***Number* 标志。

make 命令能操作大于 2 吉字节的目标和相关的文件，但是 Makefile 文件本身不能这么大。

审计命令 **auditcat**、**auditconv** 和 **auditselect** 支持大于 2 吉字节的 **trail** 文件，但是它们不支持大于 2 吉字节的 **bin** 文件。

> **注意:** 不要尝试使用 **rdist** 命令发送大型文件给 AIX 4.2 以前的版本或无操作系统机器。这样做会导致未定义的行为，少数情况下会丢失数据。

附录 B. 命令的功能列表

本附录按功能列出命令。

- 通信
  - 异步终端仿真
  - 基本联网实用程序
  - 常规通信工具
  - 邮件工具
  - 消息处理程序
  - 网络文件系统
  - 网络信息服务
  - 网络管理
  - STREAMS
  - 传输控制协议 / 网际协议
  - 网络计算系统（NCS）
- 文件和目录
  - 目录
  - 编辑器
  - 文件
  - 文件内容
  - 文本格式化
  - 文本格式化宏软件包
- 常规操作
  - 设备和终端
  - 文档和培训
  - 文件系统
  - 游戏
  - iFOR/LS
  - 逻辑卷
  - 网络安装管理（NIM）
  - 数值数据
  - 性能调整
  - 进程和命令
  - 队列
  - 屏幕输出
  - 安全性和系统访问
  - shell
  - 系统记帐和统计信息
  - acct/* 命令

© Copyright IBM Corp. 1997, 2003

- 系统资源
- 软件安装
- 用户界面
- 宏

• 编程工具
- 调试器
- 消息
- 源程序
- 目标文件
- 杂类语言
- C 语言工具
- 汇编程序和编译器
- 对象数据管理器（ODM）

通信

命令列表: 异步终端仿真

ate
启动异步终端仿真（ATE）程序。

xmodem
用 **xmodem** 协议传输文件，它在异步传输期间检测数据传输错误。

命令列表: 基本联网实用程序

ct
拨号附加终端并发出登录进程。

cu
直接或间接连接到另一系统。

cut
从文件每一行写出选定的字节、字符或字段。

rmail
处理通过基本联网实用程序（BNU）接收到的远程邮件。

tip
连接到远程系统。

uucheck
检查 BNU 要求的文件和目录。

uucico
将基本联网实用程序（BNU）的命令、数据和执行文件传输到远程系统。

uuclean
从 BNU 假脱机目录除去文件。

uucleanup
从基本联网实用程序（BNU）的假脱机目录里删除选定的文件。

uucp
从一个操作系统中复制文件到另一个系统。

uucpadm
输入基本 BNU 配置信息。

uucpd

处理 BNU 和 TCP/IP 之间的通信。

uudecode

为使用电子邮件传输的二进制文件编码或解码。

uudemon.admin

提供关于 BNU 文件传输状态的阶段性信息。

uudemon.cleanu

清除 BNU 假脱机目录和日志文件。

uudemon.hour

使用 BNU 程序启动到远程系统的文件传输调用。

uudemon.poll

轮询在 BNU 的 **Poll** 文件中列出的系统。

uuid_gen

为对象、类型和接口生成“通用唯一标识符（UUID）”。

uuencode

为使用电子邮件传输的二进制文件编码或解码。

uukick

使用调试方式访问一个指定的远程系统。

uulog

提供关于一个系统中 BNU 文件传输活动的信息。

uuname

提供关于本地系统能访问的其它系统的信息。

uupick

完成传输并处理 **uuto** 命令所发出的文件。

uupoll

强制对远程 BNU 系统轮询。

uuq

显示 BNU 作业队列，从队列中删除指定的作业。

uusched

基本联网实用程序（BNU）文件传输程序的调度工作。

uusend

向远程主机发送文件。

uusnap

显示 BNU 与远程系统的联系状态。

uustat

报告 BNU 操作的状态并提供对 BNU 操作的有限控制。

uuto

从一个系统复制文件到另一系统。

uutry

在打开调试的情况下联系指定的远程系统，并允许用户覆盖缺省的重试时间。

Uutry

在打开调试的情况下，联系指定的远程系统，并将调试输出保存到临时文件。

uux

在另一个基于 UNIX 的系统上运行命令。

uuxqt

执行基本联网实用程序（BNU）的远端命令请求。

命令列表：常规通信工具

connect

连接到远程计算机。

enroll

	设置用于实现安全通信通道的密码。
getty	设置端口的特征。
mesg	允许或拒绝写入消息。
no	配置网络选项。
pdelay	启用或报告延迟登录端口的可用性。
pdisable	禁用登录端口。
penable	启用或者报告登录端口的可用性。
phold	禁用或者报告保留的登录端口的可用性。
pshare	启用或者报告共享登录端口的可用性。
rdist	维护多个主机上的相同文件副本。
rdump	在远程机器的设备上备份文件。
wall	给所有登录的用户写一条消息。
write	给系统上的其他用户发送信息。
writesrv	允许用户发送消息到远程系统和从远程系统接收消息。

命令列表: 邮件工具

bellmail	给系统用户发送消息并显示来自系统用户的消息。
bffcreate	用备份格式创建安装映象文件。
biff	在当前会话期间启用或禁用邮件通知。
comsat	向用户通知进入的邮件。
from	确定邮件发出人。
imapd	启动因特网消息访问协议（IMAP）服务器进程。
Mail 或 **mail**	发送和接收邮件。
mailq	打印邮件队列的内容。
mailstats	显示邮件流量的统计信息。
mailx	发送和接收邮件。
newaliases	从 **/etc/aliases** 文件构建别名数据库的新副本。
pop3d	启动邮局协议版本 3（POP3）服务器进程。

rmail

处理通过基本联网实用程序（BNU）接收到的远程邮件。

sendmail

为本地或网络传送路由邮件。

smdemon.cleanu

定期内务处理，清除 **sendmail** 队列。

xget

通过安全的通信通道接收保密邮件。

xsend

通过安全的通信通道发送保密邮件。

命令列表：消息处理程序

ali

列出邮件别名及其地址。

anno

注释消息。

ap

对地址进行语法分析并重定格式。

bugfiler

将程序错误报告自动存储到指定的邮件目录。

burst

将摘要分解成消息。

comp

编写消息。

conflict

搜索别名以及密码冲突。

dist

将消息重新分布到附加地址。

dp

对日期进行分析并重定格式。

folder

选择、列出文件夹和消息。

folders

列出邮件目录中的所有文件夹和消息。

forw

转发消息。

inc

将新邮件合并到一个文件夹。

install_mh

设置邮箱目录。

mark

创建、修改和显示消息序列。

mhl

制作消息的格式化清单。

mhmail

发送或接收邮件。

mhpath

打印文件夹和消息的全路径名。

msgchk

检查消息。

msh

创建一个 MH 外壳。

next

显示下一条消息。

packf

将文件夹中的内容压缩到一个文件。

pick

根据内容选择消息，并创建和修改序列。

post

路由消息。

prev

显示上一条消息。

prompter

调用提示编辑器。

rcvdist

将进入消息的副本发送到附加收件人。

rcvpack

在压缩文件中保存进入消息。

rcvstore

将来自标准输入的新邮件合并放入一个文件夹中。

rcvtty

向用户通知进入消息。

refile

在文件夹间移动文件。

repl

应答消息。

rmf

除去文件夹以及它们所包含的消息。

rmm

除去消息活动状态。

scan

产生一个每行一条消息的扫描清单。

send

发送消息。

sendbug

把系统错误报告邮寄到指定的地址。

show

显示消息。

slocal

处理进入邮件。

sortm

对消息进行排序。

spost

路由消息。

vmh

为和 MH 命令一起使用调用可视界面。

whatnow

为草稿处理调用提示界面。

whom

处理消息处理程序（MH）地址。

命令列表: 网络文件系统

automount

自动安装 NFS 文件系统。

biod

	处理文件的客户机请求。
bootparamd	为无盘客户机提供引导信息。
chnfs	更改系统的配置以调用指定数目的 **biod** 和 **nfsd** 守护程序。
chnfsexp	更改用于将目录导出到 NFS 客户机的选项。
chnfsmnt	更改用于从 NFS 服务器安装目录的选项。
exportfs	将目录导出和取消导出到 NFS 客户机。
lockd	处理锁定请求。
mknfs	配置系统文件以运行 NFS。
mknfsexp	将目录导出到 NFS 客户机。
mknfsmnt	从 NFS 服务器安装目录。
mountd	应答来自客户机的安装文件系统请求。
nfsd	启动客户机的系统操作请求。
nfso	配置网络文件系统（NFS）网络选项。
nfsstat	显示关于网络文件系统（NFS）和远程过程调用（RPC）的统计信息。
on	在远程系统上执行命令。
portmap	将 RPC 程序号转换成因特网端口号。
rexd	对远程机器执行程序。
rmnfs	更改系统配置以停止调用 NFS 守护程序。
rmnfsexp	取消从 NFS 客户机导出目录。
rmnfsmnt	除去 NFS 安装。
rpcgen	生成 C 代码来实现 RPC 协议。
rpcinfo	报告远程过程调用（RPC）服务器的状态。
rpc.pcnfsd	处理来自 PC-NFS（个人计算机网络文件系统）客户机的服务请求。
rstatd	返回从内核获取的性能统计信息。
rup	显示在本地网络上远程主机的状态。
rusers	报告登录到远程机器的用户列表。
rusersd	响应来自 **rusers** 命令的查询。
rwall	

