Oracle® Solaris 管理：设备和文件系统

文件号码 E25884
2012 年 2 月

ORACLE

版权所有 © 2004, 2012, Oracle 和/或其附属公司。 保留所有权利。

本软件和相关文档是根据许可证协议提供的，该许可证协议中规定了关于使用和公开本软件和相关文档的各种限制，并受知识产权法的保护。除非在许可证协议中明确许可或适用法律明确授权，否则不得以任何形式、任何方式使用、拷贝、复制、翻译、广播、修改、授权、传播、分发、展示、执行、发布或显示本软件和相关文档的任何部分。除非法律要求实现互操作，否则严禁对本软件进行逆向工程设计、反汇编或反编译。

此文档所含信息可能随时被修改，恕不另行通知，我们不保证该信息没有错误。如果贵方发现任何问题，请书面通知我们。

如果将本软件或相关文档交付给美国政府，或者交付给以美国政府名义获得许可证的任何机构，必须符合以下规定：

U.S. GOVERNMENT END USERS:

Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

本软件或硬件是为了在各种信息管理应用领域内的一般使用而开发的。它不应被应用于任何存在危险或潜在危险的应用领域，也不是为此而开发的，其中包括可能会产生人身伤害的应用领域。如果在危险应用领域内使用本软件或硬件，贵方应负责采取所有适当的防范措施，包括备份、冗余和其它确保安全使用本软件或硬件的措施。对于因在危险应用领域内使用本软件或硬件所造成的一切损失或损害，Oracle Corporation 及其附属公司概不负责。

Oracle 和 Java 是 Oracle 和/或其附属公司的注册商标。其他名称可能是各自所有者的商标。

Intel 和 Intel Xeon 是 Intel Corporation 的商标或注册商标。所有 SPARC 商标均是 SPARC International, Inc 的商标或注册商标，并应按照许可证的规定使用。AMD、Opteron、AMD 徽标以及 AMD Opteron 徽标是 Advanced Micro Devices 的商标或注册商标。UNIX 是 The Open Group 的注册商标。

本软件或硬件以及文档可能提供了访问第三方内容、产品和服务的方式或有关这些内容、产品和服务的信息。对于第三方内容、产品和服务，Oracle Corporation 及其附属公司明确表示不承担任何种类的担保，亦不对其承担任何责任。对于因访问或使用第三方内容、产品或服务所造成的任何损失、成本或损害，Oracle Corporation 及其附属公司概不负责。

120411@25097

目录

前言 17

1 管理可移除介质（概述） 21
可移除介质方面的新增功能 21
　　可移除介质管理的变更和改进 21
何处查找管理可移除介质任务 24
可移除介质的功能和优点 24
手动挂载和自动挂载的比较 25
访问可移除介质概述 25

2 管理可移除介质（任务） 27
管理可移除介质（任务列表） 27
准备可移除介质 28
　　可移除介质注意事项 28
　▼如何加载可移除介质 29
　▼如何格式化磁盘 (`rmformat`) 30
　▼如何在可移除介质上创建文件系统 31
　▼如何在 DVD-RAM 上创建文件系统 31
　▼如何检查可移除介质上的文件系统 32
　▼如何修复可移除介质上的坏块 32
　　将读/写保护和口令保护应用于可移除介质 33
　▼如何在可移除介质上启用或禁用写保护 33
　▼如何在可移除介质上启用或禁用读/写保护以及设置口令 33

3 访问可移除介质（任务） 35
访问可移除介质（任务列表） 35
访问可移除介质 36

使用可移除介质名称 36
访问可移除介质数据的指导原则 36
▼ 如何添加新的可移除介质驱动器 36
▼ 如何禁用或启用可移除介质服务 37
▼ 如何访问可移除介质上的信息 37
▼ 如何从可移除介质复制信息 38
▼ 如何确定可移除介质是否仍在使用 38
▼ 如何弹出可移除介质 39
在远程系统上访问可移除介质（任务列表） 40
▼ 如何使本地介质可供其他系统使用 40
▼ 如何访问远程系统上的可移除介质 41

4 写入 CD 和 DVD（任务） 43
使用音频 CD 以及数据 CD 和 DVD 43
CD/DVD 介质常用术语 44
写入 CD 和 DVD 数据及音频 CD 45
通过 RBAC 限制用户对可移除介质的访问 46
▼ 如何通过 RBAC 限制用户对可移除介质的访问 46
▼ 如何标识 CD 或 DVD 写入者 46
▼ 如何检查 CD 或 DVD 介质 47
创建数据 CD 或 DVD 47
▼ 如何为数据 CD 或 DVD 创建 ISO 9660 文件系统 48
▼ 如何创建多会话数据 CD 48
创建音频 CD 50
▼ 如何创建音频 CD 50
▼ 如何提取 CD 上的声道 51
▼ 如何复制 CD 52
▼ 如何删除 CD-RW 介质 53

5 管理设备（概述/任务） 55
在设备管理方面的新增功能? 55
定制驱动程序配置 56
Solaris PCI 资源管理器 56
新的 InfiniBand 管理功能 56
新的 InfiniBand 诊断工具和命令 57

新的 Ethernet over InfiniBand 设备 57
新的热插拔功能 57
设备命名增强功能 57
支持 PCI Express (PCIe) 58
有关其他设备管理任务的参考信息 59
管理 Oracle Solaris OS 中的设备 59
确定设备支持 59
关于设备驱动程序 59
设备的自动配置 60
显示设备配置信息 62
解决有故障的设备 66
向系统添加外围设备 67
▼ 如何添加外围设备 68
▼ 如何添加设备驱动程序 69
访问设备 69
如何创建设备信息 70
如何管理设备 70
设备命名约定 70
逻辑磁盘设备名称 71
逻辑磁带设备名称 73
逻辑可移除介质设备名称 74

6 动态配置设备（任务） 75
动态重新配置和热插拔 75
连接点 77
拆离 PCI 或 PCIe 适配卡 78
连接 PCI 或 PCIe 适配卡 78
使用 (`hotplug`) 命令进行 PCIe 热插拔 79
使用 `cfgadm` 命令执行 SCSI 热插拔（任务列表） 81
使用 `cfgadm` 命令进行 SCSI 热插拔 81
▼ 如何显示有关 SCSI 设备的信息 82
▼ 如何取消配置 SCSI 控制器 82
▼ 如何配置 SCSI 控制器 83
▼ 如何配置 SCSI 设备 83
▼ 如何断开 SCSI 控制器连接 84

▼ SPARC: 如何连接 SCSI 控制器 85
▼ SPARC: 如何将 SCSI 设备添加到 SCSI 总线中 85
▼ SPARC: 如何更换 SCSI 控制器上的相同设备 86
▼ SPARC: 如何移除 SCSI 设备 88
对 SCSI 配置问题进行故障排除 89
▼ 如何解决失败的 SCSI 取消配置操作 90
使用 cfgadm 命令执行 PCI 或 PCIe 热插拔（任务列表） 90
使用 cfgadm 命令执行 PCI 或 PCIe 热插拔 91
PCIe LED 指示灯行为 91
▼ 如何显示 PCI 插槽配置信息 91
▼ 如何移除 PCI 适配卡 93
▼ 如何添加 PCI 适配卡 94
对 PCI 配置问题进行故障排除 96
使用 cfgadm 命令进行 SATA 热插拔 96
▼ 如何取消配置 SATA 设备 97
▼ 如何配置 SATA 设备 97
重新配置协调管理器 (Reconfiguration Coordination Manager, RCM) 脚本概述 98
什么是 RCM 脚本? 98
RCM 脚本的功能 98
RCM 脚本进程的工作原理 98
RCM 脚本任务 99
应用程序开发者 RCM 脚本（任务列表） 99
系统管理员 RCM 脚本（任务列表） 100
命名 RCM 脚本 100
安装或删除 RCM 脚本 101
▼ 如何安装 RCM 脚本 101
▼ 如何删除 RCM 脚本 102
▼ 如何测试 RCM 脚本 102
磁带备份 RCM 脚本示例 102

7 使用 USB 设备（概述） 107
USB 设备的新增功能 107
USB 接口关联描述符支持 108
EHCI 同步传输支持 108
USB 设备可热插拔行为 108

x86: 在基于 GRUB 的引导中支持 USB CD 和 DVD 109
USB 虚拟键盘和鼠标支持 109
Oracle Solaris 对 USB 设备的支持 109
USB 设备概述 110
常用的 USB 首字母缩略词 110
USB 总线说明 111
关于 Oracle Solaris OS 中的 USB 115
USB 2.0 功能 115
USB 键盘和鼠标设备 117
USB 主机控制器和集线器 118
USB 电缆原则 119

8 使用 USB 设备（任务） 121
在 Oracle Solaris OS 中管理 USB 设备（任务列表） 121
使用 USB 海量存储设备（任务列表） 122
使用 USB 海量存储设备 123
使用 USB 磁盘设备 124
使用非符合 USB 海量存储设备 124
热插拔 USB 海量存储设备 125
准备使用 USB 海量存储设备 128
▼ 如何显示 USB 设备信息 129
▼ 如何在 USB 海量存储设备中创建文件系统 130
▼ 如何在 USB 海量存储设备中修改分区和创建 PCFS 文件系统 132
▼ 如何在 USB 海量存储设备中创建 Solaris 分区和修改分片 135
▼ 如何挂载或取消挂载 USB 海量存储设备 136
USB 海量存储设备的疑难解答提示 138
禁用特定的 USB 驱动程序 138
▼ 如何禁用特定的 USB 驱动程序 139
▼ 如何移除未使用的 USB 设备链接 139
使用 USB 音频设备（任务列表） 139
使用 USB 音频设备 140
热插拔多个 USB 音频设备 141
▼ 如何添加 USB 音频设备 141
▼ 如何确定系统的主音频设备 141
▼ 如何更改主 USB 音频设备 142

USB 音频设备问题疑难解答 142
使用 cfgadm 命令热插拔 USB 设备（任务列表） 143
使用 cfgadm 命令热插拔 USB 设备 143
▼ 如何显示 USB 总线信息 (cfgadm) 144
▼ 如何取消配置 USB 设备 145
▼ 如何配置 USB 设备 146
▼ 如何以逻辑方式断开 USB 设备连接 146
▼ 如何以逻辑方式连接 USB 设备 146
▼ 如何以逻辑方式断开 USB 设备子树连接 147
▼ 如何重置 USB 设备 147
▼ 如何更改多重配置 USB 设备的缺省配置 148

9 使用 InfiniBand 设备（概述/任务） 149
InfiniBand 设备概述 149
InfiniBand 软件包 151
动态重新配置 IB 设备（任务列表） 151
动态重新配置 IB 设备 (cfgadm) 153
取消配置 IB 设备考虑因素 153
▼ 如何显示 IB 设备信息 154
▼ 如何取消配置 IB Port、HCA_SVC 或 VPPA 设备 155
▼ 如何配置 IB 端口、HCA_SVC 或 VPPA 设备 156
▼ 如何取消配置 IB 伪设备 157
▼ 如何配置 IB 伪设备 157
▼ 如何显示 HCA 的内核 IB 客户机 157
▼ 如何在有活动 EoIB 设备的情况下动态重新配置 HCA 158
▼ 如何在热移除之后重新配置并恢复 EoIB 接口 159
配置 IB HCA 160
▼ 如何更新 IB p_key 表 160
▼ 如何显示 IB 通信服务 160
▼ 如何添加 VPPA 通信服务 160
▼ 如何删除现有 IB 端口、HCA_SVC 或 VPPA 通信服务 161
将 uDAPL 应用程序接口用于 InfiniBand 设备 161
▼ 如何启用 uDAPL 162
更新 DAT 静态注册表 163
管理 IPoIB 设备 (dladm) 164

▼如何显示物理数据链路信息 164
▼如何创建 IB 分区链路 164
▼如何显示 IB 分区链路信息 165
▼如何删除 IB 分区链路 166
监视并排除 IB 设备故障 166

10 管理磁盘（概述） 171
在磁盘管理方面的新增功能 171
通过物理位置标识设备 172
多磁盘扇区容量支持 174
支持在 2 TB 磁盘上安装和引导 Oracle Solaris OS 175
Solaris iSCSI 目标和启动器中的 iSNS 支持 175
Solaris COMSTAR iSCSI 支持 176
x86: GRUB 引导环境中的磁盘管理 176
支持容量超过 2 TB 的 SCSI 磁盘 176
有关磁盘管理任务的参考信息 177
磁盘管理概述 177
磁盘术语 177
关于磁盘标签 178
EFI 磁盘标签 178
关于磁盘分片 181
`format` 实用程序 182
对磁盘分区 184
分区表术语 185
显示分区表信息 185
使用浮动分片 (free hog slice) 187

11 管理磁盘（任务） 189
管理磁盘（任务列表） 189
确定系统中的磁盘 190
▼如何确定系统中的磁盘 190
格式化磁盘 192
▼如何确定磁盘是否已格式化 192
▼如何格式化磁盘 193
显示磁盘分片 194

▼如何显示磁盘分片信息 194
创建和检查磁盘标签 196
▼如何为磁盘设置标签 196
▼如何检查磁盘标签 200
恢复损坏的磁盘标签 202
▼如何恢复损坏的磁盘标签 202
添加第三方磁盘 204

12 SPARC：设置磁盘（任务） 205
SPARC: 为 ZFS 文件系统设置磁盘（任务列表） 205
SPARC: 为 ZFS 文件系统设置磁盘 206
▼SPARC: 如何为 ZFS 根文件系统设置磁盘 206
SPARC: 为 ZFS 根文件系统创建磁盘分片 207
▼SPARC: 如何为 ZFS 根文件系统创建磁盘分片 207
▼SPARC: 如何为 ZFS 根文件系统安装引导块 211
▼SPARC: 如何为 ZFS 文件系统设置磁盘 211

13 x86：设置磁盘（任务） 213
x86: 为 ZFS 文件系统设置磁盘（任务列表） 213
x86: 为 ZFS 文件系统设置磁盘 214
创建和更改 Solaris fdisk 分区 221
x86: 创建 `fdisk` 分区指南 221
▼x86: 如何创建 Solaris `fdisk` 分区 221
更改 `fdisk` 分区标识符 224
▼如何更改 Solaris `fdisk` 标识符 224

14 使用 COMSTAR 配置存储设备 227
COMSTAR 和 iSCSI 技术（概述） 227
COMSTAR 软件和硬件要求 229
配置 COMSTAR（任务列表） 229
配置 COMSTAR 230
使用 COMSTAR 配置 iSCSI 设备 232
▼如何启用 STMF 服务 232
▼如何备份和存储 COMSTAR 配置 233

▼如何创建 iSCSI LUN 233
▼如何创建 iSCSI 目标 235
▼如何为 iSER 配置 IB HCA 235
▼如何配置 iSCSI 启动器 236
▼如何删除搜索到的 iSCSI 目标 237
创建 iSCSI 目标门户组 239
▼如何访问 iSCSI 磁盘 240
使 SCSI 逻辑单元可用 241
▼如何使逻辑单元可用于所有系统 242
▼如何限制 LUN 访问选中的系统 242
使用 COMSTAR 配置光纤通道设备 243
为 COMSTAR 配置光纤通道端口 243
使逻辑单元可用于 FC 和 FCoE 246
使用 COMSTAR 配置 FCoE 设备 247
配置 FCoE 端口 248
启用以太网接口上的 802.3x PAUSE 和巨型帧 (Jumbo Frame) 248
▼如何创建 FCoE 目标端口 248
▼如何验证 FCoE 目标端口正在运行 249
▼如何删除 FCoE 目标端口 250
使用 COMSTAR 配置 SRP 设备 250
结合 SRP 使用 COMSTAR 视图 250
▼如何启用 SRP 目标服务 251
▼如何验证 SRP 目标状态 251
在基于 iSCSI 的存储网络中配置验证 251
▼如何配置对 iSCSI 启动器的 CHAP 验证 252
▼如何配置对 iSCSI 目标的 CHAP 验证 253
使用第三方 RADIUS 服务器简化 iSCSI 配置中的 CHAP 管理 254
设置 Oracle Solaris iSCSI 多路径设备 256
▼如何为目标启用多个 iSCSI 会话 257
监视 iSCSI 配置 259
▼如何显示 iSCSI 配置信息 259
修改 iSCSI 启动器和目标参数 261
调优 iSCSI 参数 261
▼如何修改 iSCSI 启动器和目标参数 263
iSCSI 配置问题的故障排除 266
从本地系统到 iSCSI 目标无连接 266

iSCSI 设备或磁盘在本地系统上不可用 267
在使用 iSNS 搜索方法时使用 LUN 屏蔽 268
一般 iSCSI 错误消息 268

15 配置和管理 Oracle Solaris Internet 存储名称服务 (Internet Storage Name Service, iSNS) 273
iSNS 技术（概述） 273
配置 iSNS 服务器 274
设置 iSNS 管理设置 275
使用命令行界面配置 iSNS 277
管理 iSNS 服务器和客户端 280
▼ 如何显示搜索域集的状态 281
▼ 如何显示搜索域的状态 281
▼ 如何显示客户端的状态 281
▼ 如何从搜索域中删除客户端 281
▼ 如何从搜索域集中删除搜索域 282
▼ 如何禁用搜索域集 282
▼ 如何删除搜索域集 282

16 format 实用程序（参考） 283
有关使用 format 实用程序的建议和要求 283
format 菜单和命令说明 284
partition 菜单 285
x86: fdisk 菜单 286
analyze 菜单 287
defect 菜单 288
format 命令的输入规则 289
为 format 命令指定数字 289
指定 format 命令名称 290
为 format 命令指定磁盘名称 290
获取有关 format 实用程序的帮助 290

17 管理文件系统（概述） 293
Oracle Solaris 文件系统中的新增功能 293
文件系统监视工具 (fsstat) 293

Oracle Solaris ZFS 文件系统 294
有关文件系统管理任务的参考信息 294
文件系统概述 295
Oracle Solaris 文件系统的类型 295
缺省的 Oracle Solaris 文件系统 300
挂载和取消挂载文件系统概述 301
已挂载文件系统表 301
虚拟文件系统表 301
NFS 环境 302
自动挂载 (autofs) 303
Oracle Solaris SMB 服务 303
确定文件系统的类型 304
如何确定文件系统的类型 304

18 创建和挂载文件系统（任务） 305
创建 Oracle Solaris 文件系统 305
创建 ZFS 文件系统 305
创建临时文件系统 306
创建 LOFS 文件系统 306
挂载和取消挂载 Oracle Solaris 文件系统 306
/etc/vfstab 文件的字段说明 307
取消挂载 Oracle Solaris 文件系统的先决条件 309
创建和挂载 Oracle Solaris 文件系统 309
▼ 如何创建 ZFS 文件系统 309
▼ 如何创建和挂载传统 UFS 文件系统 310
▼ 如何创建和挂载 TMPFS 文件系统 311
▼ 如何创建和挂载 LOFS 文件系统 312
▼ 如何向 /etc/vfstab 文件添加项 313
▼ 如何挂载文件系统（/etc/vfstab 文件） 314
▼ 如何挂载 NFS 文件系统（mount 命令） 315
▼ x86: 如何从硬盘挂载 PCFS (DOS) 文件系统（mount 命令） 315
▼ 如何停止所有进程访问文件系统 316
▼ 如何取消挂载文件系统 317

19 配置附加交换空间（任务） 319
关于交换空间 319
交换空间与虚拟内存 319
交换空间与 TMPFS 文件系统 320
交换空间与转储设备配置 320
交换空间和动态重新配置 321
在 SAN 环境中配置交换空间 321
如何了解是否需要更多交换空间? 322
与交换相关的错误消息 322
与 TMPFS 相关的错误消息 322
如何分配交换空间 323
交换区域和 /etc/vfstab 文件 323
规划交换空间 323
为基于 ZFS 的系统分配交换空间 324
监视交换资源 324
在 Oracle Solaris ZFS 根环境中添加或更改交换空间 326
▼ 如何在 Oracle Solaris ZFS 根环境中添加交换空间 326

20 复制文件和文件系统（任务） 327
用于复制文件系统的命令 327
在文件系统之间复制目录（cpio 命令） 329
▼ 如何在文件系统之间复制目录 (cpio) 329
将文件和文件系统复制到磁带 330
将文件复制到磁带（tar 命令） 330
▼ 如何将文件复制到磁带 (tar) 330
▼ 如何列出磁带中的文件 (tar) 331
▼ 如何从磁带中检索文件 (tar) 332
使用 pax 命令将文件复制到磁带 333
▼ 如何将文件复制到磁带 (pax) 333
使用 cpio 命令将文件复制到磁带 333
▼ 如何将目录中的所有文件复制到磁带 (cpio) 333
▼ 如何列出磁带中的文件 (cpio) 335
▼ 如何从磁带中检索所有文件 (cpio) 335
▼ 如何从磁带中检索特定文件 (cpio) 336
将文件复制到远程磁带设备 337

▼ 如何将文件复制到远程磁带设备（tar 和 dd） 337
▼ 如何从远程磁带设备中提取文件 338

21 管理磁带机（任务） 341
选择要使用的介质 341
备份设备名称 342
指定磁带机的反绕选项 343
指定磁带机的不同密度 343
显示磁带机状态 344
▼ 如何显示磁带机状态 344
处理盒式磁带 345
重新张紧盒式磁带 345
反绕盒式磁带 345
磁带机维护和介质处理指南 345

索引 347

前言

《系统管理指南：设备和文件系统》是包含 Oracle Solaris 系统管理信息重要内容的一套书中的一本。该指南包含基于 SPARC 和基于 x86 的系统的信息。

本书假设您已经完成以下任务：

- 安装 Oracle Solaris 11 操作系统
- 设置了任何计划使用的网络软件

对于 Oracle Solaris 11 发行版，系统管理员感兴趣的新增功能已在相应各章的名为“...的新增功能”的各节中加以介绍。

注 – 此 Oracle Solaris 发行版支持使用 SPARC 和 x86 系列处理器体系结构的系统。支持的系统可以在 http://www.oracle.com/webfolder/technetwork/hcl/index.html 上的 Oracle Solaris Hardware Compatibility List（Oracle Solaris 硬件兼容性列表）中找到。本文档列举了在不同类型的平台上进行实现时的所有差别。

目标读者

本书适用于所有负责管理一个或多个运行 Oracle Solaris 11 发行版的系统的人员。要使用本书，您应当具备 1 到 2 年的 UNIX 系统管理经验。参加 UNIX 系统管理培训课程可能会对您有所帮助。

系统管理指南系列书籍的结构

下表列出了系统管理指南系列中各本书包含的主题。

书名	主题
《在 SPARC 平台上引导和关闭 Oracle Solaris》	引导和关闭系统、管理引导服务、修改引导行为、从 ZFS 引导、管理引导归档文件以及在 SPARC 平台上进行引导故障排除
《在 x86 平台上引导和关闭 Oracle Solaris》	引导和关闭系统、管理引导服务、修改引导行为、从 ZFS 引导、管理引导归档文件以及在 x86 平台上进行引导故障排除

书名	主题
《Oracle Solaris 管理：常见任务》	使用 Oracle Solaris 命令、引导和关闭系统、管理用户帐户和组、管理服务、硬件错误、系统信息、系统资源和系统性能、管理软件、打印、控制台和终端及对系统和软件问题进行故障排除
《Oracle Solaris 管理：设备和文件系统》	可移除介质、磁盘和设备、文件系统以及备份和还原数据
《Oracle Solaris 管理：IP 服务》	TCP/IP 网络管理、IPv4 和 IPv6 地址管理、DHCP、IPsec、IKE、IP 过滤器和 IPQoS
《Oracle Solaris Administration: Naming and Directory Services》	DNS、NIS 和 LDAP 命名与目录服务，包括从 NIS 转换至 LDAP
《Oracle Solaris 管理：网络接口和网络虚拟化》	自动和手动 IP 接口配置包括 WiFi 无线、网桥管理、VLAN、聚合、LLDP 和 IPMP、虚拟 NIC 和资源管理。
《Oracle Solaris 管理：网络服务》	Web 高速缓存服务器、与时间相关的服务、网络文件系统（NFS 和 Autofs）、邮件、SLP 和 PPP
《Oracle Solaris 管理：Oracle Solaris Zones、Oracle Solaris 10 Zones 和资源管理》	资源管理功能，使用此功能可控制应用程序使用可用系统资源的方式；Oracle Solaris Zones 软件分区技术，用于虚拟化操作系统服务以便为运行的应用程序创建独立环境；Oracle Solaris 10 Zones，用于托管运行于 Oracle Solaris 11 内核的 Oracle Solaris 10 环境
《Oracle Solaris 管理：安全服务》	审计，设备管理，文件安全，BART，Kerberos 服务，PAM，加密框架，密钥管理，特权，RBAC，SASL，安全 Shell 和病毒扫描
《Oracle Solaris Administration: SMB and Windows Interoperability》	SMB 服务，使您能够配置 Oracle Solaris 系统让 SMB 客户端能够共享 SMB；SMB 客户端，你您能够访问共享 SMB；以及本机标识映射服务，使您能够在 Oracle Solaris 系统和 Windows 系统之间映射用户和组标识
《Oracle Solaris 管理：ZFS 文件系统》	ZFS（Zettabyte 文件系统）存储池以及文件系统的创建和管理、快照、克隆、备份、使用访问控制列表 (Access Control List, ACL) 保护 ZFS 文件、在安装区域的 Solaris 系统中使用 ZFS、仿真卷以及故障排除和数据恢复
《Trusted Extensions 配置和管理》	专用于 Trusted Extensions 的系统安装、配置和管理
《Oracle Solaris 11 安全准则》	保护 Oracle Solaris 系统及其安全性功能的使用方案（如区域、ZFS 和 Trusted Extensions）
《从 Oracle Solaris 10 转换到 Oracle Solaris 11》	在安装、设备、磁盘和文件系统管理、软件管理、网络、系统管理、安全性、虚拟化、桌面功能、用户帐户管理和用户环境方面为从 Oracle Solaris 10 转换至 Oracle Solaris 11 提供系统管理信息和示例

获取 Oracle 支持

Oracle 客户可以通过 My Oracle Support 获取电子支持。有关信息，请访问 http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info，或访问 http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs（如果您听力受损）。

印刷约定是什么

下表介绍了本书中的印刷约定。

表 P–1 印刷约定

字体或符号	含义	示例
`AaBbCc123`	命令、文件和目录的名称；计算机屏幕输出	编辑 `.login` 文件。 使用 `ls -a` 列出所有文件。 `machine_name% you have mail.`
`AaBbCc123`	用户键入的内容，与计算机屏幕输出的显示不同	`machine_name%` **`su`** `Password:`
AaBbCc123	命令行占位符：要使用实名或值替换	要删除文件，请键入 **`rm`** *filename*。
AaBbCc123	书名，新词或术语，或者需要强调的词	阅读《用户指南》的第 6 章。 这些称为 *class* 选项。 尚**未**保存更改。

命令中的 shell 提示符示例

下表显示了 C shell、Bourne shell 和 Korn shell 的缺省系统提示符和超级用户（或管理）提示符。

表 P–2 Shell 提示符

Shell	提示符
C shell	`machine_name%`
C shell 超级用户	`machine_name#`
Bourne shell 和 Korn shell	`$`
Bourne shell 和 Korn shell 超级用户	`#`

◆ ◆ ◆ 第 1 章

1 管理可移除介质（概述）

本章提供有关在 Oracle Solaris OS 中管理可移除介质的一般原则。

以下是本章中概述信息的列表。

- 第 页中的 “可移除介质方面的新增功能”
- 第 页中的 “何处查找管理可移除介质任务”
- 第 页中的 “可移除介质的功能和优点”
- 第 页中的 “手动挂载和自动挂载的比较”
- 第 页中的 “访问可移除介质概述”

可移除介质方面的新增功能

可移除介质管理的变更和改进

Oracle Solaris 11：之前管理可移除介质的功能已经删除，更换了新的服务和方法，能够更好地管理可移除介质。

提供了下列新功能：

- 新的可移除介质服务由 SMF 管理。

```
online         12:17:54 svc:/system/hal:default
online         12:17:56 svc:/system/filesystem/rmvolmgr:default
online         12:17:26 svc:/system/dbus:default
```

- 现在，可移除介质自动挂载到 `/media` 目录下。但是，为了实现兼容性，从以前的介质挂载点 `/cdrom` 和 `/rmdisk` 提供指向 `/media` 的符号链接。

  例如，按以下方式挂载紧凑型闪存卡 (`/dev/dsk/c4d0p0:1`)：

```
$ ls /media/NIKON
```

  例如，按以下方式挂载 USB 内存条 (`/dev/dsk/c3t0d0s0`)：

```
$ ls /media/U3
```

- 缺省的可移除介质卷管理器 rmvolmgr 负责进行以下操作：
  - 挂载和取消挂载卷。
  - rmvolmgr 的根实例在系统引导时启动。但是，可以将会话配置文件配置为在您登录时启动 rmvolmgr 实例。在用户会话中运行 rmvolmgr 时，其仅挂载由当前用户或会话拥有的设备，因此不与根实例相冲突。
  - 如果存在 rmvolmgr，它将取消挂载所有由它挂载的介质。
  - 为了实现兼容性，rmvolmgr 会在 /cdrom 或 /rmdisk 目录下创建指向 /media 下的实际挂载点的符号链接。
  - 为了实现 CDE 兼容性，可以使用特殊的 rmvolmgr 运行模式。
- 硬件抽象层 (hardware abstraction layer, HAL) 守护进程 hald 提供了连接到系统的设备的视图。硬件配置更改（通过热插拔或其他机制）时，该视图会自动更新。

  HAL 将硬件表示为设备对象。设备对象由唯一设备标识符 (unique device identifier, UDI) 标识，并且具有一组称为设备属性的关键字-值对。有些属性源自实际硬件，有些属性由设备信息文件（.fdi 文件）合并而来，有些属性与实际设备配置相关。

删除了下列功能：

- 本发行版仅支持 USB 磁盘。
- vold 守护进程、volfs 文件系统和 volfs 服务已经被删除。

  ```
  svc:/system/filesystem/volfs
  ```
- 不再提供 /vol 目录下的可移除介质的逻辑设备名称，例如 /vol/dev/rdsk/... 或 /vol/dev/aliases/...。

  要通过可移除介质的逻辑设备名称来访问可移除介质，应使用 /dev 设备。例如：

  ```
  /dev/rdsk/c0t6d0s2
  ```
- 有些 vold 设备昵称不再可用。以下 eject -l 输出列出了每个设备的可用设备昵称，在本示例中，为挂载的介质路径名 (/media/SOL_11_X86_4)：

  ```
  $ eject -l
  /dev/dsk/c2t0d0s2    cdrom,cdrom0,cd,cd0,sr,sr0,SOL_11_X86_4,/media/SOL_11_X86_4
  ```

  以逗号分隔的列表显示了可用于弹出每个设备的昵称。
- 在 vold.conf 和 rmmount.conf 中所做的定制不再可用，这是因为这些配置文件不存在。有关管理介质定制的信息，请参见第　　页中的 “定制可移除介质管理”。
- 以 vol* 开头的命令（volcheck 和 volrmmount 除外）。

向后兼容性

以下功能通过先前的 Oracle Solaris 可移除介质功能提供向后兼容性：

- 可移除介质挂载点已移到 /media 目录中，该目录用来挂载可移除介质，如 CD-ROM 和 USB 设备。为了实现兼容性，提供了从以前的介质挂载点（如 /cdrom 和 /rmdisk）指向 /media 的符号链接。
- rmformat 命令仍然可用。该命令的输出与在以前的 Solaris 发行版中禁用 vold 时该命令的输出一样。

  例如：

```
# rmformat
Looking for devices...
     1. Logical Node: /dev/rdsk/c0t6d0s2
        Physical Node: /pci@1f,4000/scsi@3/sd@6,0
        Connected Device: TOSHIBA  DVD-ROM SD-M1401 1009
        Device Type: DVD Reader
        Bus: SCSI
        Size: 2.9 GB
        Label: <None>
        Access permissions: <Unknown>
```

- eject 命令可用，但已得到增强。有关更多信息，请参见第　　页中的 “弹出可移除介质”。

挂载和取消挂载可移除介质

在此发行版中删除了大多数以 vol* 开头的命令。可以使用修改版本的 rmmount 和新的 rmumount 命令来挂载和取消挂载可移除介质。

可以使用这些命令按设备名称、标签或挂载点进行挂载。例如，要挂载 iPod：

```
% rmmount ipod
```

例如，要取消挂载 DVD 上的文件系统：

```
# rmumount cdrom
cdrom /dev/dsk/c0t6d0s5 unmounted
cdrom /dev/dsk/c0t6d0s0 unmounted
```

有关更多信息，请参见 rmmount(1M)。

弹出可移除介质

与以前的 Solaris 发行版一样，使用 eject 命令可取消挂载并弹出可移除介质。不过，可以使用以下 eject 选项：

-f　强制设备弹出，即使设备忙也是如此。

-l　显示可弹出的设备的路径和昵称。

-t　为设备提供 CD-ROM 托盘闭合命令。并非所有设备都支持此选项。

例如，要按卷标弹出可移除介质：

```
% eject mypictures
```

有关更多信息，请参见eject(1)。

定制可移除介质管理

对于 vold.conf 和 rmmount.conf 文件中可用的大多数定制，您需要使用桌面卷管理器首选项或者修改 .fdi 文件。

- 对于 rmmount.conf 操作，您需要使用桌面卷管理器操作 gconf 或 HAL 调用。
- 以前，能够以超级用户身份代表普通用户运行 rmmount.conf 操作。现在，通过在 /usr/lib/hal 目录中安装调用可执行文件来实现此目的。

禁用可移除介质功能

要阻止卷在用户会话之外挂载，请禁用 rmvolmgr 服务。例如：

```
# svcadm disable rmvolmgr
```

何处查找管理可移除介质任务

请参见以下内容，以查找管理可移除介质的逐步说明。

可移除介质管理任务	更多信息
格式化可移除介质	第 2 章，管理可移除介质（任务）
访问可移除介质	第 3 章，访问可移除介质（任务）
写数据 CD 和 DVD 以及音乐 CD	第 4 章，写入 CD 和 DVD（任务）

可移除介质的功能和优点

Oracle Solaris 发行版为用户和软件开发者提供了用于处理可移除介质的标准接口。可移除介质服务提供以下优点：

- 自动挂载可移除介质。有关手动挂载和自动挂载的比较，请参见下一节。
- 不必成为管理员即可访问可移除介质。
- 允许您为网络中的其他系统提供对本地系统上任何可移除介质的自动访问。有关更多信息，请参见第 3 章，访问可移除介质（任务）。

手动挂载和自动挂载的比较

下表对手动挂载（不使用可移除介质服务）和自动挂载（使用可移除介质管理）可移除介质所涉及的步骤进行了比较。

表 1–1 手动挂载和自动挂载可移除介质的比较

步骤	手动挂载	自动挂载
1	插入介质。	插入介质。
2	成为管理员。	对于 USB 磁盘，使用 `volcheck` 命令。
3	确定介质设备的位置。	可移除介质服务自动执行以前手动挂载和使用可移除介质所需的许多任务。
4	创建挂载点。	
5	确保当前不在挂载点目录中。	
6	挂载设备并使用适当的 `mount` 选项。	
7	退出管理员帐户。	
8	使用介质上的文件。	使用介质上的文件。
9	成为管理员。	
10	取消挂载介质设备。	
11	弹出介质。	弹出介质。
12	退出管理员帐户。	

访问可移除介质概述

在本质上，使用可移除介质服务可以访问可移除介质，就像手动挂载那样，但是前者更容易且不需要管理员访问权限。

如果介质含有文件系统和标签，那么介质标签名称会用来命名 */media/pathname* 挂载点。如果标签不存在，那么磁盘型号名称会用来命名该介质，例如 `/media/cdrom`。通用 *nickname* 仅用于传统符号链接。例如，`/rmdisk/rmdisk0`。

如果系统具有多种类型的可移除设备，请参见下表以了解其访问点。

表 1–2 如何访问可移除介质上的数据

访问	插入	无标签的介质路径名称	有标签的介质路径名称示例
可移除硬盘上的文件	命令行上可移除硬盘和类型 `volcheck`	`/media/`*usb-disk* 或传统路径 `/rmdisk/rmdisk0`	`/media/00JB-00CRA0`

表 1–2 如何访问可移除介质上的数据 （续）

访问	插入	无标签的介质路径名称	有标签的介质路径名称示例
DVD 上的文件	DVD，然后等待几秒	`/media/cdrom`	`/media/sol_10_sparc`
DVD 上的文件	DVD，然后等待几秒	`/media/cdrom`	`/media/SOL_11_X86`

可以使用 `rmmount -l` 命令确定系统上已挂载的介质。例如：

```
# rmmount -l
/dev/dsk/c5t0d0p0       rmdisk6,/media/FD-05PUB
/dev/dsk/c4t0d3p0       rmdisk5,/media/223UHS-SD-MMC
/dev/dsk/c2t0d0s2       cdrom1,cd1,sr1,SOL_11_X86,/media/SOL_11_X86
/dev/dsk/c3t0d0p0       rmdisk2,/media/00JB-00CRA0
```

在以上输出中，已挂载的设备为：

`/dev/dsk/c5t0d0p0` USB 磁盘

`/dev/dsk/c4t0d3p0` USB 读卡器中的 CF 卡

`/dev/dsk/c2t0d0s2` DVD-ROM

`/dev/dsk/c3t0d0p0` 可移除 USB 磁盘

◆ ◆ ◆ 第 2 章

2 管理可移除介质（任务）

本章介绍在 Oracle Solaris OS 中如何从命令行管理可移除介质。

与管理可移除介质相关的过程信息，请参见第 27 页中的 “管理可移除介质（任务列表）”。有关可移除介质的背景信息，请参见第 1 章，管理可移除介质（概述）。

管理可移除介质（任务列表）

以下任务列表介绍了有关管理可移除介质的任务。

任务	说明	参考
1. 装入介质。	将磁盘插入驱动器，然后键入 `volcheck` 命令。	第 29 页中的 “如何加载可移除介质”
2.（可选）格式化磁盘。	格式化磁盘。	第 30 页中的 “如何格式化磁盘 (`rmformat`)”
3.（可选的）添加 PCFS 文件系统。	添加 PCFS 文件系统以使用该介质传输文件。	第 31 页中的 “如何在可移除介质上创建文件系统”
	将 UDFS 文件系统添加到 DVD-RAM 设备。	第 31 页中的 “如何在 DVD-RAM 上创建文件系统”
4.（可选的）检查介质。	验证介质上文件系统的完整性。	第 32 页中的 “如何检查可移除介质上的文件系统”
5.（可选的）修复介质上的坏块。	如有必要，请修复介质上的任何坏块。	第 32 页中的 “如何修复可移除介质上的坏块”
6.（可选的）应用读/写保护和口令保护。	如有必要，请在介质上应用读/写保护或口令保护。	第 33 页中的 “如何在可移除介质上启用或禁用写保护”

准备可移除介质

以下各节介绍如何准备可移除介质，以供使用。

可移除介质注意事项

操作磁盘时请注意以下事项：

- SunOS 文件系统格式包括基本的“位”格式设置和用于支持 SunOS 文件系统的结构。DOS 文件系统的完整格式包括基本的“位”格式设置和用于支持 MS-DOS 或 NEC-DOS 文件系统的结构。为每种类型的文件系统准备介质的过程是不同的。因此，在格式化磁盘之前，请考虑要执行哪个过程。有关更多信息，请参见第 27 页中的 “管理可移除介质（任务列表）”。
  - 有关可移除介质名称的信息，请参见第 36 页中的 “使用可移除介质名称”。
  - 对于未命名的磁盘（即没有“标签”的磁盘），将为其指定缺省名称 `unnamed_floppy`。
  - 对于未命名的磁盘（即没有“标签”的磁盘），将为其指定缺省名称 `unnamed_floppy`。

  Oracle Solaris 系统可以为以下文件系统类型执行格式化操作：

  - ZFS 或 UFS
  - MS-DOS 或 NEC-DOS (PCFS)
  - UDFS

  在 Oracle Solaris 系统（SPARC 或 x86）上，可以格式化以下密度的磁盘。

磁盘大小	磁盘密度	容量
3.5”	高密度 (High density, HD)	1.44 MB
3.5”	双倍密度 (Double density, DD)	720 KB

  缺省情况下，磁盘驱动器会将磁盘格式化为相近的密度。此缺省设置意味着，除非另行指定，否则 1.44 MB 驱动器尝试将磁盘格式化为 1.44 MB，而不管该磁盘实际上是否为 1.44 MB 磁盘。换言之，磁盘可以格式化为其自身容量或更低容量，驱动器可以按自身容量或更低容量格式化磁盘。

格式化磁盘

可以使用 `rmformat` 命令格式化和保护可重写磁盘。文件系统自动挂载。因此，如果介质包含现有文件系统，则您可能必须先卸载介质然后才能格式化介质。

`rmformat` 命令具有以下三个格式化选项：

- quick－此选项在格式化磁盘时不进行检查，或只对介质的特定磁道进行有限的检查。
- long－此选项完整地格式化磁盘。对于某些设备，使用此选项时，驱动器可能会对整个介质进行检查。
- force－此选项执行完整格式化，且不需要用户确认。对于具有口令保护机制的磁盘，此选项会在格式化之前清除口令。此功能在忘记口令的情况下很有用。在没有口令保护的磁盘上，此选项强制执行 long 格式化。

▼ 如何加载可移除介质

有关可移除介质硬件注意事项的信息，请参见第 28 页中的 “可移除介质注意事项”。

1 **插入介质。**

2 **确保介质已格式化。**

如果不确定，请插入介质并检查系统控制台窗口中的状态消息，如步骤 3中所述。如果需要格式化介质，请转到第 30 页中的 “如何格式化磁盘 (rmformat)”。

3 **可选如果使用的是传统的非 USB 磁盘设备，请通知卷管理。**

```
$ volcheck -v
```

可能会显示两种状态消息：

media was found

卷管理检测到介质并试图将其挂载到第 36 页中的 “使用可移除介质名称”中所述的目录。

如果介质已正确格式化，则控制台不会显示任何错误消息。

如果介质未格式化，则仍会显示 “media was found” 消息。但是，系统控制台窗口中会显示类似以下内容的错误消息：

```
fd0: unformatted diskette or no diskette in the drive

fd0: read failed (40 1 0)

fd0: bad format
```

必须先格式化介质，卷管理才能挂载它。有关更多信息，请参见第 2 章，管理可移除介质（任务）。

no media was found

卷管理未检测到介质。请确保介质已正确插入，然后再次运行 volcheck。如果不成功，请检查介质，介质可能已损坏。还可以尝试手动挂载介质。

4 **通过列出介质的内容，验证介质是否已挂载。**

例如，为磁盘执行以下操作：

```
$ ls /media/floppy
lost+found myfiles
```

▼ 如何格式化磁盘 (rmformat)

可以使用 `rmformat` 命令格式化磁盘。缺省情况下，此命令会在介质上创建两个分区：分区 0 和分区 2（整个介质）。

1 **验证可移除介质是否正在运行。如果正在运行，则可以使用设备名称的较短昵称。**

```
# svcs hal dbus rmvolmgr
STATE          STIME    FMRI
online         Apr_09   svc:/system/dbus:default
online         Apr_09   svc:/system/hal:default
online         Apr_09   svc:/system/filesystem/rmvolmgr:default
```

有关重新启动可移除介质服务的信息，请参见第 37 页中的 “如何禁用或启用可移除介质服务”。有关标识介质设备名称的信息，请参见第 36 页中的 “使用可移除介质名称”。

2 **格式化磁盘。**

```
$ rmformat -F [ quick | long | force ] device-name
```

有关 `rmformat` 格式化选项的更多信息，请参见第 28 页中的 “格式化磁盘”。

如果 `rmformat` 输出指明存在坏块，请参见第 32 页中的 “如何修复可移除介质上的坏块”。

3 **（可选）使用 8 个字符的标签来标记磁盘。**

```
$ rmformat -b label device-name
```

有关创建 DOS 标签的信息，请参见 `mkfs_pcfs(1M)`。

示例 2–1 格式化磁盘

以下示例说明如何格式化磁盘。

```
$ rmformat -F quick /dev/rdiskette
Formatting will erase all the data on disk.
Do you want to continue? (y/n) y
.....................................................................
```

▼ 如何在可移除介质上创建文件系统

1 **可选如有必要，格式化磁盘。**

要格式化 USB 磁盘，请使用类似以下内容的语法：

```
$ rmformat -F long /dev/rdsk/c11t0d0p0
```

2 **（可选的）创建备用 Solaris 分区表。**

```
$ rmformat -s slice-file device-name
```

将显示类似以下内容的分片文件样例：

```
slices: 0 = 0, 30MB, "wm", "home" :
             1 = 30MB, 51MB :
             2 = 0, 94MB, "wm", "backup" :
             6 = 81MB, 13MB
```

3 **成为管理员。**

有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。

4 **确定相应的文件系统类型，然后选择以下操作之一：**

- 创建 PCFS 文件系统。例如：

  ```
  # mkfs -F pcfs -o nofdisk,size=9800 /dev/rdsk/c11t0d0p0
  ```

- 创建 UDFS 文件系统。例如：

  ```
  # mkfs -F udfs /dev/rdsk/c0t1d0p0
  ```

▼ 如何在 DVD-RAM 上创建文件系统

使用本程序在 DVD-RAM 上创建文件系统。

1 **成为管理员。**

有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。

2 **在 DVD-RAM 设备上创建文件系统。**

- 创建 UDFS 文件系统。例如：

  ```
  # mkfs -F udfs /dev/rdsk/c0t0d0s2
  ```

3 **挂载文件系统。**

- 挂载 UDFS 文件系统。例如：

  ```
  # mount -F udfs /dev/dsk/c0t0d0s2 /mnt
  ```

4 **验证是否可以读取或写入文件系统。**

5 **完成后，弹出 DVD-RAM。**

▼ 如何检查可移除介质上的文件系统

1 **成为管理员。**
有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。

2 **确定文件系统类型，然后选择以下操作之一：**

- 检查 UDFS 文件系统。

  ```
  # fsck -F udfs device-name
  ```

- 检查 PCFS 文件系统。

  ```
  # fsck -F pcfs device-name
  ```

示例 2–2 检查可移除介质上的 PCFS 文件系统

以下示例说明如何检查介质上的 PCFS 文件系统的一致性。

```
# fsck -F pcfs /dev/rdsk/c0t4d0s2
** /dev/rdsk/c0t4d0s2
** Scanning file system meta-data
** Correcting any meta-data discrepancies
1457664 bytes.
0 bytes in bad sectors.
0 bytes in 0 directories.
0 bytes in 0 files.
1457664 bytes free.
512 bytes per allocation unit.
2847 total allocation units.
2847 available allocation units.
```

▼ 如何修复可移除介质上的坏块

如果驱动器支持坏块管理，则只能使用 `rmformat` 命令来验证、分析和修复在验证过程中找到的坏扇区。大多数 USB 内存条都不支持坏块管理。

如果驱动器支持坏块管理，则会尽最大努力纠正坏块。如果尽最大努力也无法纠正坏块，则会显示一条消息，指明修复介质失败。

1 **修复可移除介质上的坏块。**

```
$ rmformat -c block-numbers device-name
```

提供以前的 rmformat 会话中的十进制、八进制或十六进制格式的块编号。

2 验证介质。

```
$ rmformat -V read device-name
```

将读/写保护和口令保护应用于可移除介质

在支持此功能的可移除介质上，可以应用读保护或写保护，并设置口令。

▼ 如何在可移除介质上启用或禁用写保护

1 确定要启用还是禁用写保护，然后选择以下操作之一：

- 启用写保护。

  ```
  $ rmformat -w enable device-name
  ```

- 禁用写保护。

  ```
  $ rmformat -w disable device-name
  ```

2 验证是否已启用或禁用介质的写保护。

```
$ rmformat -p device-name
```

▼ 如何在可移除介质上启用或禁用读/写保护以及设置口令

对于支持此功能的可移除介质，可以应用最多为 32 个字符的口令。

如果尝试在不支持此功能的介质上应用口令，则会收到一条警告消息。

1 确定是要启用还是禁用读/写保护，并设置口令。选择以下操作之一：

- 启用读保护或写保护。

  ```
  $ rmformat -W enable device-name
  Please enter password (32 chars maximum): xxx
  Please reenter password:

  $ rmformat -R enable device-name
  Please enter password (32 chars maximum): xxx
  Please reenter password:
  ```

- 禁用读保护或写保护并删除口令。

  ```
  $ rmformat -W disable device-name
  Please enter password (32 chars maximum): xxx
  ```

```
$ rmformat -R disable device-name
Please enter password (32 chars maximum): xxx
```

2 验证是否已启用或禁用介质的读/写保护。

```
$ rmformat -p device-name
```

◆ ◆ ◆ 第 3 章

3 访问可移除介质（任务）

本章介绍在 Oracle Solaris OS 中如何从命令行访问可移除介质。

有关与访问可移除介质关联的过程的信息，请参见以下内容：

- 第 35 页中的“访问可移除介质（任务列表）”
- 第 40 页中的“在远程系统上访问可移除介质（任务列表）”

有关可移除介质的背景信息，请参见第 1 章，管理可移除介质（概述）。

访问可移除介质（任务列表）

以下任务列表介绍了有关访问可移除介质的任务。

任务	说明	参考
1.（可选的）添加可移除介质驱动器。	如有必要，请将可移除介质驱动器添加到系统。	第 36 页中的“如何添加新的可移除介质驱动器”
2.（可选的）确定是在运行还是在不运行卷管理 (vold) 的情况下使用可移除介质。	缺省情况下，将运行卷管理 (vold)。确定是在运行还是在不运行卷管理的情况下使用可移除介质。	第 37 页中的“如何禁用或启用可移除介质服务”
3. 访问可移除介质。	在运行或不运行卷管理的情况下访问不同种类的可移除介质。	第 37 页中的“如何访问可移除介质上的信息”
4.（可选的）复制文件或目录。	从介质复制文件或目录，类似于从文件系统中的其他任何位置进行复制。	第 38 页中的“如何从可移除介质复制信息”
5. 确定介质是否仍在使用。	弹出介质之前，确定其是否仍在使用。	第 38 页中的“如何确定可移除介质是否仍在使用”

任务	说明	参考
6. 弹出介质。	完成后，从驱动器中弹出介质。	第 39 页中的“如何弹出可移除介质”

访问可移除介质

无论是否使用卷管理，都可以访问有关可移除介质的信息。有关使用 GNOME 的文件管理器访问可移除介质相关信息的信息，请参见 GNOME 桌面文档。

使用可移除介质名称

现在，可移除介质自动挂载到 `/media` 目录下。但是，为了实现兼容性，在以前的介质挂载点 `/cdrom` 和 `/rmdisk` 中提供了指向 `/media` 的符号链接。

例如，按以下方式挂载紧凑型闪存卡 (`/dev/dsk/c4d0p0:1`)：

```
$ ls /media/NIKON
```

例如，按以下方式挂载 USB 内存条 (`/dev/dsk/c3t0d0s0`)：

```
$ ls /media/U3
```

访问可移除介质数据的指导原则

大多数 DVD 都是按照可移植的 ISO 9660 标准格式化的。因此，卷管理可以挂载大多数 DVD。

要适应不同的格式，可将 DVD 分成若干个分片。这些分片在作用上与硬盘上的分区类似。9660 部分可移植。如果在挂载 DVD 时出现问题，尤其是安装 DVD 时，应确保其文件系统适合系统的体系结构。例如，可以检查 DVD 上的标签。

▼ 如何添加新的可移除介质驱动器

通常，大多数现代的总线类型都支持热插拔。这说明可以将磁盘插入空槽中，而且系统识别该磁盘。

有关热插拔设备的更多信息，请参见第 6 章，动态配置设备（任务）。

1. **成为管理员。**
2. **连接新的介质驱动器。**
   有关特定说明，请参见硬件手册。

3 **确认系统能识别新的介质驱动器。**

```
# rmformat
Looking for devices...
```

▼ 如何禁用或启用可移除介质服务

有时，可能需要在不使用可移除介质服务的情况下管理介质。本节介绍如何禁用和启用可移除介质服务。

禁用这些服务意味着必须使用 `mount` 命令手动挂载所有介质。

1 **确保未使用介质。**

如果无法确定是否已找到介质的所有用户，请使用 `fuser` 命令。请参见第 38 页中的“如何确定可移除介质是否仍在使用”。

2 **成为管理员。**

3 **选择以下操作之一：**

- 在此发行版中，您可以禁用部分或全部可移除介质功能：
  - 要阻止卷在用户会话之外挂载，请禁用 `rmvolmgr` 服务。例如：

    ```
    # svcadm disable rmvolmgr
    ```
  - 要阻止任何卷管理，请禁用 `dbus`、`hal` 和 `rmvolmgr` 服务。

    ```
    # svcadm disable rmvolmgr
    # svcadm disable dbus
    # svcadm disable hal
    ```

    禁用这些服务意味着必须使用 `mount` 命令手动挂载所有介质。
- 启用可移除介质服务。

  ```
  # svcadm enable rmvolmgr
  # svcadm enable dbus
  # svcadm enable hal
  ```

▼ 如何访问可移除介质上的信息

1 **插入介质。**

几秒后将挂载介质。

2 **列出介质的内容。**

```
% ls /media
```

示例 3–1 **访问可移除介质上的信息**

此示例说明如何访问 USB 内存条上的信息。

```
$ ls /media/usb-name
```

此示例说明如何访问 DVD 上的信息。

```
$ ls /media
sol_10_811_sparc  cdrom
```

▼ 如何从可移除介质复制信息

与任何其他文件系统一样，可以访问可移除介质上的文件和目录。仅有的重要限制与所有权和权限有关。

例如，如果您将文件从 DVD 复制到文件系统中，则您即是所有者。但是，您不会具有写入权限，因为 DVD 上的文件从没有写入权限。您必须自行更改权限。

1. **确保已挂载介质。**

   ```
   $ ls /media
   ```

   `ls` 命令会显示已挂载介质的内容。如果未显示内容，请参见第 37 页中的 “如何访问可移除介质上的信息”。

2. **（可选的）复制文件或目录。**

   例如，对于 DVD，可以执行以下操作：

   ```
   $ cp /media/sol_sparc/Solaris_11/file .
   $ ls -l
   -rwxr-xr-x   1 pmorph   gelfs  64065 Aug 2  2010 file
   ```

▼ 如何确定可移除介质是否仍在使用

1. **成为管理员。**

2. **确定正在访问介质的进程。**

   ```
   # fuser -u /media
   ```

   `-u` 显示介质的用户。

   有关更多信息，请参见 `fuser(1M)`。

3. **（可选的）中止正在访问介质的进程。**

   ```
   # fuser -u -k /media
   ```

   `-k` 中止正在访问介质的进程。

注意 – 仅在紧急情况下才能中止正在访问介质的进程。

4 **验证进程是否已中止。**

```
# pgrep process-ID
```

示例 3–2 确定介质是否仍在使用

以下示例说明用户 pmorph 正在访问 /media/sol_10_1008_sparc/Solaris_10/Tools 目录。

```
# fuser -u /media/sol_10_1008_sparc/Solaris_10/Tools
/media/sol_10_1008_sparc/Solaris_10/Tools:      723c(pmorph)     316c(pmorph)
```

▼ 如何弹出可移除介质

1 **确保未使用介质。**

请记住，如果 shell 或应用程序正在访问介质的任何文件或目录，则表明“正在使用”该介质。如果无法确定是否已找到 DVD 的所有用户（例如，隐藏在桌面工具之后的 shell 可能正在访问 DVD），请使用 fuser 命令。请参见第 38 页中的 “如何确定可移除介质是否仍在使用”。

2 **弹出介质。**

```
# eject media
```

例如，对于 DVD，可以执行以下操作：

```
# eject cdrom
```

例如，对于 USB 内存条，可以执行以下操作：

```
# eject rmdisk0
```

提示 – 可以使用 eject - l 命令查看可移除设备名称。

在远程系统上访问可移除介质（任务列表）

以下任务列表介绍了在远程系统上访问可移除介质所需的任务。

任务	说明	参考
1. 使本地介质可供远程系统使用。	将系统配置为共享其介质驱动器，以便使这些驱动器中的任何介质可供其他系统使用。	第 40 页中的“如何使本地介质可供其他系统使用”
2. 访问远程系统上的可移除介质。	在本地系统上访问远程介质。	第 37 页中的“如何访问可移除介质上的信息”

▼ 如何使本地介质可供其他系统使用

可以将系统配置为共享其介质驱动器，以便使这些驱动器中的任何介质可供其他系统使用。音乐 CD 除外。共享介质驱动器后，其他系统只需挂载这些驱动器即可访问其包含的介质。有关说明，请参见第 41 页中的“如何访问远程系统上的可移除介质”。

1. **成为管理员。**
2. **确认装入了介质。**
3. **共享介质。**

   例如：

   ```
   # share -F nfs -o ro /media/sol_10_811_sparc
   ```

4. **确定 NFS 服务器服务是否正在运行。**

   ```
   # svcs *nfs*
   ```

   如果 NFS 服务器服务正在运行，则 `svcs` 命令将返回以下输出：

   ```
   online         14:28:43 svc:/network/nfs/server:default
   ```

5. **确定 NFS 服务器的状态，然后选择以下操作之一：**

   - 如果 NFS 服务器服务正在运行，则转到步骤 7。
   - 如果 NFS 服务器服务**没有**运行，则转到下一步。

6. **启动 NFS 服务器服务。**

   ```
   # svcadm enable network/nfs/server
   ```

   验证 NFS 守护进程是否正在运行。

例如：

```
# svcs -p svc:/network/nfs/server:default
STATE          STIME    FMRI
online         Aug_30   svc:/network/nfs/server:default
               Aug_30        319 mountd
               Aug_30        323 nfsd
```

7 **验证介质是否确实可供其他系统使用。**

如果介质可用，则会显示其共享配置。

```
# share
-               /media/sol_10_811_sparc   sec=sys,ro   ""
```

示例 3–3 使本地 DVD 可供其他系统使用

以下示例说明如何使任何本地 DVD 可供网络中的其他系统使用。

```
# share -F nfs -o ro /media
# svcs *nfs*
# svcadm enable network/nfs/server
# svcs -p svc:/network/nfs/server:default
# share
-               /media/sol_10_811_sparc   ro   ""
```

▼ 如何访问远程系统上的可移除介质

通过手动将介质挂载到文件系统中，可以访问远程系统上的该介质。另外，远程系统还必须已按照第 40 页中的 “如何使本地介质可供其他系统使用”中的说明共享其介质。

1 **选择一个现有目录用作挂载点。或者，创建一个挂载点。**

```
$ mkdir /directory
```

其中 */directory* 是创建作为远程系统 DVD 挂载点的目录的名称。

2 **查找要挂载的介质的名称。**

```
$ showmount -e system-name
```

3 **以管理员身份挂载介质。**

```
# mount -F nfs -o ro system-name:/media/media-name  local-mount-point
```

system-name: 将挂载其介质的系统的名称。

media-name 要挂载的介质的名称。

local-mount-point 要挂载远程介质的本地目录。

4 **以管理员身份注销。**

5 验证是否已挂载介质。

```
$ ls /media
```

示例 3–4 在远程系统上访问 DVD 或 CD

以下示例说明如何从使用 AutoFS 的远程系统 starbug自动访问名为 sol_10_811_sparc的远程 DVD。

```
$ showmount -e starbug
export list for starbug:
/media/sol_10_811_sparc (everyone)
$ ls /net/starbug/media/
sol_10_811_sparc
```