rwalld

给网络上的所有用户发送信息。

showmount

处理来自 **rwall** 命令的请求。

spray

显示有远程已安装文件系统的所有客户机的列表。

sprayd

向主机发送指定数量的数据包并报告性能统计信息。

statd

接收 **spray** 命令发送的包。

为 NFS 上的锁定服务提供崩溃和恢复功能。

命令列表: 网络信息服务

chkey

更改加密密钥。

chmaster

执行 **ypinit** 命令并重新启动 NIS 守护程序以更改主控服务器。

chslave

重新执行 **ypinit** 命令以检索来自主控服务器的映射，并重新启动 **ypserv** 守护程序序以更改从属服务器。

chypdom

更改系统的当前域名。

domainname

显示或设置当前 NIS 域的名称。

keyenvoy

充当用户进程和 **keyserv** 守护程序之间的中介。

keylogin

解密并存储用户的密钥。

keyserv

存储公用和专用密钥。

lsmaster

显示 NIS 主控服务器的配置的特征。

lsnfsexp

显示用网络文件系统（NFS）导出的目录的特征。

lsnfsmnt

显示 NFS 可安装文件系统的特征。

makedbm

制作网络信息服务（NIS）映射。

mkclient

取消注释 **/etc/rc.nfs**文件中 **ypbind** 守护程序的条目，并启动 **ypbind** 守护程序以配置客户机。

mkkeyserv

取消注释 **/etc/rc.nfs** 文件中 **keyserv** 守护程序的条目，并使用 startsrc 命令调用该守护程序。

mkmaster

调用 **ypinit** 命令并启动 NIS 守护程序以配置主控服务器。

mkslave

执行 **ypinit** 命令以检索来自 NIS 主控服务器的映射，并启动 **ypserv** 守护程序序以更改从属服务器。

mk_niscachemgr

取消注释 **/etc/rc.nfs** 文件中 **nis_cachemgr** 守护程序的条目，并使用 **startsrc** 命令调用该守护程序。

mk_nisd

取消注释 **/etc/rc.nfs** 文件中 **rpc.nisd** 守护程序的条目，并使用 **startsrc** 命令调用该守护程序。

mk_nispasswdd

取消注释 **/etc/rc.nfs** 文件中 **rpc.nispasswdd** 守护程序的条目，并使用 **startsrc** 命令调用该守护程序。

newkey

在 **/etc/publickey** 文件中创建新密钥。

nis_cachemgr

启动 NIS+ 高速缓存管理器守护程序。

nisaddcred

创建 NIS+ 凭证信息。

nisaddent

从相应的 **/etc** 文件或 NIS 映射创建 NIS+ 表。

niscat

显示 NIS+ 表的内容。

nischgrp

更改 NIS+ 对象的组所有者。

nischmod

更改 NIS+ 对象的访问权。

nischown

更改一个或多个 NIS+ 对象或条目的所有者。

nisclient

为 NIS+ 主体初始化 NIS+ 凭证。

nisdefaults

显示当前在名称空间中活动的七个缺省值。

niserror

显示 NIS+ 错误消息。

nisgrep

用于搜索 NIS+ 表的实用程序。

nisgrpadm

在 NIS+ 组中创建、删除以及执行杂项管理操作。

nisinit

初始化工作站为 NIS+ 客户机。

nisln

在 NIS+ 对象和表条目之间创建符号链接。

nislog

nislog 命令显示事务日志的内容。

nisls

列出 NIS+ 目录内容。

nismatch

用于搜索 NIS+ 表的实用程序。

nismkdir

创建非根 NIS+ 目录。

nismkuser

创建新的 NIS+ 用户帐户。

nispopulate

在 NIS+ 域中植入 NIS+ 表。

nisrm

从名称空间除去 NIS+ 对象。

nisrmdir

从名称空间除去 NIS+ 对象。

nisrmuser

除去 NIS+ 用户帐户。

nisserver

设置 NIS+ 服务器。

nissetup	初始化 NIS+ 域。
nisshowcache	打印出共享高速缓存文件的内容。
nisstat	报告 NIS+ 服务器统计信息。
nistbladm	管理 NIS+ 表。
nistest	使用条件表达式返回 NIS+ 名称空间的状态。
nisupdkeys	更新 NIS 目录对象中的公用密钥。
revnetgroup	反转在 NIS 映射中的网络组文件中的用户和主机清单。
rm_niscachemgr	停止 **rpc.nisd** 守护程序并注释在 **/etc/rc.nfs** 文件中的条目。
rm_nisd	停止 **nis_cachemgr** 守护程序并注释在 **/etc/rc.nfs** 文件中的条目。
rm_nispasswdd	停止 **rpc.nispasswdd** 守护程序并注释在 **/etc/rc.nfs** 文件中的条目。
rmkeyserv	停止 **keyserv** 守护程序并注释 **keyserv** 守护程序在 **/etc/rc.nfs**文件中的条目。
rmyp	除去 NIS 的配置。
rpc.nispasswd	NIS+ 密码更新守护程序。
ypbind	使客户机进程能绑定或连接到 NIS 服务器。
ypcat	打印出 NIS 映射。
ypinit	在 NIS 服务器中设置 NIS 映射。
ypmatch	显示 NIS 映射中的给定键的值。
yppasswd	更改 NIS 中的网络密码。
yppasswdd	接收和执行来自 **yppasswd** 命令的请求。
yppoll	显示当前在服务器中使用的 NIS 映射的顺序号（标识号）。
yppush	提示 NIS 从属服务器复制更新的 NIS 映射。
ypserv	在本地 NIS 映射中查找信息。
ypset	将客户机引导到特定的服务器。
ypupdated	更新 NIS 映射中的信息。
ypwhich	标识 NIS 服务器或作为给定映射的主控服务器的服务器。
ypxfr	将 NIS 映射从 NIS 服务器传输到本地主机。

命令列表: 网络管理

mosy

将管理信息的结构和标识（SMI）和管理信息库（MIB）模块的 ASN.1 定义转换成用于 **snmpinfo** 命令的对象定义文件。

snmpd

启动简单网络管理协议（SNMP）代理守护程序作为后台处理。

snmpinfo

请求或修改简单网络管理协议（SNMP）代理程序管理的管理信息库（MIB）变量的值。

命令列表: STREAMS

autopush

配置自动推进的 STREAMS 模块的列表。

scls

产生模块和驱动程序名的列表。

strace

打印 STREAMS 跟踪消息。

strchg

更改流配置。

strload

装入和配置便携式流环境（PSE)。

strconf

查询流配置。

strclean

清空 STREAMS 错误记录器。

strerr（守护程序）

接收来自 STREAMS 日志驱动程序的错误日志消息。

命令列表: 传输控制协议 / 网际协议

arp

显示和修改地址解析。

chnamsv

更改主机上的基于 TCP/IP 的名称服务配置。

chprtsv

在客户机或服务器上更改打印服务配置。

f

显示用户信息。

finger

显示用户信息。

fingerd

为 **finger** 命令提供服务器功能。

ftp

在本地和远程主机之间传输文件。

ftpd

为因特网 FTP 协议提供服务器功能。

gated

为 RIP、EGP、HELLO 和 SNMP 协议提供网关路由功能。

gettable

从主机获取 NIS 格式的主机表。

host

hostent	将主机名解析成因特网地址或将因特网地址解析成主机名。
hostid	在系统配置数据库中直接控制地址映射条目。
hostname	设置或显示当前本地主机的标识符。
htable	设置或显示当前主机系统的名称。
ifconfig	将主机文件转换成被网络库例程使用的格式。
inetd	为正在使用 TCP/IP 的网络配置或显示接口参数。
ipreport	为网络提供因特网服务管理。
iptrace	从指定的包跟踪文件中生成包跟踪报告。
lpd	提供因特网协议的接口级包跟踪。
lsnamsv	在网络中提供远程打印服务器。
lsprtsv	显示存储在数据库中的名称服务信息。
mkhosts	显示存储在数据库中的打印服务信息。
mknamsv	生成主机表文件。
mkprtsv	为客户机配置主机上的基于 TCP/IP 的名称服务。
mktcpip	配置主机上的基于 TCP/IP 的打印服务。
named	为在主机上启动 TCP/IP 设置必需值。
namerslv	为域名协议提供服务器功能。
netstat	为系统配置数据库中的本地解析器例程直接操作域名服务器条目。
nslookup	显示网络状态。
ping	查询因特网域名服务器。
rcp	发送一个回送信号请求给网络主机。
remsh	在本地主机和远程主机之间或者在两个远程主机之间传输文件。
rexec	在远程主机执行指定命令或登录到远程主机。
rexecd	在远程主机上一次执行一个命令。
rlogin	为 **rexec** 命令提供服务器功能。
rlogind	用远程主机连接本地主机。
rmnamsv	为 **rlogin** 命令提供服务器功能。
	取消配置主机上的基于 TCP/IP 的名称服务。