◆ ◆ ◆ 第 4 章

写入 CD 和 DVD（任务）

本章提供有关使用 cdrw 命令写入和复制数据 CD 和 DVD 以及音频 CD 的逐步说明。

- 第 46 页中的“如何通过 RBAC 限制用户对可移除介质的访问”
- 第 46 页中的“如何标识 CD 或 DVD 写入者”
- 第 47 页中的“如何检查 CD 或 DVD 介质”
- 第 48 页中的“如何为数据 CD 或 DVD 创建 ISO 9660 文件系统”
- 第 48 页中的“如何创建多会话数据 CD”
- 第 50 页中的“如何创建音频 CD”
- 第 51 页中的“如何提取 CD 上的声道”
- 第 52 页中的“如何复制 CD”
- 第 53 页中的“如何删除 CD-RW 介质”

使用音频 CD 以及数据 CD 和 DVD

可以使用 cdrw 命令在 CD-R、CD-RW、DVD-RW 或 DVD+RW 介质设备上以 ISO 9660 格式（带有 Rock Ridge 或 Joliet 扩展）向 CD 和 DVD 中写入文件系统。

可以使用 cdrw 命令执行以下任务：

- 创建数据 CD 和 DVD。
- 创建音频 CD。
- 从音频 CD 提取音频数据。
- 复制 CD 和 DVD。
- 删除 CD-RW 介质。

以下发行版提供了 cdrw 命令：

- Oracle Solaris 10 发行版
- Oracle Solaris 11 发行版，media/cdrw 软件包

有关推荐的 CD-R 或 CD-RW 设备的信息，请转到：

http://www.oracle.com/us/sun/index.html

CD/DVD 介质常用术语

本节定义了与 CD/DVD 介质相关的常用术语。

术语	说明
CD-R	可以写入一次并且之后只能读取的 CD 读取介质。
CD-RW	可写入和删除的 CD 可重写介质。CD-RW 介质只能由 CD-RW 设备读取。
DVD-R	可以写入一次并且之后只能读取的数字视频光盘（可刻录）。这些设备的容量远远大于 CD-R 介质。
DVD+R	可以写入一次并且之后只能读取的数字视频光盘（可刻录）。DVD+R 设备的错误管理系统比 DVD-R 更完善，无论介质的质量如何，都能向介质进行更准确的烧录。
DVD-RW	存储容量与 DVD-R 相当的数字视频光盘（可重写）。在删除整个光盘后，该介质可重新录制。
DVD+RW	数字视频光盘（随机存取，可重写）具有等同于 DVD+R 的存储功能。该介质允许覆盖各个单独的块而无需删除整个光盘。
DVD-RAM	数字视频光盘（随机存取存储器，可重写）具有圆形（而非螺旋形）磁轨以及硬分段。
ISO 9660	ISO 是 Industry Standards Organization（行业标准组织）的首字母缩略词，该组织负责制定计算机存储格式的标准。 ISO 9660 文件系统是标准的 CD 或 DVD 文件系统，通过它可以在任何主要计算机平台上读取同一 CD 或 DVD。该标准于 1988 年发布，是由名为 High Sierra（以内华达州的 High Sierra 旅馆命名）的行业工作组编写的。几乎所有配有 CD 或 DVD 驱动器的计算机都可以从 ISO 9660 文件系统读取文件。
Joliet extensions（Joliet 扩展）	添加 Windows 文件系统信息。
Rock Ridge extensions（Rock Ridge 扩展）	添加 UNIX 文件系统信息。（Rock Ridge 以电影 Blazing Saddles 中的城镇命名。） **注 –** 这些扩展不互斥。可以同时指定 `mkisofs -R` 和 `-j` 选项以便与这两种系统兼容。（有关详细信息，请参见 mkisofs(1M)。）
MMC-compliant recorder（符合 MMC 标准的刻录机）	Multi Media Command（多媒体命令）的首字母缩略词，表示这些刻录机符合通用命令集。如果程序可以写入到一个符合 MMC 标准的刻录机，则应该可以写入到其他所有刻录机。
Red Book CDDA（红皮书 CDDA）	Compact Disc Digital Audio（光盘数字音频）的首字母缩略词，一种用于在光盘上存储数字音频的行业标准方法，也因术语“红皮书”格式而得名。正式的行业规范要求以 44.1 千赫 (kHz) 的抽样率在 16 位立体声中抽样一个或多个音频文件。

写入 CD 介质时常用的术语如下：

术语	说明
blanking（消隐）	从 CD-RW 介质中删除数据的过程。
`mkisofs`	用于在 CD 上创建 ISO 文件系统的命令。
session（会话）	具有引入和引出信息的完整磁轨。
track（磁轨）	完整的数据或音频单元。

写入 CD 和 DVD 数据及音频 CD

写入 CD 或 DVD 的过程不能中断，并且需要连续不断的数据流。请考虑使用 `cdrw -S` 选项模拟向介质进行写入，以验证系统是否可以充足的速率提供数据以便写入 CD 或 DVD。

写入错误可能是由以下问题之一导致的：

- 介质无法处理驱动器速度。例如，一些介质仅经过 2x 或 4x 速度的认证。
- 系统正在运行过多的超负荷进程，从而导致写入进程无法运行。
- 网络拥塞导致读取映像时产生延迟，并且映像位于远程系统上。
- 源驱动器的速度比目标驱动器慢。

如果出现上述任一问题，可以使用 `cdrw -p` 选项降低设备的写入速度。

例如，以下命令说明如何模拟以 4x 速度进行写入：

```
$ cdrw -iS -p 4 image.iso
```

注 – CD-R、CD-RW（非 MRW 格式）、DVD-R 和 DVD-RW 介质支持模拟模式 (`-S`)，但是 DVD-RAM、DVD+R、DVD+RW 和任何 MRW 格式的介质以及其他一些介质不支持模拟模式。如果不支持模拟模式，将出现以下消息：

```
Media does not support simulated writing
```

有关介质类型的更多信息，请参见第 44 页中的 “CD/DVD 介质常用术语”。

有关更多信息，请参见 `cdrw(1)`。

通过 RBAC 限制用户对可移除介质的访问

缺省情况下，所有用户都可以访问可移除介质。但是，可以通过基于角色的访问控制 (role-based access control, RBAC) 设置角色，限制用户对可移除介质的访问。通过为一组有限的用户指定角色，可以限制对可移除介质的访问。

有关使用角色的说明，请参见《Oracle Solaris 管理：安全服务》中的“基于角色的访问控制（概述）”。

▼ 如何通过 RBAC 限制用户对可移除介质的访问

1. **成为管理员。**

   有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。

2. **设置具有设备管理权限的角色。**

   有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的第 9 章 “使用基于角色的访问控制（任务）”。

```
# roleadd -m -d /export/home/muser -c "mediauser role" -A solaris.device.cdrw -P All muser
```

3. **将需要使用 `cdrw` 命令的用户添加到新创建的角色。**

```
# usermod -R muser joe
```

4. **注释 `/etc/security/policy.conf` 文件中的以下行：**

```
AUTHS_GRANTED=solaris.device.cdrw
```

   如果不执行此步骤，则所有用户仍可以访问 `cdrw` 命令，而不仅仅是作为设备管理角色的成员。

   修改此文件后，只有设备管理角色的成员才可以使用 `cdrw` 命令。将拒绝所有其他人的访问，并显示以下消息：

```
Authorization failed, Cannot access disks.
```

▼ 如何标识 CD 或 DVD 写入者

1. **在系统上标识 CD 或 DVD 写入者。**

   例如：

```
$ cdrw -l
Looking for CD devices...
    Node               |    Connected Device            |  Device type
-----------------------+--------------------------------+----------------
 cdrom0                | YAMAHA   CRW8824S         1.0d | CD Reader/Writer
```

2 **标识特定的 CD 或 DVD 写入者。**

例如：

```
$ cdrw -a filename.wav -d cdrom2
```

3 **确定介质是否为空或介质上是否存在目录。**

例如：

```
$ cdrw -M

Device : YAMAHA   CRW8824S
Firmware : Rev. 1.00 (26/04/00)
Media is blank
%
```

▼ 如何检查 CD 或 DVD 介质

无论可移除介质服务是否运行都可以使用 `cdrw` 命令。有关禁用或启用可移除介质服务的更多信息，请参见第 37 页中的 “如何禁用或启用可移除介质服务”。

1 **将 CD 或 DVD 插入驱动器。**

CD 或 DVD 可以是驱动器可读取的任何 CD 或 DVD。

2 **通过列出驱动器来检查是否已正确连接该驱动器。**

```
$ cdrw -l
 Looking for CD devices...
    Node                  Connected Device                Device type
----------------------+-------------------------------+-----------------
 cdrom1               | YAMAHA   CRW8824S         1.0d | CD Reader/Writer
```

3 **（可选的）如果列表中未显示该驱动器，请选择以下操作之一以便系统可识别该驱动器。**

- 添加驱动器，而不重新引导系统。

  ```
  # drvconfig
  # disks
  ```

  然后重新启动可移除介质服务。

创建数据 CD 或 DVD

首先使用 `mkisofs` 命令准备数据，将文件和文件信息转换为可在 CD 或 DVD 上使用的 High Sierra 格式。

▼ 如何为数据 CD 或 DVD 创建 ISO 9660 文件系统

1 **将空白 CD 或 DVD 插入驱动器。**

2 **在新 CD 或 DVD 上创建 ISO 9660 文件系统。**

```
$ mkisofs -r /pathname > cd-file-system
```

`-r`	创建 Rock Ridge 信息并将文件所有权重置为零。
/pathname	标识用于创建 ISO 9660 文件系统的路径名。
> cd-file-system	标识要放置在 CD 或 DVD 上的文件系统的名称。

3 **将文件系统复制到 CD 或 DVD 上。**

```
$ cdrw -i cd-file-system
```

`-i` *cd-file-system* 指定用于创建数据 CD 或 DVD 的映像文件。

示例 4–1 为数据 CD 或 DVD 创建 ISO 9660 文件系统

以下示例说明如何为数据 CD 或 DVD 创建 ISO 9660 文件系统。

```
$ mkisofs -r /home/dubs/dir > dubs_cd
Total extents actually written = 56
Total translation table size: 0
Total rockridge attributes bytes: 329
Total directory bytes: 0
Path table size(bytes): 10
Max brk space used 8000
56 extents written (0 Mb)
```

然后，将文件系统复制到 CD 上。

```
$ cdrw -i dubs_cd
Initializing device...done.
Writing track 1...done.
Finalizing (Can take several minutes)...done.
```

▼ 如何创建多会话数据 CD

此过程介绍如何在 CD 上放置多个会话。此过程包括将 `infoA` 和 `infoB` 目录复制到 CD 上的示例。

1 **为第一个 CD 会话创建文件系统。**

```
$ mkisofs -o infoA -r -V my_infoA /data/infoA
Total translation table size: 0
Total rockridge attributes bytes: 24507
Total directory bytes: 34816
```

```
Path table size(bytes): 98
Max brk space used 2e000
8929 extents written (17 Mb)
```

`-o infoA` 标识 ISO 文件系统的名称。

`-r` 创建 Rock Ridge 信息并将文件所有权重置为零。

`-V my_infoA` 标识将由可移除介质服务用作挂载点的卷标。

`/data/infoA` 标识要创建的 ISO 映像目录。

2 将第一个会话的 ISO 文件系统复制到 CD 上。

```
$ cdrw -iO infoA
Initializing device...done.
Writing track 1...done.
done.
Finalizing (Can take several minutes)...done.
```

`-i` *infoA* 标识要写入 CD 的映像文件的名称。

`-O` 使 CD 保持打开状态以便进行写入。

3 弹出 CD 后将其重新插入。

4 标识要在下一个写入会话中包括的 CD 介质的路径名。

```
$ eject -n
.
.
.
cdrom0 -> /vol/dev/rdsk/c2t4d0/my_infoA
```

请注意 `/vol/dev/...` 路径名。

5 标识 CD 上下一个可写入地址以便写入下一个会话。

```
% cdrw -M /cdrom
Device : YAMAHA   CRW8424S
Firmware : Rev. 1.0d (06/10/99)

Track No. |Type    |Start address
----------+--------+-------------
 1        |Audio   |0
 2        |Audio   |33057
 3        |Data    |60887
 4        |Data    |68087
 5        |Data    |75287
Leadout   |Data    |84218

Last session start address: 75287
Next writable address: 91118
```

记下 `Next writable address` 输出中的地址，以便在写入下一个会话时可以提供此地址。

6 为下一个 CD 会话创建下一个 ISO 文件系统，然后将其写入 CD。

```
$ mkisofs -o infoB -r -C 0,91118 -M /vol/dev/rdsk/c2t4d0/my_infoA
/data/infoB
Total translation table size: 0
Total rockridge attributes bytes: 16602
Total directory bytes: 22528
Path table size(bytes): 86
Max brk space used 20000
97196 extents written (189 Mb)
```

`-o` *infoB*	标识 ISO 文件系统的名称。
`-r`	创建 Rock Ridge 信息并将文件所有权重置为零。
`-C` *0,91118*	标识第一个会话的起始地址和下一个可写入地址。
`-M /vol/dev/rdsk/c2t4d0/my_infoA`	指定要合并的现有 ISO 映像的路径。
`/data/infoB`	标识要创建的 ISO 映像目录。

创建音频 CD

可以使用 `cdrw` 命令从单独的声道或者从 `.au` 和 `.wav` 文件创建音频 CD。

下表介绍了支持的音频格式：

格式	说明
`sun`	数据为红皮书 CDDA 格式的 Oracle .au 文件
`wav`	数据为红皮书 CDDA 格式的 RIFF (.wav) 文件
`cda`	包含原始 CD 音频数据的 `.cda` 文件，该音频数据是按小尾数法字节顺序以 44.1 kHz 抽样率进行抽样的 16 位 PCM 立体声
`aur`	包含原始 CD 数据的 `.aur` 文件，该数据以大尾数法字节顺序排序

如果未指定任何音频格式，则 `cdrw` 命令会尝试根据文件扩展名确定音频文件的格式。忽略扩展名中字符的大小写。

▼ 如何创建音频 CD

此过程介绍如何将音频文件复制到 CD 上。

1 将空白 CD 插入 CD-RW 驱动器。

2. **转到包含音频文件的目录。**

   ```
   $ cd /myaudiodir
   ```

3. **将音频文件复制到 CD 上。**

   ```
   $ cdrw -a track1.wav track2.wav track3.wav
   ```

   -a 选项用于创建音频 CD。

示例 4–2 创建音频 CD

以下示例说明如何创建音频 CD。

```
$ cdrw -a bark.wav chirp.au meow.wav
Initializing device...done.
Writing track 1...done.
done.
Writing track 2...done.
Writing track 3...done.
done.
Finalizing (Can take several minutes)...done.
```

以下示例说明如何创建多会话音频 CD。写入第一个会话后，将弹出 CD。在下一个写入会话之前，需要重新插入 CD。

```
$ cdrw -aO groucho.wav chico.au harpo.wav
Initializing device...done.
Writing track 1...done.
done.
Writing track 2...done.
Writing track 3...done.
done.
Finalizing (Can take several minutes)...done.
<Re-insert CD>
$ cdrw -a zeppo.au
Initializing device...done.
Writing track 1...done.
done.
Finalizing (Can take several minutes)...done.
```

▼ 如何提取 CD 上的声道

使用以下过程可从 CD 提取声道并将该声道复制到新 CD 上。

如果不使用 cdrw -T 选项指定音频文件类型，则 cdrw 命令会使用文件扩展名来确定音频文件类型。例如，cdrw 命令检测到此文件是 .wav 文件。

```
$ cdrw -x 1 testme.wav
```

1. **将音频 CD 插入 CD-RW 驱动器。**

2 提取声道。

```
$ cdrw -x -T audio-type 1 audio-file
```

-x　从音频 CD 提取音频数据。

T *audio-type*　标识要提取的音频文件的类型。支持的音频类型包括 sun、wav、cda 或 aur。

audio-file　标识要提取的声道。

3 将声道复制到新 CD。

```
$ cdrw -a audio-file
```

示例 4–3 提取和创建音频 CD

以下示例说明如何从音频 CD 提取第一个声道并将文件命名为 song1.wav。

```
$ cdrw -x -T wav 1 song1.wav
Extracting audio from track 1...done.
```

本示例说明如何将声道复制到音频 CD。

```
$ cdrw -a song1.wav
Initializing device...done.
Writing track 1...done.
Finalizing (Can take several minutes)...done.
```

▼ 如何复制 CD

此过程说明如何将音频 CD 中的所有声道提取到一个目录中，然后将其全部复制到空白 CD 上。

注 – 缺省情况下，cdrw 命令会将 CD 复制到 /tmp 目录中。复制时可能需要多达 700 MB 的空闲空间。如果 /tmp 目录中用于复制 CD 的空间不足，请使用 -m 选项指定替代目录。

---

1 将音频 CD 插入 CD-RW 驱动器。

2 为音频文件创建目录。

```
$ mkdir /music_dir
```

3 从音频 CD 提取声道。

```
$ cdrw -c -m music_dir
```

对于每个声道，将显示 Extracting audio ... 消息。

提取所有声道后，将弹出 CD。

4 **插入空白 CD，然后按回车键。**

提取声道后，将弹出音频 CD。系统将提示插入空白 CD。

示例 4–4 复制 CD

本示例说明如何将一张 CD 复制到另一张 CD。必须具有两个 CD-RW 设备才能执行此任务。

```
$ cdrw -c -s cdrom0 -d cdrom1
```

▼ 如何删除 CD-RW 介质

必须删除现有的 CD-RW 数据后，才能重写 CD。

- **通过选择以下操作之一，删除整个介质或仅删除 CD 上的最后一个会话：**
  - 仅删除最后一个会话。

    ```
    $ cdrw -d cdrom0 -b session
    ```

    使用 `-b session` 选项仅删除最后一个会话比使用 `-b all` 选项删除整个介质更快。即使仅在一个会话中使用 `cdrw` 命令创建数据或音频 CD，仍可以使用 `-b session` 选项。
  - 删除整个介质。

    ```
    $ cdrw -d cdrom0 -b all
    ```

◆ ◆ ◆ 第 5 章

管理设备（概述/任务）

本章提供 Oracle Solaris 发行版中管理外围设备（例如磁盘、DVD 驱动器和磁带设备）的概述信息和逐步说明。

以下是本章中概述信息的列表。

- 第 55 页中的“在设备管理方面的新增功能？”
- 第 59 页中的“有关其他设备管理任务的参考信息”
- 第 59 页中的“关于设备驱动程序”
- 第 60 页中的“设备的自动配置”
- 第 62 页中的“显示设备配置信息”
- 第 66 页中的“解决有故障的设备”
- 第 69 页中的“访问设备”

以下是本章中逐步说明的列表。

- 第 60 页中的“如何定制驱动程序配置”
- 第 63 页中的“如何显示系统配置信息”
- 第 66 页中的“如何解决有故障的设备”
- 第 69 页中的“如何添加设备驱动程序”
- 第 68 页中的“如何添加外围设备”

Oracle Solaris 发行版中的设备管理通常涉及在系统中添加和删除外围设备，其中还可能涉及添加支持设备的第三方设备驱动程序，以及显示系统配置信息。

在设备管理方面的新增功能？

本节提供有关 Oracle Solaris 发行版在设备管理方面新增功能的信息。

- 第 56 页中的“定制驱动程序配置”
- 第 56 页中的“Solaris PCI 资源管理器”
- 第 56 页中的“新的 InfiniBand 管理功能”
- 第 57 页中的“新的 InfiniBand 诊断工具和命令”

- 第 57 页中的 “新的 Ethernet over InfiniBand 设备”
- 第 57 页中的 “新的热插拔功能”
- 第 57 页中的 “设备命名增强功能”
- 第 58 页中的 “支持 PCI Express (PCIe)”

定制驱动程序配置

Oracle Solaris 11： 在本 Solaris 发行版中，您可以在 `/etc/driver/drv` 目录中提供补充的驱动程序配置文件 *driver.conf*。在以前的 Solaris 发行版中，您必须直接修改供应商的 *driver.conf* 文件。

在本发行版中，系统自动将驱动程序供应商的 `/kernel` 或 `/platform` *driver.conf* 文件与定制的 `/etc/driver/drv` *driver.conf* 文件合并，以便于该驱动程序显示这两组属性值。

您可以使用 `prtconf -u` 命令显示初始供应商和定制属性值。将供应商的 *driver.conf* 文件和本地定制的 *driver.conf* 文件分开会使得系统使用新的供应商的 *driver.conf* 文件进行升级，而不覆盖管理定制。

如果以前的 Solaris 发行版或系统的 `/kernel/drv` 目录中的驱动程序配置文件中存在现有的修改，您应将其复制到 `/etc/driver/drv` 目录中。

有关创建 `/etc/driver/drv/ driver.conf` 文件的信息，请参见第 60 页中的 “如何定制驱动程序配置”、`driver.conf(4)` 和 `driver(4)`。

Solaris PCI 资源管理器

Oracle Solaris 11： 在本发行版中，提供 PCI 资源管理器 (PCI resource manager, PCIRM) 来重新平衡 PCI 资源，以确保在系统引导时，指定足够的资源来支持 PCI Express (PCIe) 设备。例如，单根 I/O 虚拟 (Single Root I/O Virtualization, SR-IOV) 设备要求更多的 PCI 内存资源用于其虚拟功能。现在大多数的固件设计均先于 SR-IOV 标准，所以不会认可储备虚拟设备资源的要求。资源重新平衡进程在引导时自动发生，并且不要求管理。

新的 InfiniBand 管理功能

Oracle Solaris 11： IPoIB 的管理已同早期发行版不同。在本发行版中，您可以通过使用 `dladm` 子命令 `create-part`、`delete-part`、`view-part` 和 `show-ib` 创建、删除并查看 IPoIB 数据链路信息。这些新的 `dladm` 子命令提供更简单的方法来管理 IPoIB 数据链路，还帮助查看并对一些 IB 问题进行故障排除。例如，`show-ib` 子命令显示系统中有多少 HCA 以及相应的 HCA GUID。在新的管理模型中，有两种类型的 IP over IB 数据链路。

- 物理数据链路－类似于常规网络接口卡 (Network Interface Card, NIC) 数据链路

- IB 分区数据链路－类似于虚拟 NIC (Virtual NIC, VNIC)

缺省情况下，在新安装的系统中自动创建物理链接。然后，您可以使用 `dladm` 命令在 IB 物理链接上创建 IB 分区链接。和 NIC 数据链路不同，IB 物理链接无法进行配置，例如检测或指定供使用链接的 IP 地址。

有关使用 `dladm` 子命令管理 IB 组件的更多信息，请参见第 164 页中的 “管理 IPoIB 设备 (dladm)”。

新的 InfiniBand 诊断工具和命令

Oracle Solaris 11： `system/io/infiniband/open-fabrics` 软件包提供了 Open Fabrics Alliance (OFA) Open Fabrics Enterprise Distribution (OFED) 版本 1.5.3 中的部分命令和实用程序。该工具集合提供了以下功能：列出和查询 IB 设备、对 IB 结构进行诊断和故障排除，以及使用一组 IB 用户动词微基准来测量 IB 性能。有关更多信息，请参见第 166 页中的 “监视并排除 IB 设备故障”。

新的 Ethernet over InfiniBand 设备

Oracle Solaris 11： Ethernet over InfiniBand (`eoib`) 驱动程序是一个多线程、可下载、可复制的基于 GLD 的流驱动程序，支持数据链路提供商接口 (Data Link Provider Interface, DLPI) 和连接到 Oracle 网络 QDR InfiniBand 网关交换机的系统中的整体 IB 端口。

新的热插拔功能

Oracle Solaris 11：在本发行版中，`hotplug` 命令可用于管理 PCI EXPRESS (PCIe) 和 PCI SHPC（标准热插拔控制器）设备上的可热插拔连接。其他总线类型上不支持该功能，例如 USB 和 SCSI。

您仍然可以使用 `cfgadm` 命令管理可热插拔 USB 和 SCSI 设备，与以前的 Solaris 发行版相同。在本发行版中使用 hotplug 功能的益处是，除了启用和禁用操作外，`hotplug` 命令还为您受支持的 PCI 设备提供了脱机和联机功能。

有关更多信息，请参见第 79 页中的 “使用 (hotplug) 命令进行 PCIe 热插拔”。

设备命名增强功能

Oracle Solaris 11： `/dev` 名称空间根据需要支持多个文件系统实例。系统引导时，自动创建 `/dev` 文件系统的全局实例。在需要时创建并挂载后续的 `/dev` 实例，例如将设备添加到非全局区域时。非全局区域关闭时，可用的 `/dev` 实例会取消挂载并且不可用。

此外，采用以下方法改进了设备配置：

- **删除重新配置引导**－在以前的 Solaris 发行版中，将设备连接到电源关闭的系统时，需要重新配置引导。

  在本发行版中，将设备连接到电源关闭的系统时，不需要执行重新配置引导。自动识别新连接的设备，并且当系统重新引导时，创建适当的设备链接。

  有关更多信息，请参见dev(7FS)。
- **简化区域设备支持**－如上所述，通过为非全局区域提供 `/dev` 目录的特定实例，增强了对 Oracle Solaris 区域的设备支持。此外，区域不再依赖 `devfsadm` 守护进程来重新配置区域中的设备。
- **改进 Pseudo 设备创建**－在本发行版中，当非全局区域需要时，`/dev/pts` 目录的内容按照全局 `/dev` 名称空间以及 `/dev` 实例的请求创建。此外，`pty` 链接仅在全局区域或从中分配它们的非全局区域中可见。

  有关更多信息，请参见 `grantpt(3C)`。

有关设备配置的更多信息，请参见第 59 页中的 “管理 Oracle Solaris OS 中的设备”。

支持 PCI Express (PCIe)

Oracle Solaris 11：本 Oracle Solaris 发行版可提供对 PCI Express (PCIe) 互连的支持，此功能目的在于将外围设备连接至 SPARC 和 x86 系统中的桌面、企业、移动、通信和嵌入式应用程序。

PCIe 互连是行业标准的高性能串行 I/O 总线。有关 PCIe 技术的详细信息，请访问以下站点：

`http://www.pcisig.com/home`

本 Oracle Solaris 发行版中的 PCIe 软件提供了以下功能：

- 支持扩展的 PCIe 配置空间
- 支持 PCIe 基准线错误处理和 MSI 中断
- PCIe 设备的经修改的 IEEE-1275 属性
- 通过增强 `cfgadm` 命令的 `cfgadm_pci` 组件以支持 PCIe 热插拔（本机和基于 ACPI）
- 基于 ATTN 按钮使用情况的 PCIe 外围设备自动配置

热插拔 PCIe 外围设备与 PCI 外围设备的管理模型相同，也是使用 `cfgadm` 命令。

请检查硬件平台指南，确保系统提供 PCIe 和 PCIe 热插拔支持。此外，请认真查看有关以物理方式在系统中插入或移除适配器的说明以及设备自动配置的语义（如果适用）。

有关针对 PCIe 外围设备使用 `cfgadm` 命令的信息，请参见第 90 页中的 “使用 `cfgadm` 命令执行 PCI 或 PCIe 热插拔（任务列表）”。

有关其他设备管理任务的参考信息

下表介绍了从何处获取有关对设备进行热插拔以及添加串行设备（如打印机和调制解调器）的逐步说明。

表 5–1 有关添加设备的参考信息

设备管理任务	更多信息
添加不可热插拔的磁盘。	第 12 章，SPARC：设置磁盘（任务）或第 13 章，x86：设置磁盘（任务）
热插拔 SCSI 或 PCI 设备。	第 81 页中的 “使用 `cfgadm` 命令进行 SCSI 热插拔”或第 91 页中的 “使用 `cfgadm` 命令执行 PCI 或 PCIe 热插拔”
热插拔 USB 设备。	第 122 页中的 “使用 USB 海量存储设备（任务列表）”
添加调制解调器。	《Oracle Solaris 管理：常见任务》中的第 16 章 “管理系统控制台、终端设备和电源服务（任务）”
添加打印机。	《Oracle Solaris 管理：常见任务》中的第 15 章 “使用 CUPS 设置和管理打印机（任务）”
保护设备。	《Oracle Solaris 管理：安全服务》中的第 5 章 “控制对设备的访问（任务）”

管理 Oracle Solaris OS 中的设备

以下各节提供关于管理 Oracle Solaris OS 中的设备功能的概述信息。有关访问设备的信息，请参见第 69 页中的 “访问设备”。

确定设备支持

您可以使用设备检测工具确定本 Oracle Solaris 发行版中是否支持 x86 硬件。有关更多信息，请访问以下站点：

`http://www.oracle.com/webfolder/technetwork/hcl/hcts/device_detect.jsp`

关于设备驱动程序

一台计算机通常会使用大量外围设备和海量存储设备。例如，您的系统可能装有磁盘驱动器、键盘和鼠标，以及某种磁性备份介质。其他常用的设备包括以下各项：

- DVD 驱动器
- 打印机和绘图仪
- 光笔

- 触控式屏幕
- 数字化仪
- 图形输入板和笔尖对

Oracle Solaris 软件并不直接与所有这些设备进行通信。每种类型的设备都要求不同的数据格式、协议和传输率。

设备驱动程序是一种低级程序，可用于使操作系统与某个特定的硬件进行通信。驱动程序可用作操作系统对该硬件的“解释程序”。

▼ 如何定制驱动程序配置

在 Oracle Solaris 11 发行版中，驱动程序定制在 `/etc/driver/drv` 目录中，而非 `/kernel` 目录，与以前的发行版一样。此改进意味着在系统升级时，您的驱动程序定制不会被覆盖。在升级期间保留 `/etc/driver/drv` 目录中的文件。

对驱动程序配置进行定制通常意味着添加或修改每设备参数或者添加或修改影响所有设备的全局属性。

1. **成为管理员。**
2. **将初始的供应商所提供的 *driver.conf* 文件复制到 `/etc/driver/drv` 目录中。例如：**

   ```
   # cp /kernel/drv/sd.conf /etc/driver/drv/sd.conf
   ```

3. **修改参数条目并保存文件。**

   例如，`sd.conf` 包括目标 0，lun 0 处的 `sd` 设备的以下条目：

   ```
   name="sd" class="scsi" target=0 lun=0;
   ```

   若要为该设备添加 `retries` 参数，请按照以下所示修改现有的条目：

   ```
   name="sd" class="scsi" target=0 lun=0 retries=4;
   ```

4. **显示定制的属性值。例如：**

   ```
   # prtconf -u
   sd, instance #1
              Admin properties:
                  name='retries' type=int items=1
                      value=00000004
   ```

设备的自动配置

内核包含一个小的通用核心，其中带有一个平台特定的组件和一组模块。在本 Oracle Solaris 发行版中，内核是自动配置的。

内核模块是用于在系统上执行特定任务的一个硬件组件或软件组件。**可装入**内核模块的一个示例是访问设备时装入的设备驱动程序。

与平台无关的内核为 /kernel/genunix。平台特定的组件为 /platform/‘uname -m‘/kernel/unix。

内核模块在下表中介绍。

表 5–2 Solaris 内核模块的说明

位置	目录内容
/platform/‘uname -m‘/kernel	平台特定的内核组件
/kernel	引导系统所需的通用于所有平台的内核组件
/usr/kernel	特定指令集内通用于所有平台的内核组件

系统可确定在引导时与其连接的设备。然后，内核会对其自身进行动态配置，将所需的模块装入内存中。此时，在访问磁盘和磁带等设备时就会装入设备驱动程序。此过程称为**自动配置**，因为所有的内核模块在需要时都会自动装入。

通过修改 /etc/system 文件，可以定制装入内核模块的方式。有关修改此文件的说明，请参见system(4)。

自动配置的功能和优点

自动配置的优点如下：

- 由于模块会在需要时装入，因此可更有效地使用主内存。
- 向系统中添加新设备时，无需重新配置内核。
- 可以装入并测试驱动程序，而不必重构内核和重新引导系统。

向系统中添加新设备（和驱动程序）时会使用自动配置功能。在以前的发行版中，将设备添加到关闭的系统中时，有必要执行重新配置引导。设备配置增强功能使得在将设备添加到关闭的系统中时，无需重新配置引导。

如果系统组件支持热插拔，当系统仍在运行时，可以在 Oracle Solaris OS 中添加、移除或更换设备。有关热插拔设备的信息，请参见第 6 章，动态配置设备（任务）。

不受支持的设备所需的内容

Oracle Solaris 发行版中包括支持各种标准设备所需的设备驱动程序。可以在 /kernel/drv 和 /platform/‘uname -m‘/kernel/drv 目录中找到这些驱动程序。

但是，如果购买了不受支持的设备，制造商应该提供正确安装、维护和管理设备所需的软件。

该软件至少包括设备驱动程序及其关联的配置 (.conf) 文件。.conf 文件驻留在 drv 目录中。该软件可能还包括定制维护和管理实用程序，因为该设备可能与 Oracle Solaris 实用程序不兼容。

有关不受支持的设备所需内容的更多信息，请与设备制造商联系。

显示设备配置信息

有三个命令可用于显示系统和设备的配置信息。

命令	说明	手册页
prtconf	显示系统配置信息，包括系统设备分层结构描述的内存总量和设备配置。通过此命令显示的输出取决于系统的类型。	prtconf(1M)
sysdef	显示设备配置信息，包括系统硬件、伪设备、可装入模块以及选定的内核参数。	sysdef(1M)
dmesg	显示系统诊断消息以及从上次系统重新引导以来与系统连接的设备的列表。	dmesg(1M)

有关用于标识系统上设备的设备名称的信息，请参见第 70 页中的 “设备命名约定”。

driver not attached 消息

prtconf 和 sysdef 命令可能会显示以下与驱动程序有关的消息：

device, instance #*number* (driver not attached)

此消息并不总是表示驱动程序不可用于此设备。此消息表示**当前**没有向设备实例附加任何驱动程序，因为此节点中不存在任何设备或者设备未在使用中。驱动程序是在访问设备时自动装入的。如果设备未在使用中，则会卸载这些驱动程序。

使用中的设备错误检查

指定的设备在使用中时，用以下实用程序进行检测：

- dumpadm
- format
- mkfs 和 newfs
- swap

这些增强功能说明以上实用程序可能会检测以下某些使用情况：

- 设备是 ZFS 存储池的一部分
- 设备是转储设备或交换设备
- /etc/vfstab 文件中存在设备的已挂载文件系统或某个项

例如，如果尝试使用 format 实用程序访问某个活动的设备，您将看到类似于以下内容的消息：

```
# format
Searching for disks...done
AVAILABLE DISK SELECTIONS:
       0. c1t0d0 <FUJITSU-MAY2073RCSUN72G-0401 cyl 8921 alt 2 hd 255 sec 63>
          /pci@0,0/pci1022,7450@2/pci1000,3060@3/sd@0,0
          /dev/chassis/SYS/HD0/disk
       1. c1t1d0 <FUJITSU-MAY2073RCSUN72G-0401-68.37GB>
          /pci@0,0/pci1022,7450@2/pci1000,3060@3/sd@1,0
          /dev/chassis/SYS/HD1/disk
Specify disk (enter its number): 0
selecting c1t0d0
[disk formatted]
/dev/dsk/c1t0d0s0 is part of active ZFS pool rpool. Please see zpool(1M).


FORMAT MENU:
.
.
.
```

▼ 如何显示系统配置信息

使用 `prtconf` 和 `sysdef` 命令的输出可确定与系统连接的磁盘、磁带和 DVD 设备。这些命令的输出会在设备实例旁边显示 `driver not attached` 消息。由于这些设备始终受某个系统进程监视，因此 `driver not attached` 消息通常明确表示该设备实例中不存在任何设备。

使用 `sysdef` 命令可显示系统配置信息，其中包括伪设备、可装入模块以及选定的内核参数。

- **显示系统和设备的配置信息。**
  - 显示与系统连接的所有设备。

    例如，SPARC 系统上的以下 `prtconf -v` 输出确定了与系统连接的磁盘设备。详细的磁盘信息在 `Device Minor Nodes` 一节介绍。

    ```
    $ /usr/sbin/prtconf -v | more
    Memory size: 32640 Megabytes
    System Peripherals (Software Nodes):

    SUNW,SPARC-Enterprise-T5220
    .
    .
    .
    location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__2/disk
    Device Minor Nodes:
         dev=(27,40)
            dev_path=/pci@0/pci@0/pci@8/pci@0/pci@a/LSILogic,sas@0/sd@2,0:a
                spectype=blk type=minor
                dev_link=/dev/dsk/c4t2d0s0
            dev_path=/pci@0/pci@0/pci@8/pci@0/pci@a/LSILogic,sas@0/sd@2,0:a,raw
                spectype=chr type=minor
                dev_link=/dev/rdsk/c4t2d0s0
                Device Minor Layered Under:
                mod=zfs accesstype=blk
    ```

```
                dev_path=/pseudo/zfs@0
            Minor properties:
                name='Nblocks' type=int64 items=1 dev=(27,40)
                      value=0000000074702c8f
                name='Size' type=int64 items=1 dev=(27,40)
                value=000000e8e0591e00
.
.
.
```

- 显示有关与系统连接的某个特定设备的信息。

  例如，SPARC 系统上的以下 `prtconf` 输出显示了 `/dev/dsk/c4t2d0s0` 的 `sd` 实例编号。

  ```
  # prtconf -v /dev/dsk/c4t2d0s0 | grep instance
  sd, instance #5
  ```

- 仅显示与系统连接的设备。

  ```
  # prtconf | grep -v not
  ```

- 显示设备使用信息。

  例如，以下 `fuser` 命令显示了正在访问 `/dev/console` 设备的进程。

  ```
  # fuser -d /dev/console
  /dev/console:     5742o    2269o   20322o     858o
  ```

示例 5–1 显示系统配置信息

基于 x86 的系统会显示以下 `prtconf` 输出。

```
# prtconf
System Configuration:  Oracle Corporation  i86pc
Memory size: 8192 Megabytes
System Peripherals (Software Nodes):

i86pc
    scsi_vhci, instance #0
    pci, instance #0
        pci108e,4843, instance #0
        pci8086,25e2, instance #0
            pci8086,3500, instance #7
                pci8086,3510, instance #9
                pci8086,3518, instance #10
                    pci108e,4843, instance #0
                    pci108e,4843, instance #1
            pci8086,350c, instance #8
        pci8086,25e3 (driver not attached)
        pci8086,25f8, instance #2
            pci108e,286, instance #0
                disk, instance #0
                disk, instance #2
                disk, instance #3
                disk, instance #1
        pci8086,25e5 (driver not attached)
        pci8086,25f9 (driver not attached)
        pci8086,25e7 (driver not attached)
```

```
        pci108e,4843, instance #0 (driver not attached)
        pci108e,4843, instance #1
        pci108e,4843, instance #2 (driver not attached)
        pci108e,4843 (driver not attached)
        pci108e,4843 (driver not attached)
        pci108e,4843 (driver not attached)
        pci108e,4843 (driver not attached)
        pci8086,2690, instance #6
            pci108e,125e, instance #2
            pci108e,125e, instance #3
        pci108e,4843, instance #0
        pci108e,4843, instance #1
            device, instance #0
                keyboard, instance #0
                mouse, instance #1
        pci108e,4843, instance #2
        pci108e,4843, instance #3
        pci108e,4843, instance #0
            storage, instance #0
                disk, instance #4
.
.
.
```

基于 x86 的系统会显示以下 sysdef 输出。

```
# sysdef
* Hostid
*
  29f10b4d
*
* i86pc Configuration
*
*
* Devices
*
+boot (driver not attached)
memory (driver not attached)
aliases (driver not attached)
chosen (driver not attached)
i86pc-memory (driver not attached)
i86pc-mmu (driver not attached)
openprom (driver not attached)
options, instance #0
packages (driver not attached)
delayed-writes (driver not attached)
itu-props (driver not attached)
isa, instance #0
    motherboard (driver not attached)
    pnpADP,1542, instance #0
    asy, instance #0
    asy, instance #1
    lp, instance #0 (driver not attached)
    fdc, instance #0
        fd, instance #0
        fd, instance #1 (driver not attached)
    kd (driver not attached)
    kdmouse (driver not attached)
```

```
.
.
.
```

解决有故障的设备

设备弃用机制通过故障管理框架 (fault management framework, FMA) 将设备隔离为**有故障**。通过该功能，可以安全且自动地禁用故障设备，从而避免数据丢失、数据损坏或紧急情况和系统停机。弃用过程安全地执行，考虑弃用设备后的系统稳定性。

永远不弃用关键设备。如果需要手动更换弃用的设备，除了执行手动更换步骤以外，还应在更换设备后使用 `fmadm repair` 命令，以便系统知道设备已更换。

有关更多信息，请参见 fmadm(1M)。

有关设备弃用的一般消息显示在控制台上，并写入 `/var/adm/messages` 文件中，以使您了解弃用的设备。例如：

```
Aug 9 18:14 starbug genunix: [ID 751201 kern.notice] NOTICE: One or more I/O devices have been retired
```

可以使用 `prtconf` 命令来确定特定的弃用设备。例如：

```
# prtconf
.
.
.
pci, instance #2
        scsi, instance #0
            disk (driver not attached)
            tape (driver not attached)
            sd, instance #3
            sd, instance #0 (retired)
        scsi, instance #1 (retired)
            disk (retired)
            tape (retired)
    pci, instance #3
        network, instance #2 (driver not attached)
        network, instance #3 (driver not attached)
    os-io (driver not attached)
    iscsi, instance #0
    pseudo, instance #0
.
.
.
```

▼ 如何解决有故障的设备

使用下面的步骤解决有故障的设备或已弃用的设备。

还可以通过使用 `zpool status` 或 `fmdump` 命令查看 ZFS 设备问题或故障信息。有关 ZFS 设备问题或故障信息，请参见《Oracle Solaris 管理：ZFS 文件系统》中的第 11 章 “Oracle Solaris ZFS 故障排除和池恢复”。

1 **使用 `fmadm faulty` 命令确定发生故障的设备。**

```
# fmadm faulty
--------------- ------------------------------------  -------------- ---------
TIME            EVENT-ID                              MSG-ID         SEVERITY
--------------- ------------------------------------  -------------- ---------
May 06 03:38:06 0376b4b6-fce7-c0f0-ffd9-a0a685376284  ZFS-8000-D3    Major

Host        : neo-2
Platform    : Sun-Fire-X4140    Chassis_id  : 0904QAD02C
Product_sn  :

Fault class : fault.fs.zfs.device
Affects     : zfs://pool=tank/vdev=c26c72a8ffcff889
                  faulted and taken out of service
Problem in  : zfs://pool=tank/vdev=c26c72a8ffcff889
                  faulted and taken out of service

Description : A ZFS device failed.  Refer to http://sun.com/msg/ZFS-8000-D3 for
              more information.

Response    : No automated response will occur.

Impact      : Fault tolerance of the pool may be compromised.

Action      : Run 'zpool status -x' and replace the bad device.
```

2 **更换有故障或弃用的设备或清除设备错误。**

在 ZFS 存储池中更换有故障的设备之后，清除设备错误。例如：

```
# zpool clear rpool c4t0d0s0
```

如果设备发生间歇错误，但没有更换该设备，则可以尝试按照以上所示清除错误。

3 **清除 FMA 故障。**

```
# fmadm repair 0376b4b6-fce7-c0f0-ffd9-a0a685376284
```

4 **确认故障已清除。**

```
# fmadm faulty
```

向系统添加外围设备

添加不可热插拔的新外围设备通常涉及以下操作：

- 关闭系统
- 将设备连接至系统
- 重新引导系统

使用第 68 页中的 “如何添加外围设备”向系统添加以下不可热插拔的设备：

- DVD 驱动器
- 辅助磁盘驱动器

- 磁带机
- SBUS 卡

在某些情况下，为支持新设备，可能必须添加第三方设备驱动程序。

有关热插拔设备的信息，请参见第 6 章，动态配置设备（任务）。

▼ 如何添加外围设备

1. **成为管理员。**
   有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。

2. **可选如果需要添加设备驱动程序以支持设备，请完成过程第 69 页中的“如何添加设备驱动程序”。**

3. **关闭系统。**

   ```
   # shutdown -i0 -g30 -y
   ```

   `-i0` 使系统进入 0 初始状态，即适用于关闭系统电源以添加和删除设备的状态。

   `-g30` 在 30 秒内关闭系统。缺省值为 60 秒。

   `-y` 在没有用户介入的情况下，继续关闭系统。否则，系统会提示您继续执行关机过程。

4. **在系统关闭后，选择以下方法之一关闭系统电源：**
   - 对于 SPARC 平台，如果显示 `ok` 提示，则可以安全关闭电源。
   - 对于 x86 平台，如果显示 `type any key to continue` 提示，则可以安全关闭电源。

5. **关闭所有外围设备的电源。**
   有关所有外围设备的电源开关的位置，请参阅外围设备附带的硬件安装指南。

6. **安装外围设备，确保要添加的设备与系统上的其他设备具有不同的目标号码。**
   通常，磁盘背面有一个小开关，用于选择目标号码。

   有关安装和连接设备的信息，请参阅外围设备附带的硬件安装指南。

7. **打开系统电源。**
   系统将引导至多用户模式，并会显示登录提示。

8. **通过尝试访问外围设备，验证是否已添加了该设备。**
   有关访问该设备的信息，请参见第 69 页中的“访问设备”。

▼ 如何添加设备驱动程序

在 Oracle Solaris 11 发行版中，pkg 命令用于将软件包添加到系统中。设备驱动程序软件包可能仍为 SRV4 软件包格式，所以下面的步骤使用 pkgadd 命令而非 pkg install 命令。

此过程假定已将设备添加到系统中。否则，请参见第 61 页中的 “不受支持的设备所需的内容”。

1. **成为管理员。**
   有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。
2. **将介质放入驱动器中。**
3. **安装驱动程序。**

   ```
   # pkgadd [-d] device package-name
   ```

4. **验证软件包是否已正确添加。**

   ```
   # pkgchk package-name
   #
   ```

   如果正确安装了软件包，则系统提示不会返回任何响应。

示例 5–2 添加设备驱动程序

以下示例说明如何安装和验证名为 XYZdrv 的软件包。

```
# pkgadd XYZdrv
(licensing messages displayed)
.
.
.
Installing XYZ Company driver as <XYZdrv>
.
.
.
Installation of <XYZdrv> was successful.
# pkgchk XYZdrv
#
```

访问设备

使用命令管理磁盘、文件系统和其他设备时，需要了解如何指定设备名称。在大多数情况下，可以使用逻辑设备名称来代表与系统连接的设备。逻辑设备名称和物理设备名称在系统中分别由逻辑设备文件和物理设备文件代表。

如何创建设备信息

首次引导系统时，将会创建设备分层结构，用于代表与系统连接的所有设备。内核使用设备分层结构信息将驱动程序与其相应的设备关联起来。内核还提供了一组指向执行特定操作的驱动程序的指针。

如何管理设备

devfs 文件系统管理 /devices 目录，该目录是系统中所有设备的名称空间。该目录代表包含实际的总线和设备地址的**物理**设备。

dev 文件系统管理 /dev 目录，该目录是**逻辑**设备名称的名称空间。

缺省情况下，devfsadm 命令尝试在系统中装入每个驱动程序，并连接至所有可能的设备实例。然后，devfsadm 将在 /devices 目录中创建设备文件并在 /dev 目录中创建逻辑链接。devfsadm 命令还可维护 path_to_inst 实例数据库。

为响应动态重新配置事件或文件系统访问而执行的 /dev 和 /devices 目录的更新，都将通过 devfsadmd（即 devfsadm 命令的守护进程版本）进行处理。引导系统时，将通过服务管理工具启动此守护进程。

由于 devfsadmd 守护进程会自动检测任何重新配置事件产生的设备配置变化，因此无需交互运行此命令。

有关更多信息，请参见下列参考信息：

- devfsadm(1M)
- dev(7FS)
- devfs(7FS)
- path_to_inst(4)

设备命名约定

在 Oracle Solaris OS 中以三种方式对设备进行引用。

- **物理设备名称**－代表设备信息的分层结构中的设备全路径名。物理设备名称是首次将设备添加到系统时创建的。物理设备名称位于 /devices 目录中。
- **实例名称**－代表系统上每个可能设备的内核的缩写名称。例如，sd0 和 sd1 代表两个磁盘设备的实例名称。实例名称会在 /etc/path_to_inst 文件中进行映射。
- **逻辑设备名称**－逻辑设备名称是首次将设备添加到系统时创建的。在大多数文件系统命令中，可以使用逻辑设备名称引用相应的设备。有关使用逻辑设备名称的文件命令的列表，请参见表 5–3。/dev 目录中的逻辑设备文件以符号形式链接至 /devices 目录中的物理设备文件。

前面的设备名称信息是使用以下命令显示的：

- `dmesg`
- `format`
- `sysdef`
- `prtconf`

逻辑磁盘设备名称

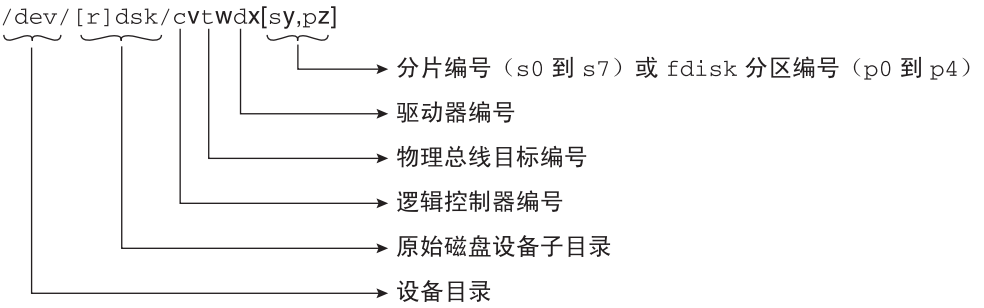
执行以下任务时可使用逻辑设备名称访问磁盘设备：

- 向系统中添加新磁盘。
- 在系统间移动磁盘。
- 访问或挂载驻留在本地磁盘上的文件系统。
- 备份本地文件系统。

许多管理命令都会使用引用磁盘分片或文件系统的参数。

通过指定以符号形式与磁盘设备链接的子目录（`/dev/dsk` 或 `/dev/rdsk`），后跟标识特定控制器、磁盘和分片的字符串来引用磁盘设备。

图 5–1 逻辑设备名称的说明



指定磁盘子目录

一些磁盘和文件管理命令需要使用**原始**（或**字符**）设备接口，或**块**设备接口。该区别是由从设备读取数据的方式导致的。

原始设备接口一次仅传输少量数据。块设备接口包括一次从中读取大数据块的缓冲区。

不同的命令需要不同的接口：

- 如果命令需要原始设备接口，请指定 /dev/rdsk 子目录。（rdsk 中的“r”代表“raw”（原始）。）
- 如果命令需要块设备接口，请指定 /dev/dsk 子目录。
- 如果不确定命令需要使用 /dev/dsk 还是 /dev/rdsk，请检查该命令的手册页。

下表显示了某些常用磁盘和文件系统命令所需的接口。

表 5–3 一些常用命令所需的设备接口类型

命令参考	接口类型	用法示例
dumpadm(1M)	Block（块）	dumpadm -d /dev/zvol/dsk/rpool/dump
prtvtoc(1M)	原始	prtvtoc /dev/rdsk/c0t0d0s0
swap(1M)	Block（块）	swap -a /dev/zvol/dsk/rpool/swap

直接控制器和面向总线的控制器

根据磁盘设备连接至直接控制器还是面向总线的控制器，可能要采用不同方式访问磁盘分区或磁盘分片。通常，直接控制器在逻辑设备名称中不包括**目标**标识符。

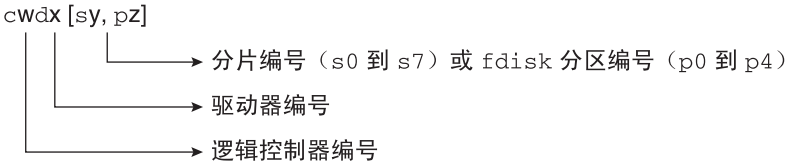
有关这两种类型控制器的约定将在以下小节中加以说明。

注 – 控制器编号是在系统初始化过程中自动指定的。这些编号为严格的逻辑编号，表示不会直接映射到物理控制器。

x86: 带有直接控制器的磁盘

要在基于 x86 的系统中带有 IDE 控制器的磁盘上指定分片，请遵循下图中显示的命名约定。

图 5–2 x86: 带有直接控制器的磁盘



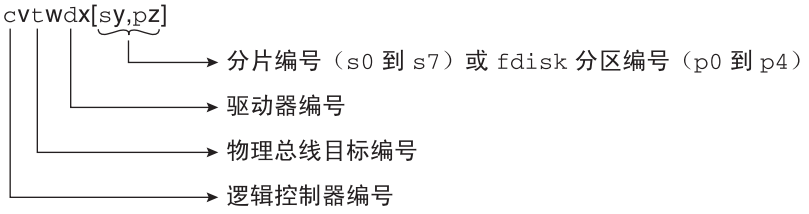
要指示整个 Solaris fdisk 分区，请指定分片 2 (s2)。

如果系统中仅有一个控制器，则 *w* 通常为 0。

带有面向总线的控制器的磁盘

要在带有面向总线的控制器（例如 SCSI）的磁盘上指定分片，请遵循下图中显示的命名约定。

图 5–3 带有面向总线的控制器的磁盘



在带有直接连接磁盘（如 UltraSPARC 系统上的 IDE 磁盘）的基于 SPARC 的系统上，命名约定与带有面向总线的控制器的系统的命名约定相同。

如果系统中仅有一个控制器，则 *w* 通常为 0。

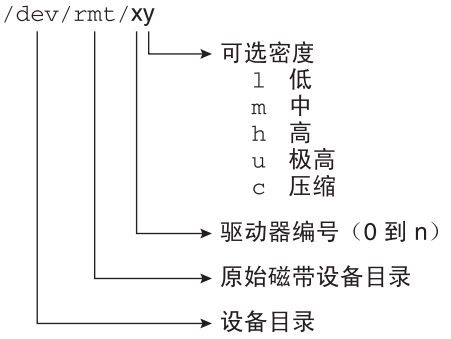
对于 SCSI 控制器，*x* 是通过装置背面的开关设置的目标地址，*y* 是连接至目标的驱动器的逻辑单元号 (logical unit number, LUN)。如果磁盘具有嵌入式控制器，则*y* 通常为 0。

要指示整个磁盘，请指定分片 2 (s2)。

逻辑磁带设备名称

逻辑磁带设备文件位于 /dev/rmt/* 目录中，作为 /devices 目录的符号链接。

图 5–4 逻辑磁带设备名称



连接至系统的第一个磁带设备为 0 (/dev/rmt/0)。磁带密度值（l、m、h、c 和 u）在第 21 章，管理磁带机（任务）中介绍。

逻辑可移除介质设备名称

由于可移除介质由可移除介质管理服务来进行管理，因此除非要手动挂载介质，否则通常不会使用逻辑设备名称。

代表系统上的可移除介质设备的逻辑设备名称在第 3 章，访问可移除介质（任务）中加以介绍。

◆ ◆ ◆ 第 6 章

6 动态配置设备（任务）

本章提供有关在 Oracle Solaris OS 中动态配置设备的说明。如果系统组件支持热插拔，当系统仍在运行时，可以在 Oracle Solaris OS 中添加、移除或更换设备。如果系统组件不支持热插拔，则可以重新引导系统以重新配置设备。

有关与动态配置设备关联的过程信息，请参见以下内容：

- 第 81 页中的“使用 `cfgadm` 命令执行 SCSI 热插拔（任务列表）”
- 第 90 页中的“使用 `cfgadm` 命令执行 PCI 或 PCIe 热插拔（任务列表）”
- 第 96 页中的“使用 `cfgadm` 命令进行 SATA 热插拔”
- 第 99 页中的“应用程序开发者 RCM 脚本（任务列表）”
- 第 100 页中的“系统管理员 RCM 脚本（任务列表）”

有关使用 `cfgadm` 命令热插拔 USB 设备的信息，请参见第 143 页中的“使用 `cfgadm` 命令热插拔 USB 设备”。

有关使用 `cfgadm` 命令热插拔 InfiniBand 设备的信息，请参见第 9 章，使用 InfiniBand 设备（概述/任务）。

有关访问设备的信息，请参见第 69 页中的“访问设备”。

动态重新配置和热插拔

热插拔是指在系统运行的同时以物理方式添加、移除或更换系统组件的能力。**动态重新配置**是指热插拔系统组件的能力。此术语还指在系统中四处移动系统资源（硬件和软件），或以某种方式禁用这些系统资源而不以物理方式将其从系统中移除的一般能力。

一般情况下，可以热插拔以下总线类型：

- USB
- 光纤通道
- 1394

- ATA
- SCSI
- Infiniband

此外，还可以使用 cfgadm 命令热插拔以下设备：

- SPARC 和 x86 平台上的 USB 设备
- SPARC 和 x86 平台上的 SCSI 或 SAS 设备
- SPARC 和 x86 平台上的 PCI 设备
- SPARC 和 x86 平台上的 PCIe 设备
- SPARC 和 x86 平台上的 SATA 设备
- SPARC 和 x86 平台上的 InfiniBand 设备

cfgadm 命令包括以下功能：

- 显示系统组件状态
- 测试系统组件
- 更改组件配置
- 显示配置帮助消息

使用 cfgadm 命令重新配置系统组件的益处是可在系统运行的同时添加、移除或更换组件。另一益处是 cfgadm 命令可引导逐步完成添加、移除或更换系统组件所需执行的步骤。

有关热插拔组件的逐步说明，请参见以下内容：

- 第 81 页中的 “使用 cfgadm 命令进行 SCSI 热插拔”
- 第 91 页中的 “使用 cfgadm 命令执行 PCI 或 PCIe 热插拔”
- 第 96 页中的 “使用 cfgadm 命令进行 SATA 热插拔”
- cfgadm(1M)

注 – 并非所有的 SCSI 和 PCI 控制器都支持使用 cfgadm 命令进行热插拔。

作为 Oracle 高可用性策略的一部分，动态重新配置预计会与其他分层产品（如替代路径设置软件或故障转移软件）结合使用。这两种产品都可在出现设备故障的情况下提供容错功能。

如果没有任何高可用性软件，则可通过手动停止相应的应用程序，取消挂载非关键的文件系统，然后继续执行添加或删除操作来更换出现故障的设备。

注 – 某些系统同时具有可热插拔和不可热插拔的插槽。有关在特定硬件配置（如在企业级系统上）中热插拔设备的信息，请参阅硬件配置文档。

连接点

cfgadm 命令可以显示有关**连接点**的信息，连接点是指系统中可以执行动态重新配置操作的位置。

连接点由以下几项组成：

- **插卡**，表示可在系统中配置的硬件组件
- **插座**，是指插入插卡的位置

连接点由逻辑连接点 ID 和物理连接点 ID (Ap_Ids) 表示。物理 Ap_Id 是连接点的物理路径名。逻辑 Ap_Id 是物理 Ap_Id 的用户友好替代项。有关 Ap_Id 的更多信息，请参阅 cfgadm(1M)。