rmprtsv

在客户机和服务器上取消配置打印服务。

route

手工控制路由表。

routed

管理网络路由表。

rsh

在远程主机执行指定命令或登录到远程主机。

rshd

为远程命令执行提供服务器功能。

ruptime

显示网络上每个主机的状态。

ruser

直接控制三个独立系统数据库中控制外来主机访问程序的条目。

rwho

显示哪些用户登录到本地网络上的主机。

rwhod

为 **rwho** 和 **ruptime** 命令提供服务器功能。

securetcpip

启用操作系统网络安全功能。

setclock

为网络中的主机设置时间和日期。

slattach

附加串行线路作为网络接口。

sliplogin

配置标准输入终端线路作为到远程主机的“串行线路网际协议”（SLIP）链接。

talk

与另一个用户对话。

talkd

为 **talk** 命令提供服务器功能。

tcpdump

打印包头。

telinit

初始化并控制进程。

telnet

使用 TELNET 接口将本地主机连接到远程主机。

telnetd

为 TELNET 协议提供服务器功能。

tftp

使用次要文件传输协议议（TFTP）在主机间传输文件。

tftpd

为次要文件传输协议提供服务器功能。

timed

在系统启动时调用时间服务器守护程序。

timedc

返回关于 **timed** 守护程序的信息。

tn

使用 TELNET 接口将本地主机连接到远程主机。

tn3270

使用 TELNET 接口将本地主机连接到远程主机。

traceroute

打印 IP 包至某个网络主机的路由。

trpt

执行 TCP 套接字的协议跟踪。

utftp
使用次要文件传输协议（TFTP）在主机间传输文件。

命令列表: 网络计算系统（NCS）

lb_admin
监视和管理位置代理注册。

llbd
管理局部位置代理数据库中的信息。

nrglbd
管理全局位置代理数据库。

文件和目录

命令列表: 目录

cd
更改当前目录。

chgrp
更改文件或目录的组所有权。

chmod
更改许可权方式。

chroot
更改命令的根目录。

delete
除去（取消链接）文件或目录。

dircmp
比较两个目录以及它们公共文件的内容。

dirname
除了指定路径的最后部分，全都写入标准输出。

dosdir
列出 DOS 文件目录。

fdformat
格式化软盘。

ls
显示目录的内容。

mkdir
创建一个或多个新目录。

mvdir
移动（重命名）一个目录

pathchk
检查路径名。

pwd
显示工作目录的路径名。

rm
除去（取消链接）文件或目录。

rmdir
除去目录。

which_fileset
搜索 **/usr/lpp/bos/AIX_file_list** 文件，查找指定文件名或命令。

命令表列: 编辑器

ctags
制作标记文件以帮助定位源文件中的对象。

ed
按行编辑文本。

edit
为新用户提供简单的行编辑器。

ex
以屏幕显示交互式编辑行。

red
按行编辑文本。

sed
提供流编辑器。

tvi
以全屏显示方式提供可信编辑器。

vedit
以全屏显示方式编辑文件。

vi
以全屏显示方式编辑文件。

view
以只读方式启动 vi 编辑器。

命令列表: 文件

ar
维护链接编辑器使用的索引库。

backup
备份文件和文件系统。

cat
连接或显示文件。

chgrp
更改文件或目录的组所有权。

chlang
更改 **/etc/environment** 文件中的语言（LANG）环境变量。

chmod
更改许可权方式。

chtz
更改 **/etc/environment** 文件中的语言（TZ）环境变量。

cksum
更改文件的校验和以及字节计数。

copy
复制文件。

cp
复制文件。

cpio
将文件复制到归档存储器和目录，以及从归档存储器和目录复制文件。

dd
转换和复制文件。

defragfs
增加文件系统的邻近可用空间。

delete
除去（取消链接）文件或目录。

dosdel

删除 DOS 文件。

dosread

复制 DOS 文件。

doswrite

将所在操作系统文件复制到 DOS 文件。

file

确定文件类型。

find

用匹配表达式查找文件。

link

执行 **link** 子例程。

ln

链接文件。

mv

移动文件

nulladm

创建指定的文件，授予文件所有者和组读写许可权，授予其它用户读许可权。

pax

抽取、写以及列出归档文件成员；复制文件和目录层次结构。

pg

将文件格式化以供显示。

restore

从本地设备复制先前由 **backup** 命令创建的备份文件系统或文件。

rm

除去（取消链接）文件或目录。

rmvfs

除去 **/etc/vfs** 文件中的条目。

split

将文件分割成几个部分。

sum

显示文件的校验和以及块计数。

tar

处理归档。

tee

显示程序的输出并将其复制到一个文件中。

touch

更新文件的访问时间与修改时间。

umask

显示或设置文件方式创建掩码。

unlink

执行 **unlink** 子例程。

命令列表：文件内容

awk

用文件匹配模式查找行，然后对它们执行指定的操作。

bdiff

使用 **diff** 命令来查找特大文件中的差异。

bfs

扫描文件。

cmp

比较两个文件。

colrm

从文件中抽取列。

comm

选择或拒绝两个排序文件的公共行。

comp

编写消息。

compress

压缩和扩展数据。

csplit

按上下文分割文件。

cut

从文件每一行写出所选的字节、字符或字段。

diff

比较文本文件。

diff3

比较三个文件。

dircmp

比较两个目录以及它们公共文件的内容。

egrep

在文件中搜索一种模式。

expand

将制表符更改成空格，写入标准输出。

fgrep

在文件中搜索一个文字串。

fold

由于输出设备限定宽度，折叠长行。

genxlt

为了利于 **lconv** 库使用，生成代码集转换表。

grep

在文件中搜索一种模式。

head

显示一或多个文件的开始几行或几字节。

iconv

将字符的编码从一种代码页编码方案转换为另一种。

join

结合两个文件的数据字段。

localedef

为产生语言环境数据库，处理语言环境和字符映射文件。

look

在已排序文件中查找行。

more

在显示屏幕每次一屏显示连续的文本。

paste

合并几个文件的行或一个文件中的后续行。

pcat

解压缩文件并将它们写入标准输出。

pack

压缩文件。

page

在显示屏幕每次一屏显示连续的文本。

rev

逆转文件每一行中的字符。

sdiff

比较两个文件并以并排的格式显示差别。

sort

对文件进行排序、合并已排序文件和检查文件以确定它们是否已排序。

spell

查找英语拼写错误。

spellin

创建拼写列表。

spellout

验证单词不在拼写列表中。

tab

将空格转换成制表符。

tail

从指定点开始将文件写入标准输出。

tr

翻译字符。

trbsd

翻译字符（BSD 版）。

tsort

为有序对的无序列表排序（一种拓扑排序）。

uncompress

压缩和扩展数据。

unexpand

和恢复的制表符一起写入标准输出。

uniq

删除文件中重复的行。

unpack

扩展文件。

untab

将制表符转换成空格。

wc

计算文件中的行数、字数和字节数。

what

显示文件中的标识信息。

zcat

压缩和扩展数据。

命令列表: 文本格式化

addbib

创建或扩展文献目录数据库。

apropos

用关键词查找定位命令。

canonls

为 Canon LASER SHOT 用 LIPS III 方式处理 **troff** 命令输出。

catman

为手册创建 cat 文件。

checkcw

为 **troff** 命令准备定宽文本。

checkeq

检查用备忘录宏格式过的文档。

checkmm

检查用备忘录宏格式过的文档。

checknr

检查 **nroff** 和 **troff** 文件。

col

为具有逆向换行和正向／逆向半换行的标准输出文本而设置的过滤器。

colcrt

为 CRT 预览过滤 **nroff** 命令输出。

cw

为 **troff** 命令准备定宽文本。

deroff

从文件里除去 **nroff**、**troff**、**tbl** 和 **eqn** 命令构造。

diction

突出显示不明了或冗长的句子。

diffmk

标出文件之间的差异。

enscript

为了便于打印，把文本文件转换成 PostScript 格式。

eqn

为 **troff** 命令格式化数学文本。

expand

将制表符更改成空格，写入标准输出。

explain

提供交互式同义词典。

fmt

在发送之前格式化邮件消息。

grap

对 **pic** 命令将要处理的图进行排版。

greek

将来自电传打字机 37 工作站的英语输出转换为适用于其它工作站的输出。

hp

为 HP2640 和 HP2621 系列的终端处理特殊功能。

hplj

为 HP Laser Jet Series 打印机对 **troff** 命令输出进行后处理。

hyphen

查找加连字符的单词。

ibm3812

为 3818 页式打印机和 3812 2型页式打印机对 **troff** 命令输出进行后处理。

ibm3816

为 3816 页式打印机和 3812 2型页式打印机对 **troff** 命令输出进行后处理。

ibm5587G

为安装（32x32/24x24）墨粉盒的 5587G 打印机对 **troff**命令输出进行后处理。

indxbib

给文献目录构建一个逆向索引。

lookbib

查找文献目录的参考。

macref

产生宏文件的交叉引用清单。

makedev

创建适用于 **troff** 命令及其预处理器阅读的二进制描述文件。

managefonts

为了更新或更改系统中已安装的字体系列集合，给用户提供一个基于菜单的简单界面。

mant

对帮助页进行排版。

mm	打印用备忘录宏格式过的文档。
mmt	对文档进行排版。
mvt	对英语视图和幻灯片进行排版。
ndx	为文件生成一个主题页面索引。
neqn	为 **nroff** 命令格式化数学文本。
newform	更改文本文件的格式。
nl	计算文件中的行数。
nroff	为在类打字机设备和行式打印机上打印格式化文本。
pic	为绘图预处理 **troff** 命令的输入。
proff	为带个人打印机数据流的打印机格式化文本。
ps630	将 Diablo 630 打印文件转换为 PostScript 格式。
ps4014	将 Tektronix 4014 文件转换为 PostScript 格式。
psc	将 **troff** 中间格式转换成 PostScript 格式。
psdit	将 **troff** 中间格式转换成 PostScript 格式。
psplot	将绘图格式的文件转换成 PostScript 格式。
psrev	将 PostScript 文件的页面顺序逆转，并选择进行打印的页面范围。
psroff	将文件从 **troff** 格式转换成 PostScript 格式。
ptx	生成一个置换索引。
refer	在文档中查找并插入文献参考。
roffbib	打印文献目录数据库。
soelim	在 **nroff** 命令文件中处理 **.so** 请求。
sortbib	对文献目录数据库排序。
spell	查找英语拼写错误。
spellin	创建拼写列表。
spellout	验证单词不在拼写列表中。
style	分析文档的表面特征。
subj	从文档生成一个主题列表。
tbl	

tc

为 **nroff** 和 **troff** 命令格式化表。

troff

为 Tektronix 4015 系统解释 **troff** 命令输出中的文本。

ul

为在排版设备上打印而对文本进行格式化。

vgrind

加下划线。

xpreview

格式化易于阅读的程序清单。

在 X 显示器上显示 **troff** 文件。

文本格式化宏软件包

man

为帮助页提供格式化工具。

me

用不同样式为创建技术论文提供格式化工具。

mm

为商业文档，如备忘录、信函和报告，提供格式化工具。

mptx

格式化由 **ptx** 命令产生的置换索引。

ms

为各种样式的文章、论文和书籍提供格式化工具。

mv

简化对视图及投影幻灯片的排版。

常规操作

命令列表：设备和终端

adfutil

仅为有配置数据库的 PS/2 适配器通过 AIX 5.1 提供将“微通道”信息合并的能力。

bterm

以双向总线（BIDI）方式仿真终端。

cancel

取消对行式打印机的请求。

captoinfo

将 termcap 文件转换成 terminfo 描述符文件。

cfgmgr

通过运行配置规则对象类中指定的程序配置设备。

chcons

重定向系统控制台到指定设备或文件，在系统的下一次启动时生效。

chdev

更改设备的特征。

chdisp

更改低功能终端（LFT）子系统使用的显示器。

chfont

更改显示器的缺省字体。

chkbd

更改在系统启动时高功能终端子系统使用的缺省键盘映射。

clear

清除终端屏幕。

devnm

命名设备。

diag

执行硬件问题确定。

digest

将 ASCII 格式的 **/etc/qconfig** 文件转换成 **/etc/qconfig.bin** 文件，该文件是一个由 **qdaemon** 命令使用的队列配置的二进制版本。

dscreen

启动“动态屏幕”实用程序。

enable

启用打印机队列

fdformat

格式化软盘。

flcopy

复制到软盘或从软盘复制。

fold

由于输出设备限定宽度，折叠长行。

format

格式化软盘。

getty

设置端口的特征。

hplj

为 HP Laser Jet Series 打印机对 **troff** 命令输出进行后处理。

ibm3812

为 3816 页式打印机和 3812 2型页式打印机对 **troff** 命令输出进行后处理。

ibm3816

为 3816 页式打印机和 3812 2型页式打印机对 **troff** 命令输出进行后处理。

ibm5587G

为安装（32x32/24x24）墨粉盒的 5587G 打印机对 **troff**命令输出进行后处理。

iconv

将字符的编码从一种代码页编码方案转换为另一种。

infocmp

管理 **terminfo** 描述。

iostat

报告中央处理器（CPU）的统计信息和 tty、磁盘、CD-ROM 的输入／输出统计信息。

keycomp

把键盘映射表文件编译成一个输入方法键盘映射文件。

lp

向行式打印机发送请求。

lpr

将打印作业列队。

lpstat

显示行式打印机的状态信息。

lptest

生成行式打印机波纹模式。

lsattr

显示系统中设备的属性特征和可能的属性值。

lscfg

显示设备的诊断信息。

lsconn	显示给定设备或某类设备可以接受的连接。
lscons	将控制台设备的名称写入标准输出。
lsdev	显示系统中的设备及其特征。
lsdisp	列出系统上当前可用的显示器。
lsfont	列出可用于显示的字体。
lskbd	列出当前可用于低功能终端（LFT）子系统的键盘映射。
lsparent	显示接受指定连接类型或设备的可能的父设备。
mkdev	往系统中添加设备。
mkfont	往系统中添加与显示相关的字体代码。
mknod	生成专用的文件。
mt (BSD)	将子命令提供给一个流磁带设备。
panel20	诊断 HIA 和 5080 控制单元之间的活动。
pdelay	启用或报告延迟登录端口的可用性。
pdisable	禁用登录端口。
penable	启用或者报告登录端口的可用性。
phold	禁用或者报告保留的登录端口的可用性。
pioattred	提供一种在虚拟打印机上格式化和编辑属性的方法。
piobe	打印机后端的打印作业管理器。
pioburst	为打印机输出生成裁纸页（首页和尾页）。
piocnvt	扩展和压缩一个预定义定义或虚拟打印机定义。
piodigest	将虚拟打印机定义属性值摘要到内存映象中，并将该内存映象保存到文件。
piofontin	从多语言字体软盘中复制字体。
pioformat	驱动打印机格式化程序。
piofquote	转换指定给 PostScript 打印机的特定控制字符。
pioout	打印机后端的设备驱动程序接口程序。
piopredef	创建预定义的打印机数据流定义。
portmir	

允许一个 TTY 流（监视器）连接到另一个 TTY 流（目标），并且监视发生在那个流上的用户会话。

pr

向标准输出写文件。

pshare

启用或者报告共享登录端口的可用性。

pstart

启用或者报告登录端口（正常、共享和延迟）的可用性。

pstat

解释不同系统表的内容并写入标准输出。

reset

初始化终端。

rmdev

从系统里除去设备。

rmt

允许远程访问磁带设备。

script

制作终端对话期的 typescript。

setmaps

设置终端映射或代码集映射。

splp

更改或显示打印机驱动程序设置。

stty

设置、复位和报告工作站操作参数。

stty-cxma

发送和报告 128 端口异步控制器的终端选项。

swapon

为页面调度和交换指定附加设备。

swcons

暂时将系统控制台输出重定向到一个指定的设备或文件。

sysdumpdev

更改运行系统中的主要或辅助转储设备指派。

tabs

在终端上设置制表符停止位。

tapechk

对流式磁带设备执行一致性校验。

tcopy

复制磁带。

tctl

将命令提供给一个流磁带设备。

termdef

查询终端特征。

tput

查询 **terminfo** 数据库中的终端描述符文件。

tset

初始化终端。

tsm

提供终端状态管理。

tty

把终端全路径名写入标准输出。

命令列表：文档和培训

apropos

用“关键词查找”定位命令。

catman

为手册创建 cat 文件。

explain

提供交互式同义词典。

help

为新用户提供信息。

learn

对使用文档、编辑器、宏和其它功能提供计算机辅助教学课程和实践。

man

在线显示帮助页条目。

命令列表: 文件系统

automount

自动安装 NFS 文件系统。

chfs

更改文件系统的属性。

chps

更改调页空间的属性。

chvfs

更改 **/etc/vfs** 文件中的条目。

crfs

添加一个文件系统。

crvfs

在 **/etc/vfs** 文件中创建条目。

defragfs

增加文件系统的邻近可用空间。

df

报告关于文件系统空间的信息。

dfsck

检查文件系统的一致性并交互式修复文件系统。

dosformat

格式化 DOS 软盘。

dumpfs

转储文件系统信息。

ff

为文件系统列出文件名和统计信息。

fsck

检查文件系统的一致性并交互式修复文件系统。

fsdb

调试文件系统。

istat

检查索引节点号。

lsfs

显示文件系统的特征。

mkfs

制作文件系统。

mklost+found

为 **fsck** 命令创建一个丢失和查找目录。

mkproto

构造原型文件系统。

mount

制造可以使用的文件系统。

ncheck

从索引节点号生成路径名。

proto

为文件系统构造原型文件。

rmfs

除去一个文件系统、其驻留的任何逻辑卷，以及在 **/etc/filesystems** 文件中的相关节。

rrestore

从远程机器的设备中复制以前的备份文件系统到本地机器。

skulker

除去不需要的文件以清除文件系统。

umount

卸装以前安装的文件系统、目录或文件。

unmount

卸装以前安装的文件系统、目录或文件。

update

定期更新超级块。

命令列表：游戏

arithmetic

测试算术技巧。

bj

启动扑克牌（二十一点）游戏。

craps

启动骰子游戏。

fish

玩钓鱼卡游戏。

fortune

从“财富游戏”数据库显示随机财富。

hangman

启动刽子手猜字游戏。

moo

启动猜数字游戏。

number

显示数字的写表单。

quiz

测试知识。

ttt

启动 tic-tac-toe 游戏。

turnoff

关闭 **usr/games** 目录下文件的许可权代码。

turnon

打开 **usr/games** 目录下文件的许可权代码。

wump

开始“寻找 wumpus”游戏。

命令列表：许可证使用管理

drm_admin

基于数据复制管理 （Data Replication Manager (DRM)）的管理员服务器，如 **glbd**，全局位置代理的副本（GLB）。

glbd

管理全局位置代理数据库。

lb_admin

监视和管理位置代理注册。

lb_find

获取全局位置代理（GLB）服务器守护程序及其属性的列表。

llbd

管理本地位置代理数据库的信息。

monitord

与 许可证使用管理 服务器通信，并为每一个可数的登录申请一个 AIX V4 并发使用许可证。

nrglbd

管理全局位置代理数据库。

命令列表：逻辑卷

chlv

只更改逻辑卷的特征。

chpv

更改卷组中的物理卷特征。

chvg

设置卷组的特征。

cplv

将逻辑卷的内容复制到新的逻辑卷。

exportvg

从一套物理卷中导出卷组的定义。

extendlv

通过从卷组里增加未分配的物理分区来增大逻辑卷的大小。

extendvg

将逻辑卷添加到卷组。

importvg

从一套物理卷中导入一个新的卷组定义。

lslv

显示关于逻辑卷的信息。

lspv

显示关于一个卷组中物理卷的信息。

lsvg

显示关于卷组的信息。

migratepv

从一个物理卷中将已分配的物理分区移到一个或多个其它的物理卷。

mirrorvg

将存在于一个给定卷组中的所有逻辑卷作成镜像。

mklv

创建一个逻辑卷。

mklvcopy

提供逻辑卷内的数据副本。

mkvg

创建一个卷组。

mkvgdata

创建一个包含关于卷组信息的文件，以用于**savevg** 和 **restvg** 命令的使用。

redefinevg

重定义设备配置数据中的给定卷组的物理卷集合。

reducevg

除去卷组中的物理卷。

reorgvg

重新组织卷组的物理分区分配。

restvg

恢复用户卷组及其所有的容器和文件，这些文件在由 **savevg** 命令创建的 **/tmp/vgdata/vgname/vgname.data** 备份映象的文件中指定。