SCSI 主机总线适配器 (Host Bus Adapter, HBA) 或 SCSI 控制器的逻辑 Ap_Id 通常由控制器编号表示，如 c0。

如果未向 SCSI HBA 指定控制器编号，则将提供内部生成的唯一标识符。以下是 SCSI 控制器的唯一标识符的示例：

```
fas1:scsi
```

SCSI 设备的逻辑 Ap_Id 通常具有以下格式：

HBA-logical-apid::device-identifier

在以下示例中，c0 是 SCSI HBA 的逻辑 Ap_Id：

```
c0::dsk/c0t3d0
```

设备标识符通常由 /dev 目录中的设备的逻辑设备名称派生而来。例如，逻辑设备名称为 /dev/rmt/1 的磁带设备具有以下逻辑 Ap_Id：

```
c0::rmt/1
```

如果 SCSI 设备的逻辑 Ap_Id 无法从 /dev 目录中的逻辑名称派生而来，则将提供内部生成的唯一标识符。以下是 /dev/rmt/1 磁带设备的标识符的示例：

```
c0::st4
```

有关 SCSI Ap_Id 的更多信息，请参阅 cfgadm_scsi(1M)。

cfgadm 命令按照一组通用状态（如已配置和未配置）和操作（如连接、配置、取消配置等）来表示所有资源和动态重新配置操作。有关这些通用状态和操作的更多信息，请参见 cfgadm(1M)。

下表显示 SCSI HBA 连接点的插座和插卡的状态。

插座状态	说明	插卡状态	说明
empty（空）	N/A（对于 SCSI HBA）	configured（已配置）	在总线上配置了一个或多个设备
disconnected（断开）	总线处于静止状态	unconfigured（未配置）	未配置任何设备
connected（已连接）	总线处于活动状态		

下表显示 SCSI 设备连接点的插座和插卡的状态。

插座状态	说明	插卡状态	说明
empty（空）	N/A（对于 SCSI 设备）	configured（已配置）	设备已配置
disconnected（断开）	总线处于静止状态	unconfigured（未配置）	设备未配置
connected（已连接）	总线处于活动状态		

如果特殊硬件未另行指示其他状态，则 SCSI 连接点的状态是未知的。有关显示 SCSI 组件信息的说明，请参见第 82 页中的“如何显示有关 SCSI 设备的信息”。

拆离 PCI 或 PCIe 适配卡

如果设备驱动程序支持热插拔，则可以移除承载非重要系统资源的 PCI 适配卡。如果 PCI 适配卡是重要的系统资源，则不可将其拆离。要使 PCI 适配卡可拆离，必须满足以下条件：

- 设备驱动程序必须支持热插拔。
- 必须可通过替代路径访问关键资源。

例如，如果系统中仅安装了一块以太网卡，则拆离该以太网卡必然会断开网络连接。此拆离操作需要其他分层软件支持，以使网络连接保持活动状态。

连接 PCI 或 PCIe 适配卡

只要满足以下条件，便可以向系统中添加 PCI 适配卡：

- 具有可用插槽。
- 设备驱动程序支持对此适配卡进行热插拔。

有关添加或移除 PCI 适配卡的逐步说明，请参见第 91 页中的“使用 cfgadm 命令执行 PCI 或 PCIe 热插拔”。

使用 (hotplug) 命令进行 PCIe 热插拔

可以使用 hotplug 命令管理可热插拔的连接，其中连接可以是连接器或端口，仅限于在 PCI Express (PCIe) 和 PCI SHPC 设备上。**热插拔连接器**是系统中的物理表现表示，组件插入该系统或从中移除。**热插拔端口**是系统设备树中的逻辑表示，在系统设备树中管理设备和系统的连接。

可以使用 hotplug 功能使设备（甚至是主板集成设备）脱机或联机而无需物理地将设备添加至系统或将设备从系统中移除。

必须启用 hotplug 服务以使用 hotplug 命令管理设备。

```
# svcadm enable svc:/system/hotplug:default
```

以下示例说明了 hotplug 命令的用法。

系统中所有 PCI/PCIe 可热插拔连接器/端口（虚拟的或物理的）显示如下：

```
# hotplug list -lv
```

在 PCIe 插槽中配置以太网卡。例如：

```
# hotplug enable /pci0,0 pcie0
```

在 PCIe 插槽中取消配置以太网卡。例如：

```
# hotplug disable /pci0,0 pcie0
```

使 PCI 设备节点脱机，意味着拆离该节点的设备驱动程序。例如：

```
# hotplug offline /pci0,0/pci1 pci.0,2
```

使 PCI 设备节点联机，意味着连接该节点的驱动程序。例如：

```
# hotplug online /pci0,0/pci1 pci.0,2
```

安装 IOV 物理功能的依赖端口。例如：

```
# hotplug install /pci@400/pci@1/pci@0/pci@4 pci.0,1
```

然后，显示检测出的 IOV 虚拟功能。例如：

```
# hotplug list -v /pci@400/pci@1/pci@0/pci@4 pci.0,1
<pci.0,1>  (ONLINE)
     { IOV physical function }
     { IOV virtual function 'pci.0,81' }
     { IOV virtual function 'pci.0,83' }
     { IOV virtual function 'pci.0,85' }
     { IOV virtual function 'pci.0,87' }
<pci.0,81>  (OFFLINE)
```

```
ethernet@0,81
<pci.0,83>  (OFFLINE)
ethernet@0,83
<pci.0,85>  (OFFLINE)
ethernet@0,85
<pci.0,87>  (OFFLINE)
ethernet@0,87
```

卸载 IOV 物理功能的依赖端口。例如：

```
# hotplug uninstall /pci@400/pci@1/pci@0/pci@4 pci.0,0
```

如果依赖 IOV 虚拟功能忙，则此操作失败。例如：

```
# hotplug uninstall /pci@400/pci@1/pci@0/pci@4 pci.0,0
ERROR: devices or resources are busy.
ethernet@0,81:
    { Network interface igbvf1 }
    { igbvf1: hosts IP addresses: 10.0.0.1 }
    { Plumbed IP Address }
```

PCI 热插拔操作 (hotplug) 故障排除

可能会看到热插拔端口中连接设备的以下维护状态。

```
/pci@0,0/pci10de,5d@e <pci.a,1> (MAINTENANCE)
.
.
./pci@0,0/pci108e,534a@d <pci.0,0> (MAINTENANCE-SUSPENDED)
```

上述消息表明发生了故障事件或维护操作。`MAINTENANCE` 状态表示设备正在使用，但并非完全可操作。`MAINTENANCE-SUSPENDED` 状态表示该设备由于维护操作而**即时暂停**。例如，重新配置设备硬件。

必须运行以下服务以使用 `hotplug` 命令。

```
svc:/system/hotplug:default
```

否则，您将看到以下消息：

```
ERROR: hotplug service is not available.
```

在没有任何支持的 I/O 总线的系统上会显示以下错误消息：

```
ERROR: there are no connections to display.
       (See hotplug(1m) for more information.)
```

上述消息可能意味着系统中可能有其他可热插拔的 I/O 设备，但是您需要使用 `cfgadm` 命令而非 `hotplug` 命令来管理这些设备。

使用 cfgadm 命令执行 SCSI 热插拔（任务列表）

任务	说明	参考
显示有关 SCSI 设备的信息。	显示有关 SCSI 控制器和设备的信息。	第 82 页中的 “如何显示有关 SCSI 设备的信息”
取消配置 SCSI 控制器。	取消配置 SCSI 控制器。	第 82 页中的 “如何取消配置 SCSI 控制器”
配置 SCSI 控制器。	配置以前取消配置的 SCSI 控制器。	第 83 页中的 “如何配置 SCSI 控制器”
配置 SCSI 设备。	配置特定的 SCSI 设备。	第 83 页中的 “如何配置 SCSI 设备”
断开 SCSI 控制器的连接。	断开特定 SCSI 控制器的连接。	第 84 页中的 “如何断开 SCSI 控制器连接”
连接 SCSI 控制器。	连接以前断开的特定 SCSI 控制器。	第 85 页中的 “SPARC: 如何连接 SCSI 控制器”
将 SCSI 设备添加到 SCSI 总线中。	将特定的 SCSI 设备添加到 SCSI 总线中。	第 85 页中的 “SPARC: 如何将 SCSI 设备添加到 SCSI 总线中”
更换 SCSI 控制器上的相同设备。	将 SCSI 总线上的设备更换为同一类型的其他设备。	第 86 页中的 “SPARC: 如何更换 SCSI 控制器上的相同设备”
移除 SCSI 设备。	从系统中移除 SCSI 设备。	第 88 页中的 “SPARC: 如何移除 SCSI 设备”
对 SCSI 配置问题进行故障排除。	解决失败的 SCSI 取消配置操作。	第 90 页中的 “如何解决失败的 SCSI 取消配置操作”

使用 cfgadm 命令进行 SCSI 热插拔

本节介绍可使用 `cfgadm` 命令执行的各种 SCSI 热插拔过程。

注 – 一般情况下，SCSI 框架支持 SCSI 设备的热插拔。但是，您应该参考硬件文档，以确认 SCSI 设备是否支持热插拔。

这些过程以特定设备为例，说明如何使用 `cfgadm` 命令来热插拔 SCSI 组件。您提供的以及 `cfgadm` 命令显示的设备信息取决于系统配置。

本节介绍的所有过程需要管理特权，一般不授予用户帐户。有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。

▼ 如何显示有关 SCSI 设备的信息

以下过程在使用 `cfgadm` 命令显示设备类型配置信息的示例中使用 SCSI 控制器 `c0` 和 `c1` 以及与其连接的设备。

注 – 如果 `cfgadm` 命令不支持 SCSI 设备，则该设备不会显示在 `cfgadm` 命令输出中。

---

1. **成为管理员。**

2. **显示有关系统上的连接点的信息。**

```
# cfgadm -l
Ap_Id                Type         Receptacle   Occupant     Condition
c0                   scsi-bus     connected    configured   unknown
c1                   scsi-bus     connected    configured   unknown
```

本示例中，`c0` 和 `c1` 代表两个 SCSI 控制器。

3. **显示有关系统的 SCSI 控制器及其连接设备的信息。**

```
# cfgadm -al
Ap_Id                Type         Receptacle   Occupant     Condition
c0                   scsi-bus     connected    configured   unknown
c0::dsk/c0t0d0       disk         connected    configured   unknown
c0::rmt/0            tape         connected    configured   unknown
c1                   scsi-bus     connected    configured   unknown
c1::dsk/c1t3d0       disk         connected    configured   unknown
c1::dsk/c1t4d0       unavailable  connected    unconfigured unknown
```

注 – `cfgadm -l` 命令显示有关 SCSI HBA 而不是 SCSI 设备的信息。使用 `cfgadm -al` 命令可显示有关 SCSI 设备（如磁盘和磁带）的信息。

▼ 如何取消配置 SCSI 控制器

以下过程在取消配置 SCSI 控制器的示例中使用 SCSI 控制器 `c1`。

1. **成为管理员。**

2. **取消配置 SCSI 控制器。**

```
# cfgadm -c unconfigure c1
```

3. **验证是否已取消配置 SCSI 控制器。**

```
# cfgadm -al
Ap_Id                 Type         Receptacle   Occupant     Condition
c0                    scsi-bus     connected    configured   unknown
c0::dsk/c0t0d0        disk         connected    configured   unknown
c0::rmt/0             tape         connected    configured   unknown
c1                    scsi-bus     connected    unconfigured unknown
```

请注意，c1 的 Occupant 列指定 unconfigured，表明 SCSI 总线没有已配置的插卡。

如果取消配置操作失败，请参见第 90 页中的 “如何解决失败的 SCSI 取消配置操作”。

▼ 如何配置 SCSI 控制器

以下过程在配置 SCSI 控制器的示例中使用 SCSI 控制器 c1。

1. **成为管理员。**

2. **配置 SCSI 控制器。**

```
# cfgadm -c configure c1
```

3. **验证 SCSI 控制器是否已配置。**

```
# cfgadm -al
Ap_Id                Type         Receptacle   Occupant     Condition
c0                   scsi-bus     connected    configured   unknown
c0::dsk/c0t0d0       disk         connected    configured   unknown
c0::rmt/0            tape         connected    configured   unknown
c1                   scsi-bus     connected    configured   unknown
c1::dsk/c1t3d0       disk         connected    configured   unknown
c1::dsk/c1t4d0       unavailable  connected    unconfigured unknown
```

前面的取消配置过程移除了 SCSI 总线上的所有设备。现在，所有设备都已配置回系统中。

▼ 如何配置 SCSI 设备

以下过程在配置 SCSI 设备的示例中使用 SCSI 磁盘 c1t4d0。

1. **成为管理员。**

2. **确定要配置的设备。**

```
# cfgadm -al
Ap_Id                Type          Receptacle   Occupant     Condition
c0                   scsi-bus      connected    configured   unknown
c0::dsk/c0t0d0       disk          connected    configured   unknown
c0::rmt/0            tape          connected    configured   unknown
c1                   scsi-bus      connected    configured   unknown
c1::dsk/c1t3d0       disk          connected    configured   unknown
c1::dsk/c1t4d0       unavailable   connected    unconfigured unknown
```

3. **配置 SCSI 设备。**

```
# cfgadm -c configure c1::dsk/c1t4d0
```

4 **验证 SCSI 设备是否已配置。**

```
# cfgadm -al
Ap_Id                  Type          Receptacle   Occupant     Condition
c0                     scsi-bus      connected    configured   unknown
c0::dsk/c0t0d0         disk          connected    configured   unknown
c0::rmt/0              tape          connected    configured   unknown
c1                     scsi-bus      connected    configured   unknown
c1::dsk/c1t3d0         disk          connected    configured   unknown
c1::dsk/c1t4d0         disk          connected    configured   unknown
```

▼ 如何断开 SCSI 控制器连接

注意 – 断开 SCSI 设备连接时必须非常谨慎，特别是在处理包含关键文件系统（如根目录 (/)、`usr`、 `var` 和 `swap` 分区）的磁盘的控制器时。动态重新配置软件无法检测到系统挂起可能导致的所有情况。使用此过程时应谨慎。

以下过程在断开 SCSI 设备连接的示例中使用 SCSI 控制器 `c1`。

1 **成为管理员。**

2 **验证断开设备连接之前是否已连接该设备。**

```
# cfgadm -al
Ap_Id                  Type          Receptacle   Occupant     Condition
c0                     scsi-bus      connected    configured   unknown
c0::dsk/c0t0d0         disk          connected    configured   unknown
c0::rmt/0              tape          connected    configured   unknown
c1                     scsi-bus      connected    configured   unknown
c1::dsk/c1t3d0         disk          connected    configured   unknown
c1::dsk/c1t4d0         disk          connected    configured   unknown
```

3 **断开 SCSI 控制器的连接。**

```
# cfgadm -c disconnect c1
WARNING: Disconnecting critical partitions may cause system hang.
Continue (yes/no)? y
```

注意 – 此命令可以暂停 SCSI 总线上的所有 I/O 活动，直到使用 `cfgadm -c connect` 命令为止。`cfgadm` 命令可执行一些基本检查，以防止断开关键分区的连接，但不能检测到所有情况。如果此命令使用不当，则可能会导致系统挂起并且可能需要重新引导系统。

4 **验证 SCSI 总线是否已断开连接。**

```
# cfgadm -al
Ap_Id                  Type          Receptacle   Occupant     Condition
c0                     scsi-bus      connected    configured   unknown
c0::dsk/c0t0d0         disk          connected    configured   unknown
```

```
c0::rmt/0            tape          connected    configured   unknown
c1                   unavailable   disconnected configured   unknown
c1::dsk/c1t3d0       unavailable   disconnected configured   unknown
c1::dsk/c1t4d0       unavailable   disconnected configured   unknown
```

控制器及其连接的所有设备都将与系统断开连接。

▼ SPARC: 如何连接 SCSI 控制器

以下过程在连接 SCSI 控制器的示例中使用 SCSI 控制器 `c1`。

1 成为管理员。

2 验证连接设备之前该设备是否已断开连接。

```
# cfgadm -al
Ap_Id                Type          Receptacle   Occupant     Condition
c0                   scsi-bus      connected    configured   unknown
c0::dsk/c0t0d0       disk          connected    configured   unknown
c0::rmt/0            tape          connected    configured   unknown
c1                   unavailable   disconnected configured   unknown
c1::dsk/c1t3d0       unavailable   disconnected configured   unknown
c1::dsk/c1t4d0       unavailable   disconnected configured   unknown
```

3 连接 SCSI 控制器。

```
# cfgadm -c connect c1
```

4 验证 SCSI 控制器是否已连接。

```
# cfgadm -al
Ap_Id                Type          Receptacle   Occupant     Condition
c0                   scsi-bus      connected    configured   unknown
c0::dsk/c0t0d0       disk          connected    configured   unknown
c0::rmt/0            tape          connected    configured   unknown
c1                   scsi-bus      connected    configured   unknown
c1::dsk/c1t3d0       disk          connected    configured   unknown
c1::dsk/c1t4d0       disk          connected    configured   unknown
```

▼ SPARC: 如何将 SCSI 设备添加到 SCSI 总线中

在如何将 SCSI 设备添加到 SCSI 总线的示例中使用的是 SCSI 控制器 `c1`。

注 – 添加设备时，应指定与设备连接的 SCSI HBA（控制器）的 `Ap_Id`，而不是设备本身的 `Ap_Id`。

---

1 成为管理员。

2 **确定当前的 SCSI 配置。**

```
# cfgadm -al
Ap_Id                  Type          Receptacle   Occupant     Condition
c0                     scsi-bus      connected    configured   unknown
c0::dsk/c0t0d0         disk          connected    configured   unknown
c0::rmt/0              tape          connected    configured   unknown
c1                     scsi-bus      connected    configured   unknown
c1::dsk/c1t3d0         disk          connected    configured   unknown
```

3 **将 SCSI 设备添加到 SCSI 总线。**

a. **键入以下 `cfgadm` 命令。**

例如：

```
# cfgadm -x insert_device c1
Adding device to SCSI HBA: /devices/sbus@1f,0/SUNW,fas@1,8800000
This operation will suspend activity on SCSI bus: c1
```

b. **在 `Continue (yes/no)?` 提示符下键入 `y` 以继续操作。**

```
Continue (yes/no)? y
SCSI bus quiesced successfully.
It is now safe to proceed with hotplug operation.
```

正在执行热插拔操作的同时，SCSI 总线上的 I/O 活动会暂停。

c. **连接设备，然后打开设备电源。**

d. **在 `Enter y if operation is complete or n to abort (yes/no)?` 提示符下键入 `y`。**

```
Enter y if operation is complete or n to abort (yes/no)? y
```

4 **验证是否已添加了设备。**

```
# cfgadm -al
Ap_Id                  Type          Receptacle   Occupant     Condition
c0                     scsi-bus      connected    configured   unknown
c0::dsk/c0t0d0         disk          connected    configured   unknown
c0::rmt/0              tape          connected    configured   unknown
c1                     scsi-bus      connected    configured   unknown
c1::dsk/c1t3d0         disk          connected    configured   unknown
c1::dsk/c1t4d0         disk          connected    configured   unknown
```

新磁盘已添加到控制器 `c1` 中。

▼ SPARC: 如何更换 SCSI 控制器上的相同设备

以下过程在 SCSI 控制器上更换相同设备的示例中使用 SCSI 磁盘 `c1t4d0`。

尝试在 SCSI 控制器上更换相同设备时，请注意以下情况：

- 如果要更换属于 ZFS 根池的镜像或非镜像引导设备，请参见以下参考资料：

http://www.oracle.com/
technetwork/articles/servers-storage-admin/
o11-091-sol-dis-recovery-489183.html

- 如果设备由传统卷管理软件控制，请参考相应的卷管理文档以了解更换设备的具体步骤。如果您享有有效的 Oracle 支持计划，请参见以下参考资料：
  - 如果使用 Solaris Volume Manager (SVM)，请参见 MOS (My Oracle Support, MOS) 文档 1010753.1。
  - 如果使用 Veritas Volume Manager (VxVM)，请参见 MOS (My Oracle Support, MOS) 文档 1003122.1、1011782.1 和 1002285.1。

1 成为管理员。

2 确定当前的 SCSI 配置。

```
# cfgadm -al
Ap_Id                Type          Receptacle   Occupant     Condition
c0                   scsi-bus      connected    configured   unknown
c0::dsk/c0t0d0       disk          connected    configured   unknown
c0::rmt/0            tape          connected    configured   unknown
c1                   scsi-bus      connected    configured   unknown
c1::dsk/c1t3d0       disk          connected    configured   unknown
c1::dsk/c1t4d0       disk          connected    configured   unknown
```

3 将 SCSI 总线上的设备更换为同一类型的其他设备。

a. 键入以下 `cfgadm` 命令。

例如：

```
# cfgadm -x replace_device c1::dsk/c1t4d0
Replacing SCSI device: /devices/sbus@1f,0/SUNW,fas@1,8800000/sd@4,0
This operation will suspend activity on SCSI bus: c1
```

b. 在 `Continue (yes/no)?` 提示符下键入 `y` 以继续操作。

正在执行热插拔操作的同时，SCSI 总线上的 I/O 活动会暂停。

```
Continue (yes/no)? y
SCSI bus quiesced successfully.
It is now safe to proceed with hotplug operation.
```

c. 关闭要移除的设备的电源，然后将其移除。

d. 添加更换设备。然后，关闭该设备的电源。

更换设备应该与要移除的设备类型相同并且具有同一地址（目标和 LUN）。

e. 在 `Enter y if operation is complete or n to abort (yes/no)?` 提示符下键入 `y`。

```
Enter y if operation is complete or n to abort (yes/no)? y
```

4 **验证是否已更换了设备。**

```
# cfgadm -al
Ap_Id                Type          Receptacle   Occupant     Condition
c0                   scsi-bus      connected    configured   unknown
c0::dsk/c0t0d0       disk          connected    configured   unknown
c0::rmt/0            tape          connected    configured   unknown
c1                   scsi-bus      connected    configured   unknown
c1::dsk/c1t3d0       disk          connected    configured   unknown
c1::dsk/c1t4d0       disk          connected    configured   unknown
```

▼ SPARC: 如何移除 SCSI 设备

以下过程在移除 SCSI 控制器上的设备示例中使用 SCSI 磁盘 `c1t4d0`。

1 **成为管理员。**

2 **确定当前的 SCSI 配置。**

```
# cfgadm -al
Ap_Id                Type          Receptacle   Occupant     Condition
c0                   scsi-bus      connected    configured   unknown
c0::dsk/c0t0d0       disk          connected    configured   unknown
c0::rmt/0            tape          connected    configured   unknown
c1                   scsi-bus      connected    configured   unknown
c1::dsk/c1t3d0       disk          connected    configured   unknown
c1::dsk/c1t4d0       disk          connected    configured   unknown
```

3 **从系统中移除 SCSI 设备。**

a. **键入以下 `cfgadm` 命令。**

例如：

```
# cfgadm -x remove_device c1::dsk/c1t4d0
Removing SCSI device: /devices/sbus@1f,0/SUNW,fas@1,8800000/sd@4,0
This operation will suspend activity on SCSI bus: c1
```

b. **在 `Continue (yes/no)?` 提示符下键入 `y` 以继续操作。**

```
Continue (yes/no)? y
SCSI bus quiesced successfully.
It is now safe to proceed with hotplug operation.
```

正在执行热插拔操作的同时，SCSI 总线上的 I/O 活动会暂停。

c. **关闭要移除的设备的电源，然后将其移除。**

d. **在 `Enter y if operation is complete or n to abort (yes/no)?` 提示符下键入 `y`。**

```
Enter y if operation is complete or n to abort (yes/no)? y
```

注 – 如果是从 SCSI RAID 阵列中移除 SCSI RAID 设备，那么必须执行此步骤。

4 验证是否已从系统中移除了设备。

```
# cfgadm -al
Ap_Id                Type          Receptacle   Occupant     Condition
c0                   scsi-bus      connected    configured   unknown
c0::dsk/c0t0d0       disk          connected    configured   unknown
c0::rmt/0            tape          connected    configured   unknown
c1                   scsi-bus      connected    configured   unknown
c1::dsk/c1t3d0       disk          connected    configured   unknown
```

对 SCSI 配置问题进行故障排除

本节提供有关 SCSI 配置问题的错误消息和可能的解决方法。有关对 SCSI 配置问题进行故障排除的更多信息，请参见 cfgadm(1M)。

错误消息

```
cfgadm: Component system is busy, try again: failed to offline:
     device-path
          Resource              Information
     ------------------  -------------------------
     /dev/dsk/c1t0d0s0   mounted filesystem "/file-system"
```

原因

尝试移除或更换已挂载文件系统的设备。

解决方法

取消挂载错误消息中列出的文件系统，并重试 cfgadm 操作。

错误消息

```
cfgadm: Component system is busy, try again: failed to offline:
     device-path
          Resource              Information
     ------------------  -------------------------
     /dev/dsk/device-name    swap area
```

原因

如果使用 cfgadm 命令移除系统资源（如交换设备或专用转储设备），则在系统资源仍然处于活动状态时将显示一条类似的错误消息。

解决方法

取消配置指定设备上的交换区域，并重试 cfgadm 操作。

错误消息

```
cfgadm: Component system is busy, try again: failed to offline:
     device-path
          Resource              Information
     ------------------  -------------------------
     /dev/dsk/device-name    dump device (swap)
```

原因
尝试移除或更换在交换区域中配置的转储设备。

解决方法
取消配置在交换区域中配置的转储设备，并重试 `cfgadm` 操作。

错误消息

```
cfgadm: Component system is busy, try again: failed to offline:
     device-path
          Resource             Information
-----------------  -------------------------
/dev/dsk/device-name   dump device (dedicated)
```

原因
尝试移除或更换专用转储设备。

解决方法
取消配置该专用转储设备，并重试 `cfgadm` 操作。

▼ 如何解决失败的 SCSI 取消配置操作

如果一个或多个目标设备繁忙并且 SCSI 取消配置操作失败，请使用此过程。否则，将来对此控制器和目标设备进行动态重新配置操作会失败，并会显示 `dr in progress` 消息。

1 **成为管理员。**

2 **重新配置控制器。**

```
# cfgadm -c configure device-name
```

使用 cfgadm 命令执行 PCI 或 PCIe 热插拔（任务列表）

以下任务列表介绍了在系统上管理 PCI 或 PCIe 设备的任务。

任务	说明	参考
显示 PCI 插槽配置信息。	显示系统上的 PCI 可热插拔设备和插槽的状态。	第 91 页中的“如何显示 PCI 插槽配置信息”
移除 PCI 适配卡。	取消配置该卡，断开插槽的电源，然后从系统中移除该卡。	第 93 页中的“如何移除 PCI 适配卡”
添加 PCI 适配卡。	将适配卡插入可热插拔的插槽中，连接插槽的电源，然后配置该卡。	第 94 页中的“如何添加 PCI 适配卡”

任务	说明	参考
对 PCI 配置问题进行故障排除。	确定错误消息和可能的解决方法，以解决 PCI 配置问题。	第 96 页中的 “对 PCI 配置问题进行故障排除”

使用 cfgadm 命令执行 PCI 或 PCIe 热插拔

本节提供有关热插拔 SPARC 和 x86 系统上的 PCI 或 PCIe 适配卡的逐步说明。

除了 `cfgadm` 命令以外，在热插拔操作过程中，`prtconf` 命令也非常有帮助。`prtconf` 命令可以显示与硬件有关的其他配置信息。

添加硬件后，可以使用 `prtconf` 命令来验证该硬件是否正确配置。例如，在配置操作之后，使用 `prtconf -D` 命令可验证是否已向新安装的硬件设备附加了驱动程序。如果在配置硬件之前未将设备驱动程序添加到系统中，则可以使用 `add_drv` 命令手动添加该驱动程序。

有关更多信息，请参见 `prtconf(1M)` 和 `add_drv(1M)`。

本节介绍的所有过程需要管理特权，一般不授予用户帐户。有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。

在这些示例中，为了简便起见，仅列出了 PCI 连接点。系统上显示的连接点取决于系统配置。

PCIe LED 指示灯行为

您可能需要观察系统上的 LED 指示灯，以直观地了解有关插槽热插拔操作的状态。如果使用 PCI Express，则 LED 的行为符合 PCI Express 规范中定义的行为，否则该行为可能与平台相关。

有关具体的详细信息，请参阅平台指南。如果使用 PCI Express，则按“Attention”（注意）按钮时，电源指示灯将会闪烁，表示状态转换开始。状态转换结束后，即会停止闪烁。

▼ 如何显示 PCI 插槽配置信息

此过程已进行了更新，包括 PCIe 配置信息。

`cfgadm` 命令可显示系统上的 PCI 可热插拔设备和插槽的状态。有关更多信息，请参见 `cfgadm(1M)`。

1 **成为管理员。**

2 显示 PCI 配置信息。

- 显示 PCI 插槽配置信息。

  例如：

```
# cfgadm
Ap_Id                Type         Receptacle   Occupant     Condition
pci1:hpc0_slot0      unknown      empty        unconfigured unknown
pci1:hpc0_slot1      unknown      empty        unconfigured unknown
pci1:hpc0_slot2      unknown      empty        unconfigured unknown
pci1:hpc0_slot3      ethernet/hp  connected    configured   ok
pci1:hpc0_slot4      unknown      empty        unconfigured unknown
```

- 显示特定的 PCI 设备信息。

  例如：

```
# cfgadm -s "cols=ap_id:type:info" pci
Ap_Id                Type         Information
pci1:hpc0_slot0      unknown      Slot 7
pci1:hpc0_slot1      unknown      Slot 8
pci1:hpc0_slot2      unknown      Slot 9
pci1:hpc0_slot3      ethernet/hp  Slot 10
pci1:hpc0_slot4      unknown      Slot 11
```

  逻辑 `Ap_Id` `pci1:hpc0_slot0` 是可热插拔插槽 `Slot 7` 的逻辑 `Ap_Id`。组件 `hpc0` 表示此插槽的可热插拔适配卡，`pci1` 表示 PCI 总线实例。`Type` 字段表示插槽中存在的 PCI 适配卡的类型。