rmlv

从卷组中除去物理卷。

rmlvcopy

从逻辑卷除去副本。

savevg

查找并备份所有属于指定卷组的文件。

synclvodm

使逻辑卷控件块、设备配置数据库、物理卷中的卷组描述符区域同步，或者重建它们。

syncvg

同步非当前逻辑卷的副本。

unmirrorvg

除去存在于卷组或指定磁盘中的镜像。

varyoffvg

释放一个卷组。

varyonvg

激活一个卷组。

命令列表：网络安装管理（NIM）

lsnim

显示关于网络安装管理（NIM）环境的信息。

nim

执行网络安装管理（NIM）对象的操作。

nimclient

容许网络安装管理（NIM）操作从一个 NIM 客户机执行。

nimconfig

初始化网络安装管理（NIM）客户机数据包。

niminit

显示关于网络安装管理环境的信息。

命令列表：数值数据

bc

提供任意精度算术语言的解释器。

dc

为任意精度整数运算提供交互式的台式计算器。

factor

对数字进行因子分解。

number

显示数字的书写格式。

units

将一种计量单位转换成为另一种等价的计量单位。

命令列表: 性能调整

acctcms	根据记帐记录产生命令使用率摘要。
acctcom	显示选定进程的记帐记录摘要。
accton	执行进程记帐过程。
filemon	监视和报告文件系统的性能。
fileplace	显示文件块在逻辑或物理卷中的位置。
gprof	显示调用图概要文件数据。
iostat	报告中央处理器（CPU）的统计信息和 tty、磁盘、CD-ROM 的输入／输出统计信息。
lsattr	显示系统中设备的属性特征以及可能的属性值。
lslv	显示关于逻辑卷的信息。
mmtu	显示、添加和删除最大传输单元（MTU）值，它用于发现路径 MTU 。
netpmon	监视活动并报告网络使用情况的统计信息。
netstat	显示网络状态。
nfsstat	显示关于网络文件系统 （NFS）和远程过程调用（RPC）的统计信息。
nice	运行指定优先级的命令。
no	配置网络选项。
nulladm	创建指定的文件，授予文件所有者和组读写许可权，授予其它用户读许可权。
ps	显示进程的当前状态。
renice	更改运行进程的优先级。
reorgvg	重新组织卷组的物理分区分配。
rmss	以不同的实内存大小仿真系统。
sar	收集、报告或存储系统活动信息。
stripnm	显示指定目标文件的符号信息。
svmon	捕获并分析虚拟内存的快照。
time	打印命令的执行时间。
timex	以秒为单位，报告一条命令的所用时间、用户时间和系统执行时间。
tprof	指定要进行概要分析的用户程序并执行它，然后生成报告。
trcnm	生成内核名称列表。
trcrpt	格式化来自跟踪日志的报告。
trcstop	停止跟踪功能。
vmstat	报告虚拟内存统计信息。

命令列表: 进程和命令

apply	为一组参数提供一条命令。
cron	自动运行命令。
cronadm	列出或除去 **crontab** 或 **at** 作业。
crontab	提交、列出或除去 **cron** 作业文件。
env	显示当前环境或为命令的执行设置环境。
fuser	使用文件或文件结构标识进程。
install	安装命令。
installbsd	安装命令（**install** 命令的 BSD 版本）。
ipcs	报告进程间通信设备状态。
kill	发送信号给正在运行的进程。
killall	取消除调用进程之外的所有进程。
lastcomm	显示关于上次执行的命令的信息。
nice	运行指定优先级的命令。
nohup	不挂断地运行命令。
ps	显示进程的当前状态。
renice	更改正在运行进程的优先级。
sleep	暂挂执行一个时间间隔。
time	打印命令的执行时间。
timex	以秒为单位，报告一条命令的所用时间、用户时间和系统执行时间。

wait 等待直到进程标识终止。
whatis 描述命令执行何种功能。
xargs 构造参数列表并运行命令。

命令列表: 队列

at 稍后运行命令。
atq 显示等待运行的作业队列。
atrm 除去 **at** 命令假脱机作业。
batch 当系统负载水平允许时运行作业。
chprtsv 在客户机或服务器上更改打印服务配置。
chque 更改队列名。
chquedev 更改打印机或绘图仪队列设备名。
chvirprt 更改虚拟打印机的属性值。
digest 将 ASCII 格式的 **/etc/qconfig** 文件转换成 **/etc/qconfig.bin** 文件，该文件是一个由 **qdaemon** 命令使用的队列配置的二进制版本。
disable 禁用打印机队列。
enq 将文件列队。
lpq 检验假脱机队列。
lpr 将打印作业列队。
lprm 从行式打印机假脱机队列除去作业。
lsallq 列出所有配置的队列的名称。
lsallqdev 将所有配置的打印机和绘图仪队列设备名在一个指定的队列中列出。
lsprtsv 显示存储在数据库中的打印服务信息。
lsque 显示队列节名。
lsquedev 显示设备节名。
lsvirprt 显示虚拟打印机的属性值。
mkprtsv 配置主机上的基于 TCP/IP 的打印服务。
mkque 往系统中添加打印机队列。
mkquedev 往系统中添加打印机队列设备。
mkvirprt 创建虚拟打印机。
piodmgr 将对象数据管理（ODM）数据库压缩进 **var/spool/lpd/pio/@local/smit** 目录中。
piolsvp 列出系统上的虚拟打印机。
piomgpdev 管理打印机伪设备。
piomkapqd 建立 SMIT 对话以创建打印队列和打印机。
piomkpq 生成打印机队列。
piomsg 向用户发送打印机后端消息。
qadm 为打印假脱机系统执行系统管理功能。
qcan 取消打印作业。
qchk 显示打印队列状态。
qdaemon 调度由 **enq** 命令列队的作业。
qhld 保留假脱机打印作业。
qmov 将假脱机打印作业移到另一队列。
qpri 在打印队列中对某项作业划分优先级。
qprt 启动打印作业。
qstatus 为打印机假脱机系统提供打印状态。
rembak 发送一项打印作业到远程服务器上的队列。
rmprtsv 在客户机和服务器上取消配置打印服务。
rmque 从系统中除去打印队列。
rmquedev 从系统中除去打印机或绘图仪队列。
rmvirprt 除去虚拟打印机。

命令列表: 屏幕输出

banner	用大字体将 ASCII 字符串写入标准输出。
cal	显示日历。
calendar	将提示信息写入标准输出。
echo	将字符串写入标准输出。
leave	在您不得不离开时提醒您。
more	在显示屏幕每次一屏显示连续的文本。
news	将系统消息项写入标准输出。
page	在显示屏幕每次一屏显示连续的文本。
tail	从指定点开始将文件写入标准输出。
vacation	返回一条消息给发送方表示邮件接收方正在度假。

命令列表: 安全性和系统访问

acledit	编辑文件访问控制信息。
aclget	显示文件访问控制信息。
aclput	设置文件访问控制信息。
audit	控制系统审计。
auditbin	管理审计信息箱。
auditcat	写入审计记录箱。
auditpr	格式化箱或流审计记录到显示设备或打印机。
auditselect	根据已定义的标准选择审计记录进行分析。
auditstream	为阅读审计记录创建一条通道。
chfn	更改用户的通用信息。
chgroup	更改组的属性。
chgrp	更改文件或目录的组所有权。
chgrpmem	更改组的管理员或成员。
chmod	更改许可权方式。
chown	更改与文件关联的用户。
chrole	更改角色属性。
chsec	更改在安全性节文件中的属性。
chsh	更改用户的登录 shell。
chtcb	更改或查询文件的**可信的计算根**属性。
chuser	更改指定用户的属性。
groups	显示组成员。
grpck	验证组定义的正确性。
last	显示关于先前登录的信息。
lastlogin	更新 **/var/adm/acct/sum/loginlog** 文件，以显示每个用户登录的最近日期。
lssec	列出在安全节文件内的属性。
lock	保留终端。
login	启动用户会话。
logname	显示登录名。
logout	停止端口上的所有进程。
lsgroup	显示组的属性。
lslicense	显示可以同时登录的最大用户数。
lsrole	显示角色属性。
lsuser	显示用户帐户的属性
makekey	生成加密密钥。
mkgroup	生成新的组。
mkpasswd	创建用户数据库的散列后备版本。
mkrole	创建新的角色。
mkuser	创建新的用户帐户。
mkuser.sys	定制新的用户帐户。

newgrp	更改主组标识。
nulladm	创建活动的记帐数据文件。
passwd	更改用户的密码。
pwdadm	管理用户的密码。
pwdck	验证本地认证信息的正确性。
rmgroup	除去组。
rmrole	除去角色。
rmuser	除去用户帐户。
Rsh	调用 Bourne shell 的受限版本。
setgroups	为会话重新设置辅助组标识。
setsenv	重新设置用户的受保护状态环境。
shell	以用户的缺省凭证和环境执行 shell。
su	更改与会话关联的用户标识。
sysck	在安装和更新过程中检查库存信息。
tcbck	审计系统的安全状态。
usrck	验证用户定义的正确性。
xss	提高无人工作站的安全性。

命令列表：shell

alias	定义或显示别名。
basename	返回字符串参数的基文件名。
bg	在后台运行作业。
bsh	调用 Bourne shell。
chsh	更改用户的登录 shell。
command	执行一个简单命令。
csh	调用 C shell。
expr	将自变量评估为表达式。
false	返回一个零出口值（**真**）或一个非零出口值（**假**）。
fc	处理命令历史列表。
fg	在前台运行作业。
getopt	分析命令行标志和参数。
hash	记忆或报告命令路径名。
jobs	显示当前会话中的作业状态。
ksh	调用 Korn shell。
line	从标准输入读取一行。
patch	对文件应用更改。
read	从标准输入读取一行。
rsh	在远程主机上或登录到远程主机执行指定的命令。
Rsh	调用 Bourne shell 的受限版本。
sh	调用缺省 shell。
shell	以用户的缺省凭证和环境执行 shell。
tee	显示程序的输出并将其复制到一个文件中。
test	评估条件表达式。
true	返回一个零出口值（**真**）或一个非零出口值（**假**）。
tsh	在可信的 shell 中解释命令。
type	写入命令类型的描述
ulimit	设置或报告用户资源限制。
unalias	除去别名定义。
xargs	构造自变量列表并运行命令。
yes	重复输出肯定响应。

命令列表: 系统记帐和统计信息

accton	执行进程记帐过程。
date	显示或设置日期或时间。
diag	执行硬件问题确定。
dp	对日期进行分析并重定格式。
du	总结磁盘使用情况。
dump	转储目标文件的选定部分。
errclear	从错误日志删除条目。
errdead	从系统转储中抽取错误记录。
errdemon	启动错误记录守护程序并将记录条目写入错误日志。
errinstall	在错误记录消息集中安装消息。
errlogger	记录操作员消息。
errmsg	添加消息到错误日志消息目录中。
errpt	处理已记录的错误报告。
errstop	终止错误记录守护程序。
errupdate	更新错误记录模板库。
getconf	将系统配置变量值写入标准输出。
id	显示指定用户的系统标识。
iostat	报告中央处理器（CPU）的统计信息和 tty、磁盘、CD-ROM 的输入／输出统计信息。
ipcs	报告进程间通信设备状态。
ipreport	从指定的包跟踪文件中生成包跟踪报告。
iptrace	提供因特网协议的接口级包跟踪。
last	显示关于先前登录的信息。
locale	写当前语言环境或所有公共语言环境的信息。
logger	向系统日志中写入条目。
pac	准备打印机／绘图仪记帐记录。
pstat	解释不同系统表的内容并写入标准输出。
sa	总结记帐记录。
sa1	将二进制数据收集并保存在 **/var/adm/sa/sa***dd* 文件中。
sa2	将每日报告写入 **/var/adm/sa/sar***dd* 文件。
sadc	提供系统活动报告软件包。
sar	收集、报告或存储系统活动信息。
snap	搜集系统配置信息。
sysdumpstart	提供命令行界面以便开始将内核转储至主要或辅助转储设备。
sysline	在终端的状态行显示系统状态。
syslogd	记录系统消息。
tput	查询 **terminfo** 数据库中的终端描述符文件。
uname	显示当前操作系统名称。
uptime	显示系统运行了多少时间。
users（BSD）	显示当前系统上的用户的压缩列表。
vmstat	报告虚拟内存统计信息。
w	打印当前系统活动摘要。
watch	观察一个可能是不可信的程序。
who	标识当前登录的用户。
whoami	显示登录名。
whois	根据用户标识或别名标识一个用户。

acct/* 命令

ac	打印连接时间记录。
acctcms	根据记帐记录产生命令使用率摘要。
acctcom	显示选定进程的记帐记录摘要。
acctcon1	执行连接时间记帐。

acctcon2 执行连接时间记帐。
acctdisk 执行磁盘使用率记帐。
acctdusg 执行磁盘使用率记帐。
acctmerg 将总的记帐文件合并成一个中间文件或一份日报。
acctprc1 执行进程记帐过程。
acctprc2 执行进程记帐过程。
accton 执行进程记帐过程。
acctwtmp 控制连接时间记帐记录，以便更改格式和保持记录的正确性。
chargefee 收取用户使用计算机资源的费用。
ckpacct 检查数据文件大小以便进程记帐。
diskusg 根据用户标识生成磁盘记帐数据。
dodisk 初始化磁盘使用率记帐。
fwtmp 控制连接时间记帐记录，以便更改格式和保持记录的正确性。
lastlogin 报告系统上的每个用户的最近登录日期。
monacct 执行每月或定期记帐。
nulladm 创建活动的记帐数据文件。
prctmp 显示会话记录文件。
prdaily 创建前一天记帐数据的 ASCII 报告。
prtacct 以 **tacct** 格式来格式化和显示文件。
remove 从 **var/adm/acct** 子目录删除文件。
runacct 运行每日记帐。
shutacct 关闭正在处理的记帐。
startup 在系统启动时打开记帐功能。
turnacct 为 **accton** 命令提供接口以打开或关闭进程记帐。
wtmpfix 控制连接时间记帐记录，以便更改格式和保持记录的正确性。

命令列表: 系统资源

chps 更改调页空间的属性。
chserver 更改子服务器对象类中的子服务器定义。
chssys 更改子系统对象类中的子系统定义。
compress 压缩和扩展数据。
lslicense 显示可以同时登录的用户范围。
lsps 显示调页空间的特征。
lssrc 获取子系统、子系统组或子服务器的状态。
mknotify 往“通知”对象类中添加通知方法定义。
mkps 往系统中添加附加调页空间。
mkserver 往子服务器对象类中添加子服务器定义。
mkssys 往子系统对象类中添加子系统定义。
pack 压缩文件。
pagesize 显示系统页面大小。
pcat 解压缩文件并将它们写入标准输出。
rmnotify 从“通知”对象类中除去通知方法定义。
rmps 从系统中除去调页空间及其驻留的任何逻辑卷。
rmserver 从子服务器类型对象类中除去子服务器定义。
rmssys 从子系统对象类中除去子系统定义。
srcmstr 启动系统资源控制器。
startsrc 启动子系统、子系统组或子服务器。
stopsrc 停止子系统、子系统组或子服务器。
swapon 为页面调度和交换指定附加设备。
tracesoff 关闭对子系统、子系统组或子服务器的跟踪。
traceson 打开对子系统、子系统组或子服务器的跟踪。
uncompress 压缩和扩展数据。
zcat 压缩和扩展数据。

命令列表: 软件安装

bootlist 更改可用于系统的 IPL 设备列表或列表里设备的顺序。
bootparamd 为无盘客户机提供引导信息。
bosboot 创建引导设备。
chitab 更改 **/etc/inittab** 文件中的记录。
ckprereq 验证所有必备软件可用且处于适当的修订级别。
fastboot 重新启动系统。
fasthalt 停止处理器。
halt 停止处理器。
init 初始化并控制进程。
installp 用兼容的安装软件包安装可用软件产品。
inudocm 显示包含补充信息的文件的内容。
inurecv 恢复由 **inusave** 命令保存的所有文件。
inurest 执行简单归档并恢复 **installp** 命令和 shell 脚本的操作。
inusave 保存安装过程中已安装或更新的文件。
inuumsg 显示具体错误或由软件产品安装过程提供的诊断消息。
logger 向系统日志中写入条目。
lppchk 验证可安装软件产品的文件。
lsitab 列出 **/etc/inittab** 文件中的记录。
lslpp 列出软件产品。
mkboot 创建引导映象、引导记录和服务记录。
mkitab 在 **/etc/inittab** 文件中做记录。
rc 执行正常的启动初始化。
reboot 重新启动系统。
refresh 请求刷新子系统或子系统组。
rmitab 除去 **/etc/inittab** 文件中的记录。
shutdown 结束系统操作。
smit 执行系统管理。
sync 更新索引节点表并将缓冲文件写入硬盘。
sysck 在安装和更新过程中检查库存信息。

命令列表: 用户界面

AIXwindows:

custom 允许用户定制 X 应用程序。
dtscript 建立在 X Window System 环境中使用的简单对话框。
mwm 运行 AIXwindows 视窗管理器。
uil 启动 AIXwindows 系统用户界面语言编译器的命令。
xmbind 配置虚拟键绑定

增强的 X 视窗:

addX11input 往 ODM 数据库添加 X11 输入扩展记录。
aixterm 初始化增强的 X 视窗终端仿真器。
bdftopcf 将位图分布格式字体转换成可移植编译格式的字体编译器。
deleteX11input 从 ODM 数据库删除 X11 输入扩展记录。
listX11input 列出加入到 ODM 数据中的 X11 输入扩展记录。
mkfontdir 从字体文件目录中创建一个 **fonts.dir** 文件。
resize 设置 **TERMCAP** 环境变量和终端设置为当前窗口大小。
rgb 从标准输入读取行并将它们插入到数据库以将颜色名和特定 RGB 值关联起来。
startx 初始化一个 X 会话。

uil	启动 AIXwindows 系统的用户界面语言编译器。
X	启动 X 服务器。
xauth	编辑和显示用于连接 X 服务器的授权信息。
xclock	连续显示当前时间。
xcmsdb	装入、查询或除去存储在屏幕根窗口上属性中的屏幕颜色个性化数据。
xdm	支持 XDMP 的 X 显示管理器。
xfs	为 X Window System 显示服务器提供字体。
xhost	控制谁能访问当前主机上的增强 X-Windows。
xinit（增强 X-Windows）	
xinit（X11R5）	初始化 X Window System。
xlock	锁定本地 X 显示器，直到输入密码。
xlsfonts	显示用于 X 的字体列表。
xmodmap	修改 X 服务器中的键映射。
xpr	格式化窗口转储文件以便输出到打印机上。
xrdb	执行 X 服务器资源数据库实用程序。
xset（X-Windows）	为 X-Windows 环境设置选项。
xsetroot	**x** 命令的根窗口参数设置实用程序。
xterm	提供 X Window System 终端仿真器。
xwd	转储增强 X-Windows 的图像。
xwud	检索增强 X-Windows 窗口的转储图像。

命令列表：宏

add_netopt	添加网络选项结构到网络选项列表。
assert	验证程序断言。
auth_destroy	破坏认证信息。
clnt_call	调用与 *clnt* 参数相关的远程过程。
clnt_control	更改或检索关于客户机对象的各种信息。
clnt_destroy	破坏客户机的 RPC 句柄。
clnt_freeres	由 RPC/XDR 系统分配的自由数据。
clnt_geterr	从客户机句柄复制错误信息。
DTOM	将在一个 **mbuf** 结构内部任何位置的地址转换为 **mbuf** 结构头。
del_netopt	从网络选项列表中删除网络选项结构。
feof、**ferror**、**clearerr** 或 **fileno**	检查流的状态。
M_HASCL	确定 **mbuf** 结构是否具有附加群集。
MTOCL	将指向 **mbuf** 结构的指针转换成指向附加群集头的指针。
MTOD	将指向 **mbuf** 结构的指针转换成为指向存储在 **mbuf** 结构中的数据的指针。
M_XMEMD	返回 **mbuf** 交叉内存描述符的地址。
man	为帮助页提供格式化工具。
m_copy	为 **mbuf** 结构列表的一部分或全部创建一个副本。
m_clget	分配页面大小的 **mbuf** 结构群集。
me	提供格式化工具以创建不同风格的技术论文。
m_getclust	从 **mbuf** 缓冲池中分配 **mbuf** 结构，并附加页面大小的群集。
mm	为商业文档，如备忘录、信函和报告提供格式化工具。
mptx	格式化由 **ptx** 命令产生的置换索引。
ms	为各种风格的文章、论文和书籍提供格式化工具。
mv	简化视图及投影幻灯片的排版。
svc_destroy	破坏远程过程调用（RPC）服务传输句柄。
svc_freeargs	由 RPC/XDR 系统分配的自由数据。
svc_getargs	译码 RPC 请求的自变量。

svc_getcaller	获取过程调用者的网络地址。
varargs	处理变长参数的列表。
xdr_destroy	破坏由 *xdrs* 参数指向的 XDR 流。
xdr_getpos	返回描述在数据流中当前位置的无符号整数。
xdr_inline	返回由 *xdrs* 参数指向流缓冲区的指针。
xdr_setpos	更改 XDR 流中的当前位置。