- 显示 PCIe 插槽配置信息。

  例如：

```
# cfgadm pci
Ap_Id                          Type         Receptacle   Occupant     Condition
pcie1                          unknown      empty        unconfigured unknown
pcie2                          unknown      empty        unconfigured unknown
pcie3                          unknown      empty        unconfigured unknown
pcie4                          etherne/hp   connected    configured   ok
pcie5                          pci-pci/hp   connected    configured   ok
pcie6                          unknown      disconnected unconfigured unknown
```

- 显示特定的 PCIe 设备信息。

  例如：

```
# cfgadm -s "cols=ap_id:busy:o_state" pci
Ap_Id                          Busy     Occupant
pcie1                          n        unconfigured
pcie2                          n        unconfigured
pcie3                          n        unconfigured
pcie4                          n        configured
pcie5                          n        configured
pcie6                          n        configured
```

注 – 在大多数情况下，逻辑 `Ap_Id` 都应该与系统机箱上用丝网印花法标记的插槽标签匹配。有关可热插拔插槽的 `cfgadm` 输出，请参阅平台指南。`Busy` 字段可确保尝试热插拔操作之前 `Ap_Id` 未转换为其他状态。

▼ 如何移除 PCI 适配卡

以下过程已经更新，可用于移除 PCIe 适配卡。但是，无论使用 PCI 还是 PCIe，移除适配卡的过程都是相同的。

1. **成为管理员。**

2. **确定 PCI 适配卡所在的插槽。**

   例如：

```
# cfgadm pci
Ap_Id                          Type         Receptacle   Occupant     Condition
pcie1                          unknown      empty        unconfigured unknown
pcie2                          unknown      empty        unconfigured unknown
pcie3                          unknown      empty        unconfigured unknown
pcie4                          etherne/hp   connected    configured   ok
pcie5                          pci-pci/hp   connected    configured   ok
pcie6                          unknown      disconnected unconfigured unknown
```

3. **停止用于打开设备的应用程序。**

   例如，如果设备是一块以太网卡，请使用 `ipadm` 命令停用相应的接口，并取消检测该接口。例如：

```
# ipadm delete-ip bge1
```

4. **按照如下方式使用 cfgadm 命令手动取消配置设备。或者，如果您有 PCIe 适配卡，也可使用自动配置方法，如按平台指南中定义的方式按插槽的“Attention”（注意）按钮。**

```
# cfgadm -c unconfigure pcie4
```

5. **确认是否已取消配置设备。**

   例如：

```
# cfgadm pci
Ap_Id                          Type         Receptacle   Occupant     Condition
pcie1                          unknown      empty        unconfigured unknown
pcie2                          unknown      empty        unconfigured unknown
pcie3                          unknown      empty        unconfigured unknown
pcie4                          unknown      connected    unconfigured unknown
pcie5                          pci-pci/hp   connected    configured   ok
pcie6                          unknown      disconnected unconfigured unknown
```

注 – 如果取消配置设备，则 `Type` 和 `Condition` 也将成为未知。

6 **手动断开插槽的电源。如果使用自动配置方法，则此步骤是不必要的。有关更多信息，请参阅平台指南。**

```
# cfgadm -c disconnect pcie4
```

7 **确认是否已断开设备连接。**

例如：

```
# cfgadm pci
Ap_Id                          Type         Receptacle   Occupant     Condition
pcie1                          unknown      empty        unconfigured unknown
pcie2                          unknown      empty        unconfigured unknown
pcie3                          unknown      empty        unconfigured unknown
pcie4                          unknown      disconnected unconfigured unknown
pcie5                          pci-pci/hp   connected    configured   ok
pcie6                          unknown      disconnected unconfigured unknown
```

8 **遵照平台指南中的相应说明移除 PCI 适配卡。移除该卡后，插座状态即为空。**

例如：

```
# cfgadm pci
Ap_Id                          Type         Receptacle   Occupant     Condition
pcie1                          unknown      empty        unconfigured unknown
pcie2                          unknown      empty        unconfigured unknown
pcie3                          unknown      empty        unconfigured unknown
pcie4                          unknown      empty        unconfigured unknown
pcie5                          pci-pci/hp   connected    configured   ok
pcie6                          unknown      disconnected unconfigured unknown
```

注 – 可以在引导时启用或禁用自动配置方法，具体取决于平台的实现。请针对环境设置合适的自动配置方法。

▼ 如何添加 PCI 适配卡

以下过程已经更新，可用于添加 PCIe 适配卡。但是，无论使用 PCI 还是 PCIe，添加适配卡的过程都是相同的。

1 **成为管理员。**

2 **标识可热插拔的插槽并打开锁。**

例如，pcie3。

```
# cfgadm pci
Ap_Id                          Type         Receptacle   Occupant     Condition
pcie1                          unknown      empty        unconfigured unknown
pcie2                          unknown      empty        unconfigured unknown
pcie3                          unknown      empty        unconfigured unknown
pcie4                          unknown      empty        unconfigured unknown
pcie5                          pci-pci/hp   connected    configured   ok
pcie6                          unknown      disconnected unconfigured unknown
```

3 遵照平台指南中的相应说明将 PCI 适配卡插入插槽。

4 确定插入 PCI 适配卡后该卡所在的插槽。

例如：

```
# cfgadm pci
Ap_Id                          Type         Receptacle   Occupant     Condition
pcie1                          unknown      empty        unconfigured unknown
pcie2                          unknown      empty        unconfigured unknown
pcie3                          unknown      disconnected unconfigured unknown
pcie4                          unknown      empty        unconfigured unknown
pcie5                          pci-pci/hp   connected    configured   ok
pcie6                          unknown      disconnected unconfigured unknown
```

5 使用 **cfgadm** 命令手动连接插槽的电源。或者，如果您有 PCIe 适配卡，也可使用自动配置方法，如按平台指南中定义的方式按插槽的“Attention”（注意）按钮。

例如：

```
# cfgadm -c connect pcie3
```

6 确认连接点已连接。

例如：

```
# cfgadm pci
Ap_Id                          Type         Receptacle   Occupant     Condition
pcie1                          unknown      empty        unconfigured unknown
pcie2                          unknown      empty        unconfigured unknown
pcie3                          unknown      connected    unconfigured unknown
pcie4                          unknown      empty        unconfigured unknown
pcie5                          pci-pci/hp   connected    configured   ok
pcie6                          unknown      disconnected unconfigured unknown
```

7 按照如下方式使用 **cfgadm** 命令手动配置 PCI 适配卡。如果使用自动配置方法，此步骤应该是不必要的。有关更多信息，请参阅平台指南。

例如：

```
# cfgadm -c configure pcie3
```

8 验证插槽中的 PCI 适配卡的配置。

例如：

```
# cfgadm pci
Ap_Id                          Type         Receptacle   Occupant     Condition
pcie1                          unknown      empty        unconfigured unknown
pcie2                          unknown      empty        unconfigured unknown
pcie3                          etherne/hp   connected    configured unknown
pcie5                          pci-pci/hp   connected    configured   ok
pcie6                          unknown      disconnected unconfigured unknown
```

9 如果此设备为新设备，请配置任何支持的软件。

例如，如果此设备是一块以太网卡，请使用 `ipadm` 命令来设置接口。例如：

```
# ipadm create-addr bge1
```

注 – 可以在引导时启用或禁用自动配置方法，具体取决于平台的实现。请针对环境设置合适的自动配置方法。

对 PCI 配置问题进行故障排除

错误消息

```
cfgadm: Configuration operation invalid: invalid transition
```

原因

尝试进行无效转换。

解决方法

检查是否正确发出了 `cfgadm -c` 命令。使用 `cfgadm` 命令检查当前的插座和插卡状态，并确保 `Ap_Id` 正确。

错误消息

```
cfgadm: Attachment point not found
```

原因

未找到指定的连接点。

解决方法

检查连接点是否正确。使用 `cfgadm` 命令显示可用连接点的列表。另外，请检查物理路径，以查看连接点是否仍然在正确的位置。

使用 cfgadm 命令进行 SATA 热插拔

SATA 控制器和端口倍增器设备的端口由设备树中的连接点表示。系统上连接并配置的 SATA 设备显示为连接点扩展名。术语“连接点”和“SATA 端口”可交换使用。

针对 SATA 设备使用的 `cfgadm` 语法与针对 SCSI 或 PCI 设备的 `cfgadm` 语法稍有不同。

您可以按如下方式显示 SATA 设备信息：

```
% cfgadm -al
Ap_Id                          Type         Receptacle   Occupant     Condition
sata0/0::dsk/c7t0d0            disk         connected    configured   ok
sata0/1::dsk/c7t1d0            disk         connected    configured   ok
sata0/2::dsk/c7t2d0            disk         connected    configured   ok
sata0/3::dsk/c7t3d0            disk         connected    configured   ok
.
.
.
```

▼ 如何取消配置 SATA 设备

一般来说，移除或更换 SATA 设备之前必须取消配置。如果尝试取消配置作为活动 ZFS 存储池一部分的设备，将看到类似以下内容的错误消息：

```
# cfgadm -c unconfigure sata5/7
Unconfigure the device at: /devices/pci@2,0/pci1022...
This operation will suspend activity on the SATA device
Continue (yes/no)? y
cfgadm: Hardware specific failure: Failed to unconfig device at ap_id: /devices/pci@2,0/pci10...
```

1. **成为管理员。**

2. **确定要取消配置的设备。**

   ```
   # cfgadm -al | grep c7t0d0
   sata0/0::dsk/c7t0d0            disk         connected    configured   ok
   ```

3. **取消配置设备。**

   ```
   # cfgadm -c unconfigure sata0/0
   ```

   如果试图通过指定单个设备来取消配置设备，将看到类似以下内容的消息：

   ```
   # cfgadm -c unconfigure sata0/0::dsk/c7t0d0
   do_control_ioctl: open failed: errno:2
   cfgadm: Library error: Cannot open ap_id: /devices/pci@0,0/pci10...
   No such file or directory
   ```

4. **确认该设备已取消配置。**

   ```
   # cfgadm | grep sata0/0
   sata0/0                        disk         connected    unconfigured ok
   ```

▼ 如何配置 SATA 设备

物理移除或替换磁盘后，即可对其进行配置。

1. **成为管理员。**

2. **配置该设备。**

   ```
   # cfgadm -c configure sata0/0
   ```

3. **确认该设备已配置。**

   ```
   # cfgadm | grep sata0/0
   sata0/0::dsk/c7t0d0            disk         connected    configured   ok
   ```

重新配置协调管理器 (Reconfiguration Coordination Manager, RCM) 脚本概述

重新配置协调管理器 (Reconfiguration Coordination Manager, RCM) 是用于管理系统组件动态移除的框架。通过使用 RCM，可以按顺序注册和释放系统资源。

可以使用新的 RCM 脚本功能来编写您自己的脚本，以关闭应用程序，或在动态重新配置过程中从应用程序中完全释放设备。如果重新配置请求影响通过脚本注册的资源，则 RCM 框架将自动启动脚本来响应该请求。

动态移除资源之前，还可以手动从应用程序中释放资源。或者，也可以使用带有 `-f` 选项的 `cfgadm` 命令来强制执行重新配置操作。但是，此选项可能会使应用程序处于未知状态。另外，从应用程序中手动释放资源通常还会导致错误。

RCM 脚本功能可以简化并更好地控制动态重新配置过程。通过创建 RCM 脚本，可以执行以下操作：

- 动态移除设备时，自动释放该设备。如果设备是通过应用程序打开，则此过程还将关闭该设备。
- 从系统中动态移除设备时，运行特定于站点的任务。

什么是 RCM 脚本？

- RCM 守护进程运行的可执行 shell 脚本（Perl、`sh`、`csh` 或 `ksh`）或二进制程序。Perl 是建议使用的语言。
- 通过使用脚本文件所有者的用户 ID 在其自身的地址空间中运行的脚本。
- 使用 `cfgadm` 命令动态重新配置系统资源时，由 RCM 守护进程运行的脚本。

RCM 脚本的功能

动态移除设备时，可以使用 RCM 脚本从应用程序中释放该设备。如果设备当前处于打开状态，则 RCM 脚本还会将其关闭。

例如，磁带备份应用程序的 RCM 脚本可以通知磁带备份应用程序关闭磁带机或磁带备份应用程序。

RCM 脚本进程的工作原理

可以按照以下方式调用 RCM 脚本：

```
$ script-name command [args ...]
```

RCM 脚本可执行以下基本步骤：

1. 通过命令行参数获取 RCM 命令。
2. 执行该命令。
3. 将结果作为名称-值对写入 `stdout` 中。
4. 以适当的退出状态退出。

RCM 守护进程每次运行一个脚本实例。例如，如果某一脚本正在运行，则在该脚本退出之前，RCM 守护进程不会运行该脚本的其他实例。

RCM 脚本命令

必须在 RCM 脚本中包括以下 RCM 命令：

- `scriptinfo`－收集脚本信息
- `register`－注册重要资源
- `resourceinfo`－收集资源信息

可以包括以下某些或所有 RCM 命令：

- `queryremove`－查询是否可以释放资源
- `preremove`－释放资源
- `postremove`－提供资源移除后的通知
- `undoremove`－撤消在 `preremove` 中执行的操作

有关这些 RCM 命令的完整说明，请参见 `rcmscript(4)`。

RCM 脚本处理环境

动态移除设备时，RCM 守护进程将运行以下命令：

- 脚本的 `register` 命令，用于收集脚本中标识的资源（设备名称）的列表。
- 脚本的 `queryremove` 和 `preremov` 命令，在移除资源之前运行，前提是脚本的已注册资源受到动态移除操作的影响。
- 脚本的 `postremove` 命令，前提是移除操作成功。但是，如果移除操作失败，RCM 守护进程会运行该脚本的 `undoremove` 命令。

RCM 脚本任务

以下各节介绍了应用程序开发者和系统管理员的 RCM 脚本任务。

应用程序开发者 RCM 脚本（任务列表）

以下任务列表介绍了创建 RCM 脚本的应用程序开发者的任务。

任务	说明	参考
1. 确定应用程序使用的资源。	确定应用程序使用的可能要动态移除的资源（设备名称）。	cfgadm(1M)
2. 确定用于释放资源的命令。	确定用于通知应用程序从应用程序中正常释放资源的命令。	应用程序文档
3. 确定用于资源移除后进行后续处理的命令。	包括用于通知应用程序资源移除的命令。	rcmscript(4)
4. 确定资源移除失败时使用的命令。	包括用于通知应用程序可用资源的命令。	rcmscript(4)
5. 编写 RCM 脚本。	根据任务 1-4 中标识的信息，编写 RCM 脚本。	第 102 页中的 “磁带备份 RCM 脚本示例”
6. 安装 RCM 脚本。	将脚本添加到相应的脚本目录中。	第 101 页中的 “如何安装 RCM 脚本”
7. 测试 RCM 脚本	通过手动运行脚本命令和启动动态重新配置操作来测试脚本。	第 102 页中的 “如何测试 RCM 脚本”

系统管理员 RCM 脚本（任务列表）

以下任务列表介绍了创建 RCM 脚本进行站点定制的系统管理员的任务。

任务	说明	参考
1. 确定要动态移除的资源。	使用 `cfgadm -l` 命令确定可能要移除的资源（设备名称）。	cfgadm(1M)
2. 确定要停止的应用程序。	确定用于正常停止应用程序的命令。	应用程序文档
3. 确定用于移除资源前后进行各种处理的命令。	确定移除资源前后要执行的操作。	rcmscript(4)
4. 编写 RCM 脚本。	根据任务 1-3 中标识的信息，编写 RCM 脚本。	第 102 页中的 “磁带备份 RCM 脚本示例”
5. 安装 RCM 脚本。	将脚本添加到相应的脚本目录中。	第 101 页中的 “如何安装 RCM 脚本”
6. 测试 RCM 脚本。	通过手动运行脚本命令和启动动态重新配置操作来测试脚本。	第 102 页中的 “如何测试 RCM 脚本”

命名 RCM 脚本

必须按以下形式命名脚本，*vendor,service*，其中：

vendor 提供脚本的供应商的股票代号，或标识供应商的任何独特名称。

service 脚本表示的服务名称。

安装或删除 RCM 脚本

必须是管理员才可以安装或删除 RCM 脚本。使用下表可确定应安装 RCM 脚本的位置。

表 6–1 RCM 脚本目录

目录位置	脚本类型
`/etc/rcm/scripts`	特定系统的脚本
`/usr/platform/‘uname -i‘/lib/rcm/scripts`	特定硬件实现的脚本
`/usr/platform/‘uname -m‘/lib/rcm/scripts`	特定硬件类的脚本
`/usr/lib/rcm/scripts`	任何硬件的脚本

▼ 如何安装 RCM 脚本

1. **成为管理员。**
2. **将脚本复制到相应目录。**

   请参见表 6–1。

   例如：

   ```
   # cp ABC,sample.pl /usr/lib/rcm/scripts
   ```

3. **将脚本的用户 ID 和组 ID 更改为所需的值。**

   ```
   # chown user:group /usr/lib/rcm/scripts/ABC,sample.pl
   ```

4. **将 SIGHUP 发送到 RCM 守护进程。**

   ```
   # pkill -HUP -x -u root rcm_daemon
   ```

▼ 如何删除 RCM 脚本

1. **成为管理员。**
2. **从 RCM 脚本目录中删除脚本。**
   例如：
   ```
   # rm /usr/lib/rcm/scripts/ABC,sample.pl
   ```
3. **将 SIGHUP 发送到 RCM 守护进程。**
   ```
   # pkill -HUP -x -u root rcm_daemon
   ```

▼ 如何测试 RCM 脚本

1. **运行脚本之前，在命令行 shell 中设置环境变量，如 RCM_ENV_FORCE。**
   例如，在 Korn shell 中，请使用以下命令：
   ```
   $ export RCM_ENV_FORCE=TRUE
   ```
2. **通过从命令行手动运行脚本命令来测试脚本。**
   例如：
   ```
   $ script-name scriptinfo
   $ script-name register
   $ script-name preremove resource-name
   $ script-name postremove resource-name
   ```
3. **确保脚本中的每个 RCM 脚本命令都会列显 stdout 的相应输出。**
4. **在相应的脚本目录中安装脚本。**
   有关更多信息，请参见第 101 页中的“如何安装 RCM 脚本”。
5. **通过启动动态删除操作来测试脚本。**
   例如，假定脚本注册了设备 /dev/dsk/c1t0d0s0。请尝试键入以下命令。
   ```
   $ cfgadm -c unconfigure c1::dsk/c1t0d0
   $ cfgadm -f -c unconfigure c1::dsk/c1t0d0
   $ cfgadm -c configure c1::dsk/c1t0d0
   ```

注意 – 请确保您熟悉这些命令，因为这些命令可以改变系统的状态，并会导致系统故障。

磁带备份 RCM 脚本示例

本示例说明如何使用 RCM 脚本执行磁带备份。

磁带备份 RCM 脚本的功能

磁带备份 RCM 脚本可执行以下步骤：

1. 设置 RCM 命令的分发表。
2. 调用与指定的 RCM 命令对应的分发例程，并对于未实现的 RCM 命令以状态 2 退出。
3. 设置 `scriptinfo` 部分。

   ```
   rcm_script_func_info=Tape backup appl script for DR
   ```
4. 通过在 `stdout` 中列显所有磁带机设备名称，在系统中注册所有磁带机。

   ```
   rcm_resource_name=/dev/rmt/$f
   ```

   如果出现错误，则该脚本将在 `stdout` 中列显错误信息。

   ```
   rcm_failure_reason=$errmsg
   ```
5. 为磁带设备设置资源信息。

   ```
   rcm_resource_usage_info=Backup Tape Unit Number $unit
   ```
6. 通过检查备份应用程序是否使用该设备，设置 `preremove` 信息。如果备份应用程序未使用该设备，则动态重新配置操作将继续进行。如果备份应用程序使用该设备，则该脚本将检查 `RCM_ENV_FORCE`。如果将 `RCM_ENV_FORCE` 设置为 `FALSE`，则该脚本将拒绝动态重新配置操作，并列显以下消息：

   ```
   rcm_failure_reason=tape backup in progress pid=...
   ```

   如果将 `RCM_ENV_FORCE` 设置为 `TRUE`，则将停止备份应用程序，重新配置操作则继续进行。

磁带备份重新配置方案的结果

以下是使用 `cfgadm` 命令移除不包含 RCM 脚本的磁带设备时的各种结果。

- 如果使用 `cfgadm` 命令并且备份应用程序未使用磁带设备，则该操作将成功。
- 如果使用 `cfgadm` 命令并且备份应用程序使用磁带设备，则该操作将失败。

以下是使用 `cfgadm` 命令移除包含 RCM 脚本的磁带设备时的各种结果。

- 如果使用 `cfgadm` 命令并且备份应用程序未使用磁带设备，则该操作将成功。
- 如果使用不带 `-f` 选项的 `cfgadm` 命令并且备份应用程序使用磁带设备，则该操作将失败，并会显示以下错误消息：

  ```
  tape backup in progress pid=...
  ```
- 如果使用 `cfgadm -f` 命令并且备份应用程序使用磁带设备，则该脚本会停止备份应用程序，`cfgadm` 操作将成功。

示例－磁带备份 RCM 脚本

```
#! /usr/bin/perl -w
   #
   # A sample site customization RCM script.
   #
   # When RCM_ENV_FORCE is FALSE this script indicates to RCM that it cannot
   # release the tape drive when the tape drive is being used for backup.
   #
   # When RCM_ENV_FORCE is TRUE this script allows DR removing a tape drive
   # when the tape drive is being used for backup by killing the tape
   # backup application.
   #

    use strict;

    my ($cmd, %dispatch);
    $cmd = shift(@ARGV);
# dispatch table for RCM commands
    %dispatch = (
            "scriptinfo"    =>      \&do_scriptinfo,
            "register"      =>      \&do_register,
            "resourceinfo"  =>      \&do_resourceinfo,
            "queryremove"   =>      \&do_preremove,
            "preremove"     =>      \&do_preremove
    );


    if (defined($dispatch{$cmd})) {
            &{$dispatch{$cmd}};
    } else {
            exit (2);
    }

    sub do_scriptinfo
    {
            print "rcm_script_version=1\n";
            print "rcm_script_func_info=Tape backup appl script for DR\n";
            exit (0);
    }

    sub do_register
{
            my ($dir, $f, $errmsg);

            $dir = opendir(RMT, "/dev/rmt");
            if (!$dir) {
                 $errmsg = "Unable to open /dev/rmt directory: $!";
                 print "rcm_failure_reason=$errmsg\n";
                 exit (1);
            }

            while ($f = readdir(RMT)) {
                # ignore hidden files and multiple names for the same device
                if (($f !~ /^\./) && ($f =~ /^[0-9]+$/)) {
                        print "rcm_resource_name=/dev/rmt/$f\n";
                    }

            }
```

```
            closedir(RMT);
            exit (0);
    }
sub do_resourceinfo
    {
      my ($rsrc, $unit);

      $rsrc = shift(@ARGV);
      if ($rsrc =~ /^\/dev\/rmt\/([0-9]+)$/) {
           $unit = $1;
           print "rcm_resource_usage_info=Backup Tape Unit Number $unit\n";
           exit (0);
       } else {
           print "rcm_failure_reason=Unknown tape device!\n";
            exit (1);
        }
    }

    sub do_preremove
    {
            my ($rsrc);

            $rsrc = shift(@ARGV);

            # check if backup application is using this resource
            #if (the backup application is not running on $rsrc) {
                    # allow the DR to continue
            #        exit (0);
            #}
            #
            # If RCM_ENV_FORCE is FALSE deny the operation.
            # If RCM_ENV_FORCE is TRUE kill the backup application in order
            # to allow the DR operation to proceed
            #
            if ($ENV{RCM_ENV_FORCE} eq 'TRUE') {
                 if ($cmd eq 'preremove') {
                         # kill the tape backup application
                 }
                 exit (0);
            } else {
               #
               # indicate that the tape drive can not be released
               # since the device is being used for backup by the
               # tape backup application
               #
               print "rcm_failure_reason=tape backup in progress pid=...\n"
;
               exit (3);
            }
    }
```

◆ ◆ ◆ 第 7 章

7 使用 USB 设备（概述）

本章概述了 Oracle Solaris OS 中的通用串行总线 (Universal Serial Bus, USB) 设备。

以下是本章中概述信息的列表。

- 第 107 页中的 “USB 设备的新增功能”
- 第 110 页中的 “USB 设备概述”
- 第 115 页中的 “关于 Oracle Solaris OS 中的 USB”

有关 USB 设备的最新信息，请访问以下站点：

```
http://www.oracle.com/us/sun/index.html
```

有关 USB 设备的常规信息，请访问以下站点：

```
http://developers.sun.com/solaris/developer/support/driver/usb.html
```

有关在 Oracle Solaris OS 中使用 USB 设备的逐步说明，请参见第 8 章，使用 USB 设备（任务）。

有关动态重新配置和热插拔的常规信息，请参见第 6 章，动态配置设备（任务）。

有关配置 USB 打印机的信息，请参见《Oracle Solaris 管理：常见任务》中的第 15 章 “使用 CUPS 设置和管理打印机（任务）”。

USB 设备的新增功能

以下各节介绍了 Oracle Solaris 发行版中 USB 的新增功能。

- 第 108 页中的 “USB 接口关联描述符支持”
- 第 108 页中的 “EHCI 同步传输支持”
- 第 108 页中的 “USB 设备可热插拔行为”
- 第 109 页中的 “x86: 在基于 GRUB 的引导中支持 USB CD 和 DVD”
- 第 109 页中的 “USB 虚拟键盘和鼠标支持”

USB 接口关联描述符支持

Oracle Solaris 11：USB 设备节点类型，即 IA 节点，是为接口关联描述符 (Interface Association Descriptor, IAD) 支持而创建的。此功能指的是，驱动程序可能会支持同一设备的多个接口，如 Web 摄像头的视频和音频接口。如果未发现 IA 节点的驱动程序，则将结点驱动程序 (`usb_ia`) 绑定到 IA 节点以创建接口节点。有关更多信息，请参见 `usb_ia(7D)`。

EHCI 同步传输支持

Oracle Solaris 11：USB EHCI 主机控制器驱动程序为 USB 2.0 或高速同步设备提供了同步传输支持。有关更多信息，请参见 `usb_isoc_request(9S)`。

USB 设备可热插拔行为

Oracle Solaris 11：此 Solaris 发行版介绍了一种新的设备属性，即**可热插拔**，用于标识可在不重新引导系统的情况下连接的或断开连接的设备以及在无用户介入的情况下自动配置的或取消配置的设备。所有 USB 和 1394 设备均被标识为可热插拔设备，具有第 123 页中的 “使用 USB 海量存储设备”中说明的优点。此外，不可移除介质 USB 和 1394 设备不再被标识为可移除介质设备，并且不再具有**可移除介质**属性。

这些更改主要是在内核级别进行的，可以提高对不可移除介质 USB 和 1394 设备的支持，并改进这些设备的性能。但是，这些更改不影响这些设备的使用。例如，挂载和取消挂载这些设备的工作由 `rmvolmgr` 控制。从用户的角度而言，仅设备的可热插拔和可移除介质属性才是可见的更改。

USB 和 1394 (FireWire) 支持

不可移除的 USB 存储设备和 1394 海量存储设备在驱动程序级别上都被标识为可热插拔的设备。此行为意味着，这些设备可以在不重新引导系统的情况下连接或断开，并且可以无需干预自动配置或取消配置。这些更改是在内核级别进行的，因此不会影响这些设备的使用。例如，挂载和取消挂载这些设备的工作由可移除介质管理服务控制。

此外，还可使用 `format` 实用程序来访问不可移除 USB 设备和 1394 海量存储设备并设置标签。但是，在 `/etc/driver/drv/scsa2usb.conf` 文件中将 `remvalue` 设置为 true 可以覆盖这些设备的新增可热插拔行为。将此参数设置为 true 意味着将该设备视为驱动程序级别的可移除介质设备，前提是该行为是首选行为。

有关使用这些设备的更多信息，请参见 `scsa1394(7D)` 和第 122 页中的 “使用 USB 海量存储设备（任务列表）”。

x86: 在基于 GRUB 的引导中支持 USB CD 和 DVD

Oracle Solaris 11：可以在基于 GRUB 的引导环境中使用以下 USB 功能：

- 从 USB CD 驱动器或 DVD 驱动器安装
- 从 USB 存储设备引导。在从 USB 存储设备引导之前，必须要在 USB 驱动器上安装 Solaris 发行版。

有关引导的更多信息，请参见《在 SPARC 平台上引导和关闭 Oracle Solaris》或《在 x86 平台上引导和关闭 Oracle Solaris》。

USB 虚拟键盘和鼠标支持

Oracle Solaris 11：借助 USB 虚拟键盘和鼠标支持，可以挂接多个键盘和多个鼠标，这种情况下该键盘组或鼠标组的行为如同一个虚拟键盘或鼠标。这意味着每个物理设备的输入都会并入单个输入流中。例如，如果在一个键盘上键入 SHIFT，而在另一个键盘上键入 A，则回显的字符为大写的 A。

另外，还支持将 USB 键盘或鼠标添加到手提电脑中，并将这些设备作为一个整体设备用于手提电脑的 PS/2 键盘和触摸板。

此外，虚拟键盘和鼠标功能还支持条形码读取器。

有关更多信息，请参阅 `virtualkm(7D)`。

Oracle Solaris 对 USB 设备的支持

在基于 SPARC 和 x86 的系统中，均支持以下 USB 1.1 和 USB 2.0 设备，特别注明的除外：

- 音频设备－USB 2.0 集线器仅支持 USB 1.x 设备。
  - 不支持任何 USB 2.0 音频设备。
  - 与连接至 USB 2.0 端口的 USB 2.0 集线器连接的 USB 1.x 音频设备只能在 Oracle Solaris 11 发行版中使用。有关更多信息，请参见 `usb_ac(7D)` 和 `usb_as(7D)`。
  - USB 驱动程序不支持的设备可能包含 `libusb` 应用程序，如 `gphoto2`、`gtkam` 和 `pilotlink`。有关更多信息，请参阅 `/usr/sfw/share/doc/libusb/libusb.txt`。
- 通用 USB 驱动程序 (`ugen(7D)`)
- HID 设备（键盘和鼠标设备，`hid(7D)`）
- 集线器 (`hubd(7D)`)
- 打印机
- 串行设备（Edgeport (`usbser_edge(7D)`)、Prolific (`usbsprl(7D)`)、Keyspan (`usbsksp(7D)`)）

- 存储设备 (scsa2usb(7D))
- 用户空间 USB 设备管理库 (libusb(3LIB))

使用下表可确定特定 USB 1.1 和 USB 2.0 设备的 Solaris 支持信息。

注：

- 仅支持 USB 1.x 音频设备。不支持任何 USB 2.0 音频设备。
- 可使用与连接至 USB 2.0 端口的 USB 2.0 集线器连接的 USB 1.x 音频设备。有关更多信息，请参见 usb_ac(7D) 和 usb_as(7D)。
- USB 驱动程序不支持的设备可能包含 libusb 应用程序，如 gphoto2、gtkam 和 pilotlink。有关更多信息，请参阅 /usr/sfw/share/doc/libusb/libusb.txt。

有关与海量存储设备关联的任务信息，请参见第 8 章，使用 USB 设备（任务）。

有关 ugen 的更多信息，请参见第 113 页中的 “USB 驱动程序增强功能”。

USB 设备概述

通用串行总线 (Universal Serial Bus, USB) 由 PC 行业开发，用于提供将外围设备（如键盘、鼠标设备和打印机）连接到系统的低成本解决方法。

USB 连接器适用于以某种方式连接的唯一一种类型的电缆。USB 的主要设计意图是缓解对不同设备的多种连接器类型的需求。此设计可减少系统后面板上各种连接的混乱。

设备与外部 USB 集线器或计算机本身的根集线器上的 USB 端口连接。由于集线器有多个端口，因此从集线器可以引出多个设备树分支。

有关更多信息，请参见 usba(7D) 或转至以下站点：

http://www.usb.org/home

常用的 USB 首字母缩略词

下表介绍了 Oracle Solaris OS 中使用的 USB 首字母缩略词。有关 USB 组件和对应的首字母缩略词的完整说明，请访问以下站点：

http://www.usb.org/home

首字母缩略词	定义	更多信息
UGEN	USB 通用驱动程序	ugen(7D)

首字母缩略词	定义	更多信息
USB	通用串行总线	usb(7D)
USBA	通用串行总线体系结构 (Solaris)	usba(7D)
USBAI	USBA 客户机驱动程序接口 (Solaris)	N/A
HCD	USB 主机控制器驱动程序	N/A
EHCI	增强型主机控制器接口	ehci(7D)
OHCI	开放主机控制器接口	ohci(7D)
UHCI	通用主机控制器接口	uhci(7D)

USB 总线说明

USB 规范是公用免费规范。该规范定义了总线和连接器的电子和机械接口。

USB 采用由集线器为 USB 设备提供连接点的拓扑结构。主机控制器包含根集线器，该集线器是系统中所有 USB 端口的起点。有关集线器的更多信息，请参见第 118 页中的“USB 主机控制器和集线器”。

图 7–1 USB 物理设备分层结构

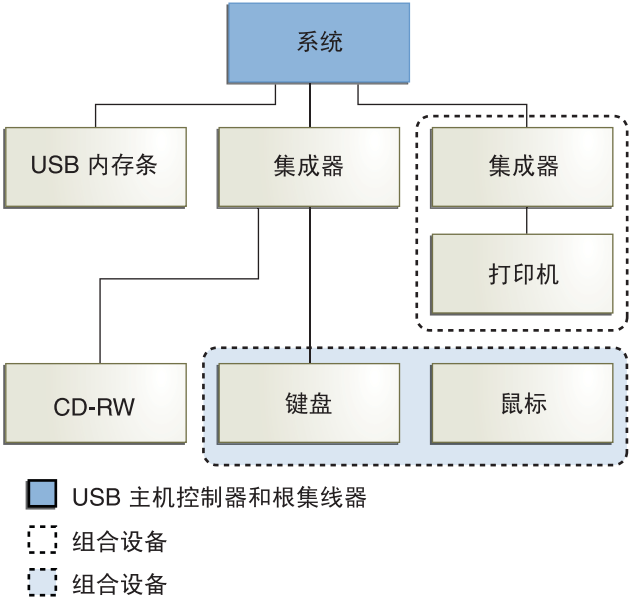


图 7–1 显示了带有三个活动 USB 端口的系统。第一个 USB 端口连接 USB 内存条。第二个 USB 端口连接外部集线器，该集线器又连接 cdrw 设备和键盘/鼠标复合设备。作为**复合设备**，此键盘包含 USB 控制器，该控制器可以操作键盘和连接的鼠标。键盘和鼠标共享公用的 USB 总线地址，因为它们由同一个 USB 控制器定向。

图 7–1 还显示了作为**组合设备**的集线器和打印机的示例。该集线器是与打印机封装在同一个箱子中的外部集线器。打印机与集线器永久连接。集线器和打印机具有各自独立的 USB 总线地址。

以下列出了图 7–1 中显示的一些设备的设备树路径名。

内存条	`/pci@1f,4000/usb@5/storage@1`
键盘	`/pci@1f,4000/usb@5/hub@2/device@1/keyboard@0`
鼠标	`/pci@1f,4000/usb@5/hub@2/device@1/mouse@1`
cdrw 设备	`/pci@1f,4000/usb@5/hub@2/storage@3`
打印机	`/pci@1f,4000/usb@5/hub@3/printer@1`

USB 设备和驱动程序

具有类似属性和服务的 USB 设备组成设备类。每个设备类都有对应的驱动程序。类中的设备由同一个设备驱动程序对进行管理。但是，USB 规范也允许不属于特定类的供应商专有设备。

人工接口设备 (Human Interface Device, HID) 类包含用户控制的设备，如以下设备：

- 键盘
- 鼠标设备
- 游戏控制器

通信设备类包括以下设备：

- 调制解调器
- 以太网适配器

其他设备类包括以下类：

- 音频
- 显示器
- 打印机
- 存储设备

每个 USB 设备都包含反映其所属类的描述符。设备类可指定其成员在配置和数据传输中的行为方式。可以从以下站点获取其他的类信息：

`http://www.usb.org/home`

有关 Oracle Solaris 发行版支持的 USB 设备的更多信息，请参见 usb(7D)。

USB 驱动程序增强功能

包括以下 USB 驱动程序增强功能。

- **USB CDC ACM 设备支持** – acm 驱动程序支持符合 USB 通信类设备规范的抽象控制模型的设备和一些具有调制解调器功能的 PCMCIA 卡。

  pppd 守护进程可通过 /dev/term/[0~9]* 项访问这些设备。有关更多信息，请参见 pppd(1M)。

  有关更多信息，请参见 usbsacm(7D)。

- **通用 USB 驱动程序** – 现在可通过使用标准的 UNIX read(2) 和 write(2) 系统调用的应用程序来访问和处理 UBS 设备，并且无需编写专用内核驱动程序。其他功能包括：
  - 应用程序可访问原始设备数据和设备状态。
  - 驱动程序支持控制、批处理和中断（出入）传输。

  ugen 驱动程序不再需要明确绑定到设备。缺省情况下，usb_mid 将绑定到缺少类驱动程序的设备，并导出与 libusb 兼容的 ugen 接口。例如，可以插入非海量存储设备的 USB 摄像头并可使用 libusb 应用程序来进行访问。此外，scsa2usb 和 usbprn 驱动程序都可导出 ugen 接口，并且可以在这些设备类上直接使用 libusb 应用程序。

  有关更多信息，请参阅 ugen(7D)。

- **USB 串行驱动程序支持**
  - Digi Edgeport USB 支持 – Edgeport USB 驱动程序只能用于 Edgeport 设备，而不能用于其他 USB 串行设备。
    - 新设备可以作为 /dev/term/[0-9]* 和 /dev/cua/[0-9]* 进行访问。
    - USB 串行端口可以像任何其他串行端口一样使用，但不可用作本地串行控制台。用户可以很容易地理解这样一个事实，即数据是通过 USB 端口运行的。

    有关更多信息，请参见 usbser_edge(7D) 或转至以下站点：
    - http://www.digi.com
    - http://www.oracle.com/us/sun/index.html
  - Keyspan – Keyspan USB 串行驱动程序只能用于 Keyspan 设备，该驱动程序当前支持 USA-19HS 和 USA-49WLC 模型。

    有关更多信息，请参见 usbsksp(7D)。
  - Prolific – Prolific USB 串行驱动程序只能用于基于 PL2303 芯片组的设备。

    有关更多信息，请参见 usbsprl(7D)。

  有关 USB 到串行设备支持的更多信息，请转至以下站点：

http://www.oracle.com/us/sun/index.html

- **对用户编写的内核和用户级驱动程序的文档和二进制支持**－有关 USB 驱动程序开发的最新信息，请访问：
  - http://www.oracle.com/us/sun/index.html
  - 《Writing Device Drivers》中的第 20 章 “USB Drivers”
  - 《Writing Device Drivers》中的附录 C “Making a Device Driver 64-Bit Ready”
  - 《Device Driver Tutorial》
  - Intro(7)、Intro(9F) 和 Intro(9S)

EHCI、OHCI 和 UHCI 驱动程序

EHCI 驱动程序的功能包括：

- 符合支持 USB 2.0 的增强型主机控制器接口的标准。
- 支持高速控制、批处理、中断和同步传输。
- USB 2.0 芯片包含一个 EHCI 控制器以及一个或多个 OHCI 或 UHCI 控制器。
- USB 1.1 设备插入后会动态分配给 OHCI 或 UHCI 控制器。USB 2.0 设备插入后会动态分配给 EHCI 控制器。

使用 prtconf 命令输出可确定系统是支持 USB 1.1 设备还是 USB 2.0 设备。例如：

```
# prtconf  -D | egrep "ehci|ohci|uhci"
```

如果 prtconf 输出标识了 EHCI 控制器，则系统支持 USB 2.0 设备。

如果 prtconf 输出标识了 OHCI 或 UHCI 控制器，则系统支持 USB 1.1 设备。

Oracle Solaris USB 体系结构 (USB Architecture, USBA)

USB 设备可以表示为两级设备树节点。设备节点表示整个 USB **设备**。一个或多个子**接口**节点表示设备上的各个 USB 接口。

驱动程序绑定是通过使用兼容的名称属性实现的。有关更多信息，请参阅 IEEE 1275 USB 绑定的 3.2.2.1 节以及《Writing Device Drivers》。驱动程序可以绑定至整个设备并控制所有接口，也可以只绑定至一个接口。如果供应商或类驱动程序不要求绑定至整个设备，则可以将通用 USB 多接口驱动程序绑定至设备级节点。此驱动程序会通过使用 IEEE 1275 绑定规范的 3.3.2.1 节中定义的兼容名称属性，尝试将各驱动程序绑定至每个接口。

Oracle Solaris USB 体系结构 (USB Architecture, USBA) 遵循 USB 1.1 和 USB 2.0 规范，并且是 Oracle Solaris 设备驱动程序接口 (Device Driver Interface, DDI) 的一部分。USBA 模型与 Oracle 公用 SCSI 体系结构 (Oracle Common SCSI Architecture, SCSA) 类似。如下图所示，USBA 是一个薄层，用于为客户机驱动程序提供通用抽象 USB 传输层，进而为这些驱动程序提供实现核心通用 USB 功能的服务。

图 7–2 Oracle Solaris USB 体系结构 (USB Architecture, USBA)

关于 Oracle Solaris OS 中的 USB

本节介绍您应了解的有关 Oracle Solaris OS 中的 USB 的信息。

USB 2.0 功能

包括以下 USB 2.0 功能：

- **更好的性能**－提高了连接到 USB 2.0 控制器的设备的数据吞吐量，速度最快可达 USB 1.1 设备的 40 倍。

  访问高速 USB 设备（如 DVD 和硬盘）时，可以利用高速 USB 协议。
- **向下兼容性**－与 1.0 和 1.1 版本设备和驱动程序兼容，因而可以使用相同的电缆、连接器和软件界面。

有关 USB 设备和术语的说明，请参见第 110 页中的 “USB 设备概述”。

USB 2.0 设备功能和兼容性问题

USB 2.0 设备的定义是遵循 USB 2.0 规范的高速设备。您可以参阅 `http://www.usb.org/home` 中的 USB 2.0 规范。

要确定 USB 设备的速度，请检查 `/var/adm/messages` 文件中类似于以下内容的消息：

```
Dec 13 17:05:57 mysystem usba: [ID 912658 kern.info] USB 2.0 device
(usb50d,249) operating at hi speed (USB 2.x) on USB 2.0 external hub:
storage@4, scsa2usb0 at bus address 4
```

以下是本 Oracle Solaris 发行版中支持的一些 USB 设备：

- 海量存储设备，如 CD-RW、硬盘、DVD、数码相机、磁盘、磁带机、内存条以及多格式读卡器
- 键盘和鼠标设备
- 音频设备，如扬声器和麦克风

有关 Oracle Solaris 发行版中已验证的 USB 设备的完整列表，请访问：

`http://www.oracle.com/us/sun/index.html`

通过修改 `scsa2usb.conf` 文件，可以使用其他存储设备。有关更多信息，请参见 `scsa2usb(7D)`。

Oracle Solaris USB 2.0 设备支持包括以下功能：

- USB 总线速度从 12 MB/秒提高到 480 MB/秒。这种提高意味着如果将支持 USB 2.0 规范的设备连接到 USB 2.0 端口，其运行速度将大大高于支持 USB 1.1 规范的同类设备。

  USB 2.0 端口可能是以下端口之一：

  - USB 2.0 PCI 卡上的端口
  - 连接到 USB 2.0 端口的 USB 2.0 集线器上的端口
  - SPARC 或 x86 计算机主板上的端口

  较旧的 SPARC 平台可能需要 USB 2.0 PCI 卡。
- 有关已针对 Oracle Solaris 发行版验证过的 USB 2.0 PCI 卡的列表，请访问：

  `http://www.oracle.com/us/sun/index.html`
- USB 1.1 设备的工作方式与过去一样，即使同一系统中同时有 USB 1.1 和 USB 2.0 设备也是如此。
- 尽管 USB 2.0 设备能够在 USB 1.x 端口上运行，但如果将其连接到 USB 2.0 端口，其性能会大大提高。
- USB 2.0 主机控制器具有一个高速的增强型主机控制器接口 (Enhanced Host Controller Interface, EHCI) 以及一个或多个内嵌了 OpenHCI 主机控制器接口 (OpenHCI Host Controller Interface, OHCI) 或通用主机控制器接口 (Universal Host Controller Interface, UHCI) 的控制器。连接到 USB 2.0 端口的设备会被动态分配给 EHCI 或 OHCI 控制器，具体取决于它们是否支持 USB 2.0。

注 – 对于连接到 USB 2.0 PCI 卡上某一端口的 USB 2.0 存储设备，以及在同一硬件配置中用于早期 Solaris 发行版的 USB 2.0 存储设备，在升级到本发行版后，可以更改设备名称。进行这种更改是由于上述设备现在被视为 USB 2.0 设备，并且由 EHCI 控制器接管。上述设备的控制器编号（`/dev/[r]dsk/c`*w*`t`*x*`d`*y*`s`*z* 中的 *w*）将被更改。

另请注意，USB 设备的速度限制为父端口可支持的速度。例如，如果 USB 2.0 外部集线器的下游设备依次是 USB 1.x 集线器和 USB 2.0 设备，则与 USB 2.0 外部集线器连接的设备将以全速而非高速运行。

有关 USB 2.0 设备支持的更多信息，请参见 `ehci(7D)` 和 `usba(7D)`。

总线供电设备

总线供电的集线器使用来自与之连接的 USB 总线的电力，为与其连接的设备供电。必须特别小心不要使这些集线器过载，因为这些集线器为其下游设备提供的电力有限。

对 USB 设备进行电能预算。此功能具有以下限制：

- 禁止层叠两个总线供电的集线器。
- 每个总线供电的集线器允许每个端口的最大电流仅为 100 mA。
- 仅允许将自备电源或低总线供电的设备连接至总线供电的集线器。不允许连接高总线供电的设备。某些集线器或设备会报告虚假的电源来源，因此该种连接可能是不可预测的。

USB 键盘和鼠标设备

使用 USB 键盘和鼠标设备时，请牢记以下问题：

- 在重新引导**期间**或在 SPARC 系统中出现 `ok` 提示符时，请勿移动键盘和鼠标。系统重新引导**后**，可以随时将键盘和鼠标移动到另一个集线器。插入键盘和鼠标后，它们便会再次完全正常运行。
- 在某些第三方 USB 键盘上，紧靠小键盘左侧的键可能不起作用。
- **SPARC** – 在 SPARC 系统中使用 USB 键盘和鼠标设备时，请牢记以下问题：
  - USB 键盘上的电源键与类型 5 键盘上的电源键作用不同。在 USB 键盘上，可以使用 SUSPEND/SHUTDOWN 键暂停或关闭系统。但是，不能使用该键来打开系统电源。
  - 引导过程完成之前，OpenBoot PROM (OBP) 限制键盘和鼠标设备只能与主板根集线器端口连接。
  - USB 键盘和鼠标设备不能与类型 3、4 或 5 键盘同时用于传统 SPARC 系统。
- 有关多键盘和鼠标设备支持的信息，请参见 `virtualkm(7D)`。

USB 滚轮鼠标支持

支持以下滚轮鼠标功能：

- 支持超过 3 个按键的 USB 或 PS/2 鼠标设备。
- 支持 USB 或 PS/2 鼠标设备的滚轮鼠标滚动功能。此项支持表明滚动 USB 或 PS/2 鼠标的滚轮会在鼠标焦点所在的应用程序或窗口中产生滚动效果。StarOffice、Firefox 和 GNOME 应用程序均支持滚轮鼠标的滚动功能。但其他应用程序可能不支持滚轮鼠标的滚动功能。

USB 主机控制器和集线器

USB 集线器负责以下操作：

- 监视在其端口上插入或移除设备
- 对其端口上的各个设备进行电源管理
- 控制传输到其端口的电源

USB 主机控制器具有一个嵌入式集线器，称为**根集线器**。系统的后面板上显示的端口是根集线器的端口。USB 主机控制器负责以下操作：

- 定向 USB 总线。各个设备都不能任意定向总线。
- 使用该设备确定的轮询间隔轮询设备。为说明轮询间隔的时间，假定设备具有足够缓冲。
- 在 USB 主机控制器和与之连接的设备之间发送数据。不支持对等通信。

USB 集线器设备

- 请勿在基于 SPARC 的系统或基于 x86 的系统上将集线器层叠超过四级。在 SPARC 系统中，OpenBoot PROM 不能可靠地探测超过四级的设备。
- 请勿以层叠样式将一个总线供电的集线器插入另一个总线供电的集线器中。总线供电的集线器没有自己的电源。
- 请勿将需要大功率电源的设备连接至总线供电的集线器。这些设备可能会被拒绝与总线供电的集线器连接，或者可能会消耗其他设备的集线器的电源。这类设备的示例为 USB 磁盘设备。

SPARC: USB 电源管理

SPARC 系统完全支持暂停和恢复 USB 设备。但是，请勿暂停繁忙的设备，并且请勿在暂停关机的情况下关闭系统电源时移除设备。

USB 框架可通过启用电源管理，尽可能地对基于 SPARC 的系统上的所有设备进行电源管理。对 USB 设备进行电源管理意味着集线器驱动程序将暂停与设备连接的端口。支

持**远程唤醒**的设备可以通知系统唤醒设备路径中的所有内容，以便可以使用该设备。如果应用程序将 I/O 发送到该设备，则主机系统也可以唤醒设备。

缺省情况下，会对所有的 HID 设备（键盘、鼠标、集线器和存储设备）、集线器设备和存储设备进行电源管理，前提是这些设备支持远程唤醒功能。仅在两个打印作业之间对 USB 打印机进行电源管理。通用 USB 驱动程序 (UGEN) 管理的设备仅在关闭时才会进行电源管理。

实施电源管理以减少能耗时，首先会关闭 USB 叶设备的电源。关闭与集线器端口连接的所有设备的电源后，集线器电源会在一定延迟后关闭。为实现最有效的电源管理，请勿层叠多个集线器。

有关在 SPARC 系统中使用 SUSPEND/SHUTDOWN 键的信息，请参见第 117 页中的 “USB 键盘和鼠标设备”。

USB 电缆原则

连接 USB 电缆时，请牢记以下原则：

- 始终使用符合 USB 2.0 并且以全速 (480 Mbit/sec) 运行的 20/28 AWG 电缆来连接 USB 2.0 设备。
- 支持的最大电缆长度为 5 米。
- 请勿使用电缆延伸架。为获得最佳效果，请使用自备电源的集线器来延伸电缆长度。

有关更多信息，请访问：

`http://www.usb.org/about/faq/ans5`

◆ ◆ ◆ 第 8 章

使用 USB 设备（任务）

本章提供有关在 Oracle Solaris OS 中使用 USB 设备的逐步说明。本章特别涉及如何在 Oracle Solaris 11发行版中使用带有 HAL 服务的 USB 设备。

有关与使用 USB 设备相关的过程信息，以下主题：

- 第 121 页中的 “在 Oracle Solaris OS 中管理 USB 设备（任务列表）”
- 第 122 页中的 “使用 USB 海量存储设备（任务列表）”
- 第 139 页中的 “使用 USB 音频设备（任务列表）”
- 第 143 页中的 “使用 `cfgadm` 命令热插拔 USB 设备（任务列表）”

有关 USB 设备的最新信息，请访问以下站点：

`http://www.oracle.com/us/sun/index.html`

有关使用 USB 设备的概述信息，请参见第 7 章，使用 USB 设备（概述）。

在 Oracle Solaris OS 中管理 USB 设备（任务列表）

使用本指南可以确定在 Oracle Solaris OS 中管理 USB 设备的所有任务。每个任务都指向一系列其他任务，如使用 USB 设备、热插拔 USB 设备以及添加 USB 音频设备。

有关在 Oracle Solaris OS 中使用 USB 组件的信息，请参见第 115 页中的 “关于 Oracle Solaris OS 中的 USB”。

任务	说明	参考
使用 USB 海量存储设备。	USB 海量存储设备必须包含文件系统，才能向其中添加数据。此外，必须先格式化 USB 磁盘，然后才能在这些磁盘上创建和挂载文件系统。 本节还介绍如何以物理方式在系统中添加或移除 USB 设备。	第 122 页中的“使用 USB 海量存储设备（任务列表）”
添加 USB 音频设备。	使用此任务列表可以确定与添加 USB 音频设备关联的任务。	第 139 页中的“使用 USB 音频设备（任务列表）”
使用 `cfgadm` 命令在系统中添加或移除 USB 设备。	可以使用 `cfgadm` 命令以逻辑方式在系统中添加或移除 USB 设备。	第 143 页中的“使用 `cfgadm` 命令热插拔 USB 设备（任务列表）”

使用 USB 海量存储设备（任务列表）

任务	说明	参考
添加或移除 USB 海量存储设备。	选择以下操作之一可添加 USB 海量存储设备： 添加 USB 海量存储设备。 添加用于访问数字图像的 USB 相机。 移除 USB 海量存储设备。	 第 126 页中的“如何添加 USB 海量存储设备” 第 127 页中的“如何添加 USB 相机” 第 128 页中的“如何移除 USB 海量存储设备”
添加非符合的 USB 海量存储设备。	通过向 `scsa2usb.conf` 添加项，可以添加非符合的 USB 海量存储设备。	第 125 页中的“如何使用非符合 USB 海量存储设备”
准备使用 USB 海量存储设备。 显示 USB 设备信息。	准备使用 USB 海量存储设备。 显示有关 USB 设备的信息。	第 128 页中的“准备使用 USB 海量存储设备” 第 129 页中的“如何显示 USB 设备信息”
在 USB 海量存储设备中创建文件系统。	必须先在设备中创建文件系统，然后才能向其中添加数据。	第 130 页中的“如何在 USB 海量存储设备中创建文件系统”
在 USB 海量存储设备中修改分区并创建文件系统。	在 USB 海量存储设备中创建文件系统之前，可能需要修改现有的分区。	第 132 页中的“如何在 USB 海量存储设备中修改分区和创建 PCFS 文件系统”

任务	说明	参考
在 USB 海量存储设备中创建 Solaris 分区并修改相应的分片。	使用 `rmformat` 修改分片信息之前，需要使用分片信息创建一个文件。	第 135 页中的 “如何在 USB 海量存储设备中创建 Solaris 分区和修改分片”
挂载 USB 海量存储设备。	挂载 USB 海量存储设备。	第 136 页中的 “如何挂载或取消挂载 USB 海量存储设备”
（可选的）禁用 USB 设备驱动程序。	如果不需要系统支持 USB，可以禁用 USB 设备驱动程序。	第 139 页中的 “如何禁用特定的 USB 驱动程序”
（可选的）移除未使用的 USB 设备链接。	使用 `devfsadm` 命令移除 USB 设备链接。	第 139 页中的 “如何移除未使用的 USB 设备链接”

使用 USB 海量存储设备

支持以下 USB 可移除海量存储设备：

- CD-RW
- 硬盘
- DVD
- 数字相机
- 磁盘设备
- SmartMedia 和 CompactFlash 设备

有关将 USB 海量存储设备用于 Oracle Solaris ZFS 文件系统的信息，请参见第 107 页中的 “USB 设备的新增功能”

有关 Oracle Solaris OS 所支持的 USB 设备的完整列表，请参见：

`http://www.oracle.com/us/sun/index.html`

在以前的 Solaris 发行版中，所有的 USB 存储设备都被标识为可移除介质设备，这些设备可提供以下许多优点，包括自动挂载。在 Oracle Solaris 11 发行版中，USB 海量存储设备被标识为可热插拔设备，而且具有 USB 可移除设备的优点（如下所述）。有关可热插拔行为的更多信息，请参见第 108 页中的 “USB 和 1394 (FireWire) 支持”。

- 自动挂载可热插拔设备。
- 支持使用标准 MS-DOS 或 Windows (FAT) 文件系统的 USB 存储设备。
- 可以使用用户友好的 `rmformat` 命令来创建分片。还可以使用 `fdisk` 命令对 USB 设备进行分区，但决不能使用 `format` 实用程序或 `rmformat -F` 命令以物理方式格式化 USB 驱动器。
- 可使用 `rmformat` 命令显示插入了介质的所有 USB 设备。有关示例，请参见第 129 页中的 “如何显示 USB 设备信息”。
- 由于不再需要 `mount` 命令，因此非超级用户现在可以访问 USB 存储设备。设备自动挂载并且位于 `/media` 目录下。

- 可以通过可移除介质服务来管理这些设备，也可以不通过可移除介质服务。
- 可以挂载和访问使用 FAT 文件系统的磁盘。例如：

```
mount -F pcfs /dev/dsk/c2t0d0s0:c /mnt
```

- 现在，除了支持 LOG SENSE 页面的 USB 存储设备之外，其他所有 USB 存储设备都会进行电源管理。包含 LOG SENSE 页面的设备通常是通过 USB-to-SCSI 桥接设备进行连接的 SCSI 驱动器。
- 使用 USB 海量存储设备时，应用程序的工作方式可能有所不同。如果要将应用程序用于 USB 存储设备，请牢记以下问题：
  - 应用程序可能对介质的存储容量作出不正确的假设，因为以前仅有较小存储容量的设备（如磁盘）才是可移除设备。
  - 当应用程序发出请求，要求设备弹出介质，而此移除操作却不适用于该设备（例如硬盘驱动器）时，请求会成功，但不会产生任何结果。
  - 如果要使用以前的 Solaris 发行版中的性能，即将所有的 USB 海量存储设备都视为可移除介质设备，则可通过更新 /etc/driver/drv/scsa2usb.conf 文件来强制使用旧性能。

有关使用 USB 海量存储设备的更多信息，请参见 scsa2usb(7D)。

使用 USB 磁盘设备

USB 磁盘设备显示为可移除介质设备。USB 磁盘设备不是通过 fd（磁盘）驱动程序进行管理的。发出用于 fd（本机磁盘）驱动程序的 ioctl(2) 调用的应用程序将失败。仅发出 read(2) 和 write(2) 调用的应用程序将成功。其他应用程序（如 SunPCI 和 rmformat）也将成功。

将 USB 磁盘设备视为 SCSI 可移除介质设备。可在 /media 目录下访问该设备。

有关如何使用 USB 磁盘设备的更多信息，请参见第 1 章，管理可移除介质（概述）。

使用非符合 USB 海量存储设备

即使某些设备未将其自身标识为符合 USB 海量存储类，或以不正确的方式标识其自身，USB 海量存储驱动程序也仍可能支持这些设备。scsa2usb.conf 文件包含一个属性覆盖列表，其中列有供应商 ID、产品 ID、用于匹配海量存储设备的修订版，以及用于覆盖缺省设备属性的字段。缺省情况下，会注释掉此列表中的项。要启用对特定设备的支持，可以复制并取消注释这些项。

如果将 USB 海量存储设备连接至运行此 Solaris 发行版的系统，但该系统无法使用它，则可以检查 /kernel/drv/scsa2usb.conf 文件，查看是否存在与该设备对应的注释掉的项。根据 scsa2usb.conf 文件中给出的信息，查看是否可以使用覆盖信息支持特定设备。