编程工具

命令列表: 调试器

adb	提供一般用途的调试程序。
dbx	为调试和运行程序提供环境。
od	以指定格式显示文件。
prof	显示目标文件概要数据。
savecore	保存操作系统的核心转储。
syscall	执行指定的子例程调用。
trace	记录选定的系统事件。
trcdead	从系统转储映象中抽取跟踪缓冲区。
trcnm	生成内核名称列表。
trcrpt	格式化来自跟踪日志的报告。
trcstop	停止跟踪功能。
trcupdate	添加、替代或删除跟踪报告格式模板。

命令列表: 消息

dspcat	显示所有或部分消息编目。
dspmsg	显示消息编目中选定的消息。
gencat	创建和修改消息编目。
mkcatdefs	预处理消息源文件。
mkstr	创建错误消息文件。
runcat	通过管道将输出数据从 **mkcatdefs** 命令传输到 **gencat** 命令。
xstr	从 C 程序中提取字符串以实现共享字符串。

命令列表: 源程序

admin
创建并控制 SCCS 文件。

asa
打印 PORTRAN 文件。

cdc
更改 SCCS 增量中的注释。

comb
组合 SCCS 增量。

ctags
制作标记文件以帮助定位源文件中的对象。

delta
在 SCCS 文件中创建一个变化量。

get
创建 SCCS 文件的一个指定版本。

prs
显示源代码控制系统（SCCS）文件。

rmdel
从 SCCS 文件中除去变化量。

sact
显示当前 SCCS 文件编辑状态。

sccs
SCCS 命令的管理程序。

sccsdiff
比较 SCCS 文件的两个版本。

sccshelp
提供 SCCS 消息或命令的信息。

unget
取消前一 **get** 命令。

unifdef
从文件中除去 ifdef 行。

val
验证 SCCS 文件。

vc
用标识关键字代替分配的值。

vgrind
格式化易于阅读的程序清单。

whereis
定位程序的源、二进制文件或手册。

which
定位包括别名和路径（仅对于 **csh**（C shell）命令）的程序文件。

命令列表: 目标文件

ld
链接目标文件。

lorder
为在对象库中的成员文件寻找最佳顺序。

make
维护最新的程序版本。

nm
显示目标文件的符号表。

prof

size
显示目标文件概要数据。

slibclean
显示“扩展公共目标文件格式”（XCOFF）目标文件的节大小。

除去内核和库存储器中当前未使用的模块。

strings
在目标文件或二进制文件中查找可打印的字符串。

strip
通过除去被绑定程序和符号调试程序使用的信息，减小“扩展公共目标文件格式”（XCOFF）目标文件的大小。

命令列表：杂类语言

bc
提供任意精度算术语言的解释器。

bs
编译和解释中等大小的程序。

m4
预处理文件，扩展宏定义。

sno
提供 SNOBOL 解释器。

命令列表：C 工具

cb
将 C 源代码转换成一种易于阅读的格式。

cflow
生成外部引用的 C 流程图。

cpp
执行文件包含和 C 语言源文件的宏替换。

cxref
创建 C 程序交叉引用清单。

execerror
将错误消息写入标准错误。

indent
重新格式化 C 语言程序。

ipcrm
除去消息队列、信号集或共享内存标识符。

lex
生成 C 语言程序，该程序和输入流的简单词法分析的模式相匹配。

lint
检查 C 语言程序中的潜在问题。

m4
处理文件，扩展宏定义。

mkstr
创建错误消息文件。

regcmp
将模式编译成 C 语言 **char** 声明。

tic
将 **terminfo** 描述符文件由源程序翻译成已编译格式。

xstr
从 C 程序中提取字符串以实现共享字符串。

yacc
从由上下文无关语法规范组成的输入生成 LR(1) 语法分析程序。

命令列表: 汇编程序和编译器

汇编程序:

as
汇编一个源文件。

FORTRAN:

asa
打印 PORTRAN 文件。

fpr
打印 PORTRAN 文件。

fsplit
将 FORTRAN 源代码分割成为单独的例程文件。

struct
将 FORTRAN 程序转换成 RATFOR 程序。

命令列表: 对象数据管理器（ODM）

odmadd
向已创建的对象类添加对象。

odmchange
更改指定对象类中的选定对象的内容。

odmcreate
产生对于 ODM 应用程序开发必要的 **.c**（源）和 **.h**（包含）文件并创建空的对象类。

odmdelete
从指定的对象类中删除选定的对象。

odmdrop
除去对象类。

odmget
从指定的对象类中检索对象到 **odmadd** 格式。

odmshow
在屏幕上显示对象类定义。

restbase
从引导映象读取基本定制信息，并将其恢复到系统引导阶段 1 期间使用的“设备配置”数据库。

savebase
将“设备配置”数据库中的基本定制设备信息保存到引导设备中。

附录 C. 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。关于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中描述的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.

以下条款不适合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（无论是明示的，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。

© Copyright IBM Corp. 1997, 2003

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、兼容性或任何其它关于非 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。那些 Web 站点中的资料不是本 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。

本信息包含在日常商业操作中所使用的数据和报告的示例。为了尽可能完整地说明它们，示例中包含了个人、公司、商标和产品的名称。所有这些名称都是虚构的，任何与实际商业企业的名称、地址的相似之处纯属巧合。

版权许可证:

本信息源语言中包含样本应用程序，它说明了各种操作平台上的编程技术。为了开发、使用、销售或分发符合编写样本程序的操作平台的应用程序编程接口的应用程序，您可以任何形式复制、修改和分发这些样本程序而无需对 IBM 付费。这些示例未在所有环境下彻底测试过。因此，IBM 不能保证或暗示这些程序的可靠性、适用性或功能。为了开发、使用、销售或分发符合 IBM 的应用程序编程接口的应用程序，您可以任何形式复制、修改和分发这些程序而无需对 IBM 付费。

此处包含的任何性能数据都是在受控环境中测得的。因此，在其它操作环境中获得的结果可能会有明显的不同。有些测量可能是在开发级的系统上进行的，因此不保证与一般可用系统上进行的测量结果相同。此外，有些测量是通过推算而估计的，实际结果可能会有差异。本文档的用户应当验证其特定环境的适用数据。

商标

以下术语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

AIX 5L

IBM

UNIX 是 The Open Group 在美国和其它国家或地区的注册商标。

其它公司、产品或者服务名称可能是其它公司的商标或者服务标志。

索引

[A]

安全性
  命令的功能列表 259
  锁定工作站屏幕
    使用 xss 命令 174

[B]

帮助
  描述命令功能 58
绑定
  到 NIS 服务器
    使用 ypbind 守护程序 203
  配置虚拟键
    使用 xmbind 命令 145
编程语言
  命令的功能列表
    汇编程序 268
    杂类 267
    C 267
    FORTRAN 268
编辑器
  命令的功能列表 243
编译器
  命令的功能列表 268
标识服务器
  用于给定映射，主控服务器
    使用 ypwhich 命令 217
标识号，显示
  NIS 映射
    使用 yppoll 命令 211
别名
  定位 64

[C]

参数列表
  构造
    使用 xargs 命令 110
查询 / 控制程序，启动
  xntpdc 守护程序
    使用 xntpdc 命令 155
查找信息
  在 NIS 映射中
    使用 ypserv 守护程序 213
程序
  从 Imakefile 创建 Makefile 146
程序 *(续)*
  监视
    使用 watch 命令 54
程序清单
  格式化
    使用 vgrind 命令 10
重复响应
  生成 203
从属服务器，NIS
  提示复制 NIS 映射
    使用 yppush 命令 212

[D]

打印
  活动摘要
    使用 w 命令 50
  NIS 映射
    使用 ypcat 命令 205
调试器
  命令的功能列表 265
队列
  命令的功能列表 258

[F]

仿真
  xterm 命令 177

[G]

更新 NIS 映射
  使用 ypupdated 守护程序 216
工作负荷管理
  请参阅 WLM 80
工作站屏幕
  锁定
    使用 xss 命令 174
关键字
  SCCS
    替代 7
管理 X 显示器集合
  通过 XDMCP 支持
    使用 xdm 命令 118

© Copyright IBM Corp. 1997，2003

[H]

行数
  计数
    用 wc 命令 55
宏
  功能列表 264
  文本格式化 249
汇编程序
  命令的功能列表 268

[J]

记帐系统
  更正格式问题 95
键绑定
  配置虚拟
    使用 xmbind 命令 145
键映射
  修改
    使用 xmodmap 命令 149
将项添加到
  产品、lpp、历史数据库
    使用 vpdadd 命令 47
  供应商数据库
    使用 vpdadd 命令 47
脚本
  ewallevent 52
  wallevent 52
接收和执行请求
  从 yppasswd 命令
    使用 yppasswdd 守护程序 210
解析器
  用 yacc 命令创建 201
进程
  命令的功能列表 257
进程暂挂
  暂挂执行 51
卷组
  激活
    使用 varyonvg 命令 5
  释放
    使用 varyoffvg 命令 4

[K]

客户机
  指导到特定服务器
    使用 ypset 命令 215

[L]

连接
  到 NIS 服务器
    使用 ypbind 守护程序 203
路径名
  可执行文件
    查找 64
逻辑卷
  命令的功能列表 255

[M]

命令
  vacation 1
  vpdadd 47
  w 50
  wall 51
  who 66
  wlmassign 74
  wlmcntrl 77
  wlmmon 80
  wlmperf 80
  wlmstat 84
  wsm 91
  wsmaccess 92
  X 97
  xargs 110
  xdm 118
  xmbind 145
  xmodmap 149
  xntpdc 155
  xterm 177
  ypcat 205
  ypinit 206
  yppasswd 209
  yppoll 211
  yppush 212
  ypset 215
  ypwhich 217
  ypxfr 218
命令行
  运行
    使用 xargs 命令 110
目标文件
  命令的功能列表 266
目录
  命令的功能列表 242

[N]

内存管理
报告虚拟内存统计信息 39

[P]

培训
命令的功能列表 252
配置
虚拟键绑定
使用 xmbind 命令 145
屏幕输出
命令的功能列表 259
屏幕锁定
控制
使用 xss 命令 174
屏幕颜色个性化数据
查询
使用 xcmsdb 命令 117
从根窗口除去
使用 xcmsdb 命令 117
从根窗口装入
使用 xcmsdb 命令 117

[R]

软件安装
命令的功能列表 263

[S]

设备
命令的功能列表 249
时间
显示当前的
使用 xclock 命令 115
实时消息
发送给其它用户
使用 write 命令 86
守护程序
ypbind 203
yppasswdd 210
ypserv 213
ypupdated 216
授权信息
编辑和显示
使用 xauth 命令 113
数值数据
命令的功能列表 256
顺序号，显示
NIS 映射
使用 yppoll 命令 211
锁定 X 显示器直到输入密码
使用 xlock 命令 142

[T]

通信工具
命令的功能列表 231
通信通道
通过安全通道发送邮件
使用 xsend 命令 169
通信信道
安全地接收邮件
使用 xget 命令 134
图像
检索
使用 xwud 命令 199
显示
使用 xwud 命令 199
转储
使用 xwd 命令 198

[W]

网络管理程序。
命令的功能列表 239
网络计时协议命令
xntpd 152
网络计时协议守护程序
启动 xntpdc
使用 xntpdc 命令 155
网络密码
在 NIS 中更改
使用 yppasswd 命令 209
网络信息服务 203
文本格式化
宏软件包列表 249
宏软件包清单 249
命令的功能列表 246
文档
命令的功能列表 252
文件
大于 2 吉字节
命令 227
定位节 63
可执行文件
定位 64
扩展 220
命令的功能列表 243
压缩 221

文件 *(续)*
SCCS
显示 标识信息 57
验证 3
文件内容
命令的功能列表 244
文件系统
命令的功能列表 253

[X]

系统
打印活动摘要
使用 w 命令 50
系统访问 260
命令的功能列表 259
系统记帐
命令的功能列表 261
系统统计信息
命令的功能列表 261
系统资源
命令的功能列表 262
消息
从远程系统发送
使用 writesrv 命令 90
从远程系统接收
使用 writesrv 命令 90
对处理的提示
使用 whatnow 命令 59
发送给其它用户
使用 write 命令 86
列出收件人地址
使用 whom 命令 72
命令的功能列表 265
写给所有用户
使用 wall 命令 51
验证收件人地址
使用 whom 命令 72
性能调整
命令的功能列表 257
虚拟键绑定
配置
使用 xmbind 命令 145
虚拟内存
报告统计信息 39

[Y]

页替换 46
已登录，用户
识别
用 who 命令 66
映射别名表
显示
使用 ypwhich 命令 217
用户
将消息写给所有
使用 wall 命令 51
识别那些登录的
用 who 命令 66
显示登录名 69
用户名目录
搜索标识
用 whois 命令 71
搜索别名
用 whois 命令 71
邮件
发送 vacation 消息
使用 vacation 命令 1
通过安全的通信信道接收
使用 xget 命令 134
通过安全通信通道发送
使用 xsend 命令 169
游戏
猎取 wumpus 96
命令的功能列表 254
源程序
命令的功能列表 266

[Z]

指导客户机
到特定服务器
使用 ypset 命令 215
终端
命令的功能列表 249
终端仿真器
为 X Window System 提供的
使用 xterm 命令 177
主机
控制访问
使用 xhost 命令 135
主控服务器
标识给定映射
使用 ypwhich 命令 217
转储文件
格式化成打印机输出
使用 xpr 命令 161
资源数据库
为 X 服务器执行实用程序
使用 xrdb 命令 165
自变量列表
构造
使用 xargs 命令 110

字符串，共享
使用 xstr 命令 175
字符数
计数
用 wc 命令 55
字数
计数
用 wc 命令 55
字体
为 X Windows 显示服务器提供 132
字体服务器
添加 107
字体路径元素
除去 109
字体目录
添加 108

A

acct/* 命令 261
ARPANET
whois 命令的用户 71
ATE 程序
命令的功能列表 230
用 xmodem 传输文件 147
xmodem 命令 147

B

BNU
命令的功能列表 230

E

ewallevent 脚本 52

I

idcmds 231, 234, 239, 252, 257, 259

M

MH
命令的功能列表 233
MH 命令
调用可视界面以便一起使用
使用 vmh 命令 27

N

NCS 命令
功能列表 242
NFS 命令
功能列表 234
NIS 服务器
绑定到
使用 ypbind 守护程序 203
传输 NIS 映射
使用 ypxfr 命令 218
设置 NIS 映射 206
使用 ypinit 命令 206
NIS 命令
功能列表 236
ypcat 205
ypinit 206
ypmatch 208
yppasswd 209
yppoll 211
yppush 212
ypset 215
ypwhich 217
ypxfr 218
NIS 守护程序
ypbind 203
yppasswdd 210
ypserv 213
ypupdated 216
NIS 网络密码
更改
使用 yppasswd 命令 209
NIS 映射
查找信息
使用 ypserv 守护程序 213
传输到 NIS 服务器
使用 ypxfr 命令 218
打印
使用 ypcat 命令 205
更新
使用 ypupdated 守护程序 216
使用 ypinit 命令 206
提示 NIS 从属服务器进行复制
使用 yppush 命令 212
显示标识号
使用 yppoll 命令 211
在 NIS 服务器上设置 206

O

ODM（对象数据管理器）
命令的功能列表 268

S

SCCS
关键字
替代值 7
文件
显示 标识信息 57
验证 3
SCCS 命令
val 3
vc 7
what 57
shell
在 C 中定位可执行文件 64
shell 脚本
重复响应
生成 203
STREAMS 命令
功能列表 239

T

TCP/IP 命令
功能列表 239

V

vacation 命令 1
vacation 消息
发送给邮件接收方
使用 vacation 命令 1
varyoffvg 命令 4
varyonvg 命令 5
vgrind 命令 10
vi 编辑器
定义宏 15
定制 12
滚动 22
过程
滚动 22
命令方式 12
启动 27
设置命令
使用 13
文本输入方式 12
映射键 16
中断和结束 26
子命令 20
最后行方式 12
vi 命令 11
限制 12
view 命令 27
vmh 命令 27
vmo 命令 28
vmtune 命令 46
vpdadd 命令 47

W

w 命令 50
wall 命令 51
wallevent 脚本 52
watch 命令 54
wc 命令 55
whatis 命令 58
whatnow 命令 59
which_fileset 命令 64
who 命令 66
whodo 命令 69
whois 命令
描述 71
示例 72
ARPANET
使用 71
WLM
分析 80
wlmassign 命令 74
wlmcheck 命令 76
wlmcntrl 命令 77
wlmmon 80
wlmperf 80
wlmstat 命令 84
write 命令 86
writesrv 命令 90
wsm 91
wsmaccess 92
wtmpfix 命令 95

X

X 服务器
启动
使用 X 命令 97
修改键映射
使用 xmodmap 命令 149
x 命令 97
X 显示器
管理集合
使用 xdm 命令 118
锁定直到输入密码
使用 xlock 命令 142
显示 troff 文件
使用 xpreview 命令 163
xargs 命令 110

xauth 命令 113
xclock 命令 115
设置缺省值 116
xcmsdb 命令 117
xdm 命令 118
XDMCP
管理 X 显示器集合的支持
使用 xdm 命令 118
xfs 命令 132
xget 命令 134
xhost 命令 135
xinit 命令 136
xlock 命令 142
xlsfonts 命令 144
xmbind 命令 145
xmodem 命令 147
xmodem 协议 147
xmodmap 命令 149
XNS
xnsquery 命令 150
XNSrouted 守护程序 151
xnsquery 命令
XNS 150
XNSrouted 守护程序
XNS 151
xntpd 守护程序 152
xntpdc 命令 155
xpr 命令 161
xpreview 命令 163
xrdb 命令 165
xsend 命令 169
xsetroot 命令 173
xss 命令 174
xstr 命令 175
xterm 命令 177
安全性 186
菜单 186
操作 188
程序错误 191
仿真 177
环境 191
控制序列
定义 191
VT100 方式 192
xterm 描述限制 197
鼠标跟踪 197
提供终端仿真器
用于 X Window System 177
指针用法 185
资源 181
字符类 187
xwd 命令 198
xwud 命令 199
X-Windows
设置根窗口参数 173
设置环境选项 170
x_add_fs_fpe 命令 107
x_add_nfs_fpe 命令 108
x_rm_fpe 命令 109

Y

yacc 命令 201
ypbind 守护程序 203
ypcat 命令 205
ypinit 命令 206
ypmatch 命令 208
yppasswd 命令 209
接收和执行请求
使用 yppasswdd 守护程序 210
yppasswdd 守护程序 210
yppoll 命令 211
yppush 命令 212
从属服务器，NIS
提示复制 NIS 映射 212
ypserv 守护程序 213
ypset 命令 215
ypupdated 守护程序 216
ypwhich 命令 217
ypxfr 命令 218

Z

zcat 命令 220
zdump 命令 221
zic 命令 222

读者意见表

AIX 5L 版本 **5.2**
命令参考大全，卷 **6**，**v - z**

S152-0347-01

姓名

地址

单位及部门

电话号码

读者意见表

S152-0347-01

IBM

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 中国公司上海分公司，汉化部
中国上海市淮海中路 333 号瑞安广场 10 楼
邮政编码：200021

折起并封口　　**请勿使用钉书机**　　折起并封口

S152-0347-01

请沿此线
撕下或折起

IBM

中国印刷

S152-0347-01