有关推荐的 USB 海量存储设备的列表，请转至：

http://www.oracle.com/us/sun/index.html

有关更多信息，请参见 scsa2usb(7D)。

▼ 如何使用非符合 USB 海量存储设备

1. **成为管理员。**
2. **向 /etc/driver/drv/scsa2usb.conf 文件中添加项。**
   以下项适用于 USB 内存条。

   ```
   attribute-override-list = "vid=* reduced-cmd-support=true";
   ```

3. **重新引导系统或执行以下操作：**

   a. **移除该设备。**

   b. **手动更新驱动程序。**

      ```
      # update_drv -f scsa2usb
      ```

   c. **添加该设备。**

热插拔 USB 海量存储设备

热插拔设备是指在未关闭操作系统或未关闭系统电源的情况下添加或移除设备。所有的 USB 设备都是可热插拔设备。

可移除介质管理器现在可以识别热插拔设备。一旦插入该设备，便会在几秒钟内对其进行挂载。如果未发生任何情况，请查看该设备是否已挂载。

确保可移除介质服务正在运行。

```
# svcs hal dbus rmvolmgr
STATE          STIME    FMRI
online         May_03   svc:/system/dbus:default
online         May_03   svc:/system/hal:default
online         May_03   svc:/system/filesystem/rmvolmgr:default
```

如果文件系统有效并且可以进行识别，则可以从该设备将其挂载。

如果设备上的文件系统无法进行自动挂载，请尝试手动挂载。

热移除设备之前，应在 `eject -l` 命令的别名中找到该设备的名称。然后，弹出该设备的介质。如果不执行此操作，该设备会被释放，并且端口可再次使用，但是该设备中的文件系统可能已被破坏。

热插拔 USB 设备时，系统的设备分层结构中会立即显示该设备，如 prtconf 命令输出所示。移除 USB 设备时，如果未使用该设备，则会从系统的设备分层结构中移除该设备。

如果拔除设备时正在使用该设备，则该设备节点仍然存在，但控制该设备的驱动程序将停止该设备中的所有活动。向此设备发出的任何新的 I/O 活动都会返回错误。

在这种情况下，系统会提示插入原始设备。如果该设备不再可用，请停止应用程序。几秒钟后，端口将再次可用。

注 – 移除活动的设备或打开的设备可能会损害数据完整性。移除设备之前应始终先将其关闭，但连接的键盘和鼠标除外（可在活动状态时将其移除）。

▼ 如何添加 USB 海量存储设备

1. **成为管理员。**
2. **连接 USB 海量存储设备。**
3. **验证是否已添加 USB 设备。**

   例如：

```
$ rmformat
 Looking for devices...
       1. Logical Node: /dev/rdsk/c3t0d0p0
          Physical Node: /pci@0,0/pci108e,534a@2,1/storage@3/disk@0,0
          Connected Device: SanDisk  Cruzer Micro     0.3
          Device Type: Removable
          Bus: USB
          Size: 245.0 MB
          Label: <None>
          Access permissions: Medium is not write protected.
```

4. **验证设备是否在 /media 目录下自动挂载。**

   例如：

```
$ ls /media/NONAME
aa  bb
```

   还可以使用 rmmount -l 命令列出可挂载设备的路径和别名。如果设备已挂载在 /media 目录下，将看到与以下内容类似的输出：

```
$ rmmount -l
/dev/dsk/c3t0d0p0:1  rmdisk0,NONAME,/media/NONAME
```

▼ 如何添加 USB 相机

如果相机的介质使用 PCFS 文件系统，其将进行自动挂载。如果该设备未绑定至 scsa2usb 驱动程序，请使用 libusb 应用程序来传送图片。有关更多信息，请参阅 /usr/sfw/share/doc/libusb/libusb.txt。

1 **成为管理员。**

2 **插入并打开 USB 相机。**

系统将为相机创建一个逻辑设备。插入相机后，会将输出写入 /var/adm/messages 文件，以确认设备的连接。系统会将相机视为存储设备。

3 **检查写入 /var/adm/messages 文件的输出。**

```
# more /var/adm/messages
```

通过检查此输出，可以确定已创建的逻辑设备，这样便可以使用该设备来访问图像。其输出与以下内容类似：

```
Jul 15 09:53:35 buffy usba: [ID 349649 kern.info]    OLYMPUS, C-3040ZOOM,
 000153719068
Jul 15 09:53:35 buffy genunix: [ID 936769 kern.info] scsa2usb1 is
/pci@0,0/pci925,1234@7,2/storage@2
Jul 15 09:53:36 buffy scsi: [ID 193665 kern.info] sd3 at scsa2usb1:
target 0 lun 0
```

请通过执行以下操作，使该设备与可挂载的 /dev/dsk 链路项匹配：

```
# ls -l /dev/dsk/c*0 | grep /pci@0,0/pci925,1234@7,2/storage@2
 lrwxrwxrwx   1 root     root          58 Jun 14  2010 c3t0d0p0 ->
 ../../devices/pci@0,0/pci925,1234@7,2/storage@2/disk@0,0:a
```

4 **挂载 USB 相机文件系统。**

该相机的文件系统很可能是 PCFS 文件系统。如果文件系统是 PCFS，则该文件系统应进行自动挂载。

要在 x86 系统上手动挂载文件系统，可以使用类似以下内容的语法：

```
# mount -F pcfs /dev/dsk/c3t0d0p0:c /mnt
```

要在 SPARC 系统上手动挂载文件系统，可以使用类似以下内容的语法：

```
# mount -F pcfs /dev/dsk/c3t0d0s0:c /mnt
```

有关取消挂载文件系统的信息，请参见第 306 页中的 “挂载和取消挂载 Oracle Solaris 文件系统”。

有关挂载不同 PCFS 文件系统的信息，请参见 mount_pcfs(1M)。

5 **验证图像文件是否可用。**

例如：

```
# ls /mnt/DCIM/100OLYMP/
P7220001.JPG*  P7220003.JPG*  P7220005.JPG*
P7220002.JPG*  P7220004.JPG*  P7220006.JPG*
```

6 **查看并处理 USB 相机所创建的图像文件。**

例如：

```
# /usr/dt/bin/sdtimage P7220001.JPG &
```

7 **断开相机连接之前，取消挂载文件系统。**

例如：

```
# umount /mnt
```

8 **可选关闭并断开相机连接。**

▼ 如何移除 USB 海量存储设备

1 **成为管理员。**

2 **停止使用该设备的所有活动应用程序。**

3 **取消挂载该设备。**

```
$ rmumount NONAME
```

或者，以管理员身份使用 `umount` 命令。例如：

```
# umount /media/NONAME
```

有关取消挂载 USB 设备的更多信息，请参见第 136 页中的 “如何挂载或取消挂载 USB 海量存储设备”。

4 **移除该设备。**

准备使用 USB 海量存储设备

无论是否使用可移除介质服务，都可以访问有关可移除介质的信息。有关使用 GNOME 的文件管理器访问可移除介质相关信息的信息，请参见 GNOME 桌面文档。

格式化 USB 设备后，通常会将该设备挂载于 `/media/` *label* 目录下。

对于字符设备，对应的设备节点是在 `/dev/rdsk` 目录下创建的，而对于块设备，对应的设备节点是在 `/dev/dsk` 目录下创建的。设备链接是在热插拔设备时创建的。有关更多信息，请参见 `scsa2usb(7D)`。

如果无法通过可移除介质服务标识设备，请尝试手动挂载设备。首先，使用 rmformat 命令确定设备路径，然后使用 mount 命令以管理员身份手动挂载设备。

如果通过可移除介质服务挂载了设备，可以通过使用 rmumount 命令来取消挂载设备。如果手动挂载了设备，可以使用 umount 命令以管理员身份取消挂载设备。

有关挂载和取消挂载 USB 设备的更多信息，请参见第 136 页中的 “如何挂载或取消挂载 USB 海量存储设备”。

▼ 如何显示 USB 设备信息

- **显示有关 USB 设备的信息。**

  例如，使用 prtconf 命令可显示 USB 设备信息。本示例中的 prtconf 输出已截断为仅显示 USB 设备信息。

```
$ prtconf
        usb, instance #0
                hub, instance #2
                    device, instance #8
                        interface (driver not attached)
                    printer (driver not attached)
                    mouse, instance #14
                    device, instance #9
                        keyboard, instance #15
                        mouse, instance #16
                    storage, instance #7
                        disk (driver not attached)
                    communications, instance #10
                        modem (driver not attached)
                        data (driver not attached)
                storage, instance #0
                    disk (driver not attached)
                storage, instance #1
                    disk (driver not attached)
```

  例如，使用 rmformat 命令可显示 USB 设备信息。

```
$ rmformat
 Looking for devices...
       1. Logical Node: /dev/rdsk/c3t0d0p0
          Physical Node: /pci@0,0/pci108e,534a@2,1/storage@3/disk@0,0
          Connected Device: SanDisk  Cruzer Micro     0.3
          Device Type: Removable
          Bus: USB
          Size: 245.0 MB
          Label: <None>
          Access permissions: Medium is not write protected.
```

▼ 如何在 USB 海量存储设备中创建文件系统

必须先格式化 USB 磁盘，然后才能向其中添加文件系统。所有其他的 USB 海量存储设备只需一个文件系统即可使用。格式化 USB 设备时，请牢记以下要点：

- 除在 USB 磁盘上以外，请勿使用 `rmformat -F`。
- 如果缺省的分片不可接受，请使用 `rmformat -s` 命令来创建分片。使用 `fdisk` 实用程序对 USB 设备进行分区（如果需要）。有关逐步说明，请参见：
  - 第 132 页中的 “如何在 USB 海量存储设备中修改分区和创建 PCFS 文件系统”
  - 第 135 页中的 “如何在 USB 海量存储设备中创建 Solaris 分区和修改分片”
  - 如果自动挂载了 USB 设备，必须先取消挂载设备，才能在 USB 设备上创建文件系统。使用 `rmmount -l` 命令确定设备别名，然后使用 `rmumount` 命令取消挂载 USB 设备。

注 – 如果需要格式化 USB 磁盘，请仅执行步骤 4-5。

---

1 成为管理员。

2 向系统中添加 USB 设备。有关对 USB 设备进行热插拔的信息，请参见：

- 第 125 页中的 “热插拔 USB 海量存储设备”
- 第 143 页中的 “使用 `cfgadm` 命令热插拔 USB 设备（任务列表）”

3 （可选的）标识 USB 设备。

例如：

```
# rmformat
Looking for devices...
     1. Logical Node: /dev/rdsk/c2t0d0p0
        Physical Node: /pci@0,0/pci108e,534a@2,1/hub@7/floppy@1/disk@0,0
        Connected Device: MITSUMI  USB FDD          1039
        Device Type: Floppy drive
        Bus: USB
        Size: 1.4 MB
        Label: <None>
        Access permissions: Medium is not write protected.
```

在本示例中，磁盘设备为 `c2t0d0p0`。

4 将磁盘插入磁盘驱动器中（如有必要）。

5 格式化该磁盘（如有必要）。

```
% rmformat -F long raw-device
```

例如，在 SPARC 系统上：

```
% rmformat -F long /dev/rdsk/c2t0d0s2
```

例如，在 x86 系统上：

```
% rmformat -F long /dev/rdsk/c3t0d0p0
```

6 **确定文件系统类型并确保取消挂载设备。然后，选择以下操作之一：**

有关取消挂载 USB 设备的更多信息，请参见第 136 页中的 “如何挂载或取消挂载 USB 海量存储设备”。

- 创建 PCFS 文件系统。

  ```
  # mkfs -F pcfs -o nofdisk,size=size raw-device
  ```

  以 512 字节块为单位指定 `-size` 选项。

  以下示例说明如何在 SPARC 系统中的 1.4 MB 磁盘上创建 PCFS 文件系统：

  ```
  # mkfs -F pcfs /dev/rdsk/c2t0d0p0
  Construct a new FAT file system on /dev/rdsk/c2t0d0p0: (y/n)? y
  ```

  以下示例说明如何在 x86 系统中的 1.4 MB 磁盘上创建 PCFS 文件系统：

  ```
  # mkfs -F pcfs /dev/rdsk/c2t0d0s2
  Construct a new FAT file system on /dev/rdsk/c2t0d0s2: (y/n)? y
  ```

  以下示例说明如何在 SPARC 系统中的 100 MB USB 内存条上创建 PCFS 文件系统：

  ```
  # mkfs -F pcfs /dev/rdsk/c5t0d0s2:c
  ```

  以下示例说明如何在 x86 系统中的 100 MB USB 内存条上创建 PCFS 文件系统：

  ```
  # mkfs -F pcfs /dev/rdsk/c5t0d0p0:c
  ```

  完成此命令可能要用几分钟时间。

- 创建传统 UFS 文件系统。

  ```
  # newfs raw-device
  ```

  例如：

  ```
  # newfs /dev/rdsk/c4t0d0s7
  ```

  对于大容量 USB 硬盘，请考虑使用 `newfs -f 4096` 选项或 `newfs -T` 选项。

注 – 由于磁盘的存储容量有限，因此 UFS 文件系统开销会占用磁盘上的大部分空间。

有关在 USB 海量存储设备中创建 PCFS 文件系统和修改分片的详细示例，请参见以下两个过程。

▼ 如何在 USB 海量存储设备中修改分区和创建 PCFS 文件系统

以下步骤介绍如何删除现有分区、创建新分区，然后在 USB 设备中创建 PCFS 文件系统。执行此任务之前，请确保先备份所有数据。

1. **成为管理员。**

2. **启动 `fdisk` 实用程序。**

   例如：

   ```
   # fdisk /dev/rdsk/c3t0d0p0
   ```

3. **通过选择选项 3 来删除分区。**

   例如：

   ```
         Total disk size is 29 cylinders
                Cylinder size is 2048 (512 byte) blocks

                                                Cylinders
         Partition   Status    Type          Start   End   Length    %
         =========   ======    ============  =====   ===   ======   ===
             1       Active    Solaris2          1    28      28     97

   SELECT ONE OF THE FOLLOWING:
      1. Create a partition
      2. Specify the active partition
      3. Delete a partition
      4. Change between Solaris and Solaris2 Partition IDs
      5. Exit (update disk configuration and exit)
      6. Cancel (exit without updating disk configuration)
   Enter Selection: 3
   ```

4. **选择要删除的分区号。**

   例如：

   ```
         Total disk size is 29 cylinders
                Cylinder size is 2048 (512 byte) blocks

                                                Cylinders
         Partition   Status    Type          Start   End   Length    %
         =========   ======    ============  =====   ===   ======   ===
             1       Active    Solaris2          1    28      28     97

   SELECT ONE OF THE FOLLOWING:
      1. Create a partition
      2. Specify the active partition
      3. Delete a partition
      4. Change between Solaris and Solaris2 Partition IDs
      5. Exit (update disk configuration and exit)
      6. Cancel (exit without updating disk configuration)
   Specify the partition number to delete (or enter 0 to exit): 1

   Partition deleted.
   ```

5 创建分区。

例如：

```
      Total disk size is 29 cylinders
             Cylinder size is 2048 (512 byte) blocks

                                                 Cylinders
      Partition   Status    Type          Start   End   Length    %
      =========   ======    ============  =====   ===   ======   ===

WARNING: no partitions are defined!

SELECT ONE OF THE FOLLOWING:
   1. Create a partition
   2. Specify the active partition
   3. Delete a partition
   4. Change between Solaris and Solaris2 Partition IDs
   5. Exit (update disk configuration and exit)
   6. Cancel (exit without updating disk configuration)
Enter Selection: 1
```

6 选择 **FAT32** 分区类型。

```
      Total disk size is 29 cylinders
             Cylinder size is 2048 (512 byte) blocks

                                                 Cylinders
      Partition   Status    Type          Start   End   Length    %
      =========   ======    ============  =====   ===   ======   ===

WARNING: no partitions are defined!

SELECT ONE OF THE FOLLOWING:
   1. Create a partition
   2. Specify the active partition
   3. Delete a partition
   4. Change between Solaris and Solaris2 Partition IDs
   5. Exit (update disk configuration and exit)
   6. Cancel (exit without updating disk configuration)
Select the partition type to create:
   1=SOLARIS2  2=UNIX        3=PCIXOS     4=Other
   5=DOS12     6=DOS16       7=DOSEXT     8=DOSBIG
   9=DOS16LBA  A=x86 Boot    B=Diagnostic C=FAT32
   D=FAT32LBA  E=DOSEXTLBA   F=EFI        0=Exit? c
```

7 指定此分区所使用的磁盘百分比。

```
       Total disk size is 29 cylinders
              Cylinder size is 2048 (512 byte) blocks

                                                 Cylinders
      Partition   Status    Type          Start   End   Length    %
      =========   ======    ============  =====   ===   ======   ===

WARNING: no partitions are defined!

SELECT ONE OF THE FOLLOWING:
   1. Create a partition
```

```
   2. Specify the active partition
   3. Delete a partition
   4. Change between Solaris and Solaris2 Partition IDs
   5. Exit (update disk configuration and exit)
   6. Cancel (exit without updating disk configuration)
Select the partition type to create:
Specify the percentage of disk to use for this partition (or type "c" to
specify the size in cylinders). 100
```

8 选择新分区应为活动分区还是非活动分区。

```
              Total disk size is 29 cylinders
              Cylinder size is 2048 (512 byte) blocks

                                                 Cylinders
      Partition   Status    Type          Start   End   Length    %
      =========   ======    ============  =====   ===   ======   ===

WARNING: no partitions are defined!

SELECT ONE OF THE FOLLOWING:
   1. Create a partition
   2. Specify the active partition
   3. Delete a partition
   4. Change between Solaris and Solaris2 Partition IDs
   5. Exit (update disk configuration and exit)
   6. Cancel (exit without updating disk configuration)
Select the partition type to create:
Should this become the active partition? If yes, it  will be activated
each time the computer is reset or turned on.
Please type "y" or "n". n
```

9 更新磁盘配置并退出。

```
      Total disk size is 29 cylinders
             Cylinder size is 2048 (512 byte) blocks

                                                 Cylinders
      Partition   Status    Type          Start   End   Length    %
      =========   ======    ============  =====   ===   ======   ===
          1                 Win95 FAT32       1    28      28     97

SELECT ONE OF THE FOLLOWING:
   1. Create a partition
   2. Specify the active partition
   3. Delete a partition
   4. Change between Solaris and Solaris2 Partition IDs
   5. Exit (update disk configuration and exit)
   6. Cancel (exit without updating disk configuration)
Enter Selection: 5
```

10 在此分区中创建 PCFS 文件系统。

在创建新的文件系统之前，确保取消挂载设备。有关取消挂载 USB 设备的更多信息，请参见第 136 页中的“如何挂载或取消挂载 USB 海量存储设备”。

例如：

```
# mkfs -F pcfs -o fat=32 /dev/rdsk/c3t0d0p0:c
Construct a new FAT file system on /dev/rdsk/c3t0d0p0:c: (y/n)? y
```

▼ 如何在 USB 海量存储设备中创建 Solaris 分区和修改分片

以下步骤说明如何创建 Solaris 分区和修改分片。

执行此任务之前，请务必备份所有数据。

1 **成为超级用户。**

2 **启动 `fdisk` 实用程序。**

例如：

```
# fdisk /dev/rdsk/c5t0d0s2
No fdisk table exists. The default partition for the disk is:

  a 100% "SOLARIS System" partition

Type "y" to accept the default partition,  otherwise type "n" to edit the
 partition table.
y
```

3 **显示当前的分片。**

例如：

```
# prtvtoc /dev/rdsk/c5t0d0s2
* /dev/rdsk/c5t0d0s2 partition map
*
* Dimensions:
*     512 bytes/sector
*      63 sectors/track
*     255 tracks/cylinder
*   16065 sectors/cylinder
*    5836 cylinders
*    5836 accessible cylinders
*
* Flags:
*   1: unmountable
*  10: read-only
*
*                          First     Sector    Last
* Partition  Tag  Flags    Sector     Count    Sector  Mount Directory
       0      0    00          0  93755340  93755339
       2      0    00          0  93755340  93755339
```

4 **创建包含分片信息的文本文件。**

例如：

```
slices: 0 = 0, 5GB, "wm", "home" :
        1 = 8225280000, 6GB :
        2 = 0, 44GB, "wm", "backup" :
        6 = 16450560000, 15GB
```

请确保每个分片都从柱面边界开始。例如，分片 1 的初始大小为 822280000 字节，即柱面大小与 1000 的乘积（以字节为单位）。

有关更多信息，请参见 rmformat(1) 中的 -s 选项说明。

5 **通过包含以上创建的分片文件来创建分片。**

例如：

```
# rmformat -s slice_file /dev/rdsk/c5t0d0s2
```

6 **查看新的分片信息。**

例如：

```
# prtvtoc /dev/rdsk/c5t0d0s2
* /dev/rdsk/c5t0d0s2 partition map
*
* Dimensions:
*     512 bytes/sector
*      63 sectors/track
*     255 tracks/cylinder
*   16065 sectors/cylinder
*    5836 cylinders
*    5836 accessible cylinders
*
* Flags:
*   1: unmountable
*  10: read-only
*
* Unallocated space:
*       First     Sector    Last
*       Sector     Count    Sector
*    10485760   5579240  16064999
*    28647912   3482088  32129999
*    63587280  30168060  93755339
*
*                          First     Sector    Last
* Partition  Tag  Flags    Sector     Count    Sector  Mount Directory
       0      8    00          0  10485760  10485759
       1      3    01   16065000  12582912  28647911
       2      5    00          0  92274688  92274687
       6      4    00   32130000  31457280  63587279
```

▼ 如何挂载或取消挂载 USB 海量存储设备

以下步骤说明了如何挂载和取消挂载 USB 海量存储设备。

1 **成为超级用户或控制台用户。**

2 **（可选的）标识该设备。**

例如：

```
$ rmformat
 Looking for devices...
       1. Logical Node: /dev/rdsk/c3t0d0p0
          Physical Node: /pci@0,0/pci108e,534a@2,1/storage@3/disk@0,0
          Connected Device: SanDisk  Cruzer Micro     0.3
```

```
Device Type: Removable
Bus: USB
Size: 245.0 MB
Label: <None>
Access permissions: Medium is not write protected.
```

在本示例中，物理磁盘设备为 `c2t0d0p0`。

3 选择以下操作之一以挂载或取消挂载 USB 海量存储设备：

- 以控制台用户身份挂载 USB 海量存储设备。

  可以使用含有设备别名、挂载点或设备路径的 `rmmount` 命令，类似以下内容：

  ```
  $ rmmount rmdisk0
  $ rmmount NONAME
  $ rmmount /dev/dsk/c3t0d0p0:1
  ```

  例如：

  ```
  $ rmmount NONAME
  NONAME /dev/dsk/c2t0d0p0 mounted
  $ ls /media/NONAME
  AA.TXT
  ```

- 以控制台用户身份取消挂载 USB 海量存储设备。

  例如：

  ```
  $ rmumount NONAME
  NONAME /dev/dsk/c2t0d0p0 unmounted
  ```

- 以超级用户身份挂载 USB 海量存储设备。

  本示例说明如何挂载包含 UFS 文件系统的设备：

  ```
  # mount /dev/dsk/c1t0d0s2 /mnt
  ```

  本示例说明如何在 SPARC 系统上挂载包含 PCFS 文件系统的设备：

  ```
  # mount -F pcfs /dev/dsk/c3t0d0s2:c /mnt
  ```

  本示例说明如何在 x86 系统上挂载包含 PCFS 文件系统的设备：

  ```
  # mount -F pcfs /dev/dsk/c3t0d0p0:c /mnt
  ```

  本示例说明如何挂载包含只读 HSFS 文件系统的 CD：

  ```
  # mount -F hsfs -o ro /dev/dsk/c1t0d0s2 /mnt
  ```

- 以超级用户身份取消挂载 USB 海量存储设备。

  首先，确保无人正在使用该设备中的文件系统。

  例如：

  ```
  # fuser -c -u /mnt
  # umount /mnt
  ```

4 **弹出该设备（对于 DVD、CD 或磁盘设备，此操作可选）。**

例如：

```
# eject /dev/rdsk/c1t0d0s2
```

USB 海量存储设备的疑难解答提示

如果在添加或移除 USB 海量存储设备时遇到问题，请牢记以下提示。

请检查 `/var/adm/messages` 文件中枚举该设备时出现的故障。对于枚举故障，插入 USB 集线器，或移除集线器并将其直接连接至根 USB 集线器。

- 如果系统正在运行时访问连接的设备遇到问题，请尝试以下命令：

  ```
  # devfsadm
  ```

- 如果因暂停操作关闭了系统电源，请勿来回移动设备。有关更多信息，请参见第 118 页中的 “SPARC: USB 电源管理”。
- 如果应用程序在设备使用时已将其移除并且该设备不再可用，请停止应用程序。使用 `prtconf` 命令可查看是否已移除了设备节点。

禁用特定的 USB 驱动程序

通过禁用 USB 设备的客户机驱动程序，可以禁用特定类型的 USB 设备。例如，通过禁用驱动 USB 打印机的 `usbprn` 驱动程序，可以禁用这些打印机。禁用 `usbprn` 不会影响其他种类的设备，如 USB 存储设备。

下表提供了一些 USB 设备类型及其对应的驱动程序。

设备类型	要禁用的驱动程序
音频	`usb_ac` 和 `usb_as`
HID（通常为键盘和鼠标）	`hid`
存储	`scsa2usb`
打印机	`usbprn`
串行	`usbser_edge`

如果禁用仍与系统连接的 USB 设备的驱动程序，则会显示类似如下的控制台消息：

```
usba10: WARNING: usba:    no driver found for device name
```

▼ 如何禁用特定的 USB 驱动程序

1. **成为超级用户。**
2. **从 /etc/system 文件中排除驱动程序别名项。**
   例如，包含以下排除语句，用于排除 usbprn 驱动程序。
   ```
   exclude: usbprn
   ```
3. **重新引导系统。**
   ```
   # init 6
   ```

▼ 如何移除未使用的 USB 设备链接

如果在系统电源关闭时移除 USB 设备，请使用此过程。在系统电源关闭时移除 USB 设备可能会保留不存在设备的设备链接。

1. **成为超级用户。**
2. **关闭可能正在访问该设备的所有应用程序。**
3. **移除特定 USB 类的未使用的链路。**
   例如：
   ```
   # devfsadm -C -c audio
   ```
   或者，直接移除悬而未决的链路：
   ```
   # devfsadm -C
   ```

使用 USB 音频设备（任务列表）

任务	说明	参考
添加 USB 音频设备。	添加 USB 麦克风和扬声器。	第 141 页中的“如何添加 USB 音频设备”
确定系统的主音频设备。	确定哪个音频设备是主音频设备。	第 141 页中的“如何确定系统的主音频设备”
更改主 USB 音频设备。	如果移除或更改 USB 音频设备，则可能需要使某个音频设备成为主音频设备。	第 142 页中的“如何更改主 USB 音频设备”

任务	说明	参考
移除未使用的 USB 设备链接。	如果在系统电源关闭时移除 USB 音频设备，则 /dev/audio 设备可能会指向不存在的 /dev/sound/* 设备。	第 139 页中的 “如何移除未使用的 USB 设备链接”
解决 USB 音频问题。	如果 USB 扬声器无法发声，请使用本部分来解决此问题。	第 142 页中的 “USB 音频设备问题疑难解答”

使用 USB 音频设备

有关特定 Oracle Solaris 发行版中的 USB 音频支持的信息，请参见第 109 页中的 “Oracle Solaris 对 USB 设备的支持”。

此 Oracle Solaris 发行版提供了由一对协作驱动程序 usb_ac 和 usb_as 实现的 USB 音频支持。音频控制驱动程序 usb_ac 是符合 Solaris USB 体系结构的客户机驱动程序，用于向用户应用程序提供控制接口。音频流驱动程序 usb_as 可在播放和录制期间处理音频数据消息。该驱动程序将设置样例频率和精度，并对来自 usb_ac 驱动程序的请求进行编码。这两个驱动程序都符合 USB 音频类 1.0 规范。

某些音频设备可以在软件控制下设置音量。为了管理此功能，可以将 STREAMS 模块 usb_ah 推送至 HID 驱动程序的顶部。

Oracle Solaris 支持仅播放、仅录制或同时播放和录制的 USB 音频设备。支持热插拔 USB 音频设备。

- 具有 USB 连接器的 SPARC Ultra 和 x86 平台支持 USB 音频设备。
- 有关完全支持的音频数据格式的信息，请参见 usb_ac(7D)。

主音频设备为 /dev/audio。通过使用以下命令，可以验证 /dev/audio 是否指向 USB 音频：

```
% mixerctl
Device /dev/audioctl:
  Name    = USB Audio
  Version = 1.0
  Config  = external

Audio mixer for /dev/audioctl is enabled
```

连接 USB 音频设备后，可以通过 /dev/sound/*N* 设备链接使用 audioplay 和 audiorecord 命令对其进行访问。

请注意，/dev/audio 和 /dev/sound/N 设备可能是指扬声器、麦克风或组合设备。如果引用不正确的设备类型，则该命令将失败。例如，如果尝试将其与麦克风一同使用，则 audioplay 命令将失败。

对于大多数 Oracle 音频应用程序（如 audioplay 和 audiorecord），通过设置 AUDIODEV shell 变量或为这些命令指定 -d 选项，可以选择特定的缺省音频设备。但是，设置 AUDIODEV 不适用于将 /dev/audio 硬编码为音频文件的第三方应用程序。

插入 USB 音频设备时，如果没有正在使用 /dev/audio，则该设备将自动成为主音频设备 /dev/audio。有关将 /dev/audio 从板载音频更改为 USB 音频或从 USB 音频更改为板载音频的说明，请参阅第 142 页中的 “如何更改主 USB 音频设备”和 usb_ac(7D)。

热插拔多个 USB 音频设备

如果将 USB 音频设备插入系统，该设备将成为主音频设备 /dev/audio。即使在重新引导系统后，该设备也仍为主音频设备。如果插入其他 USB 音频设备，则最后一个设备将成为主音频设备。

有关 USB 音频设备问题疑难解答的其他信息，请参见 usb_ac(7D)。

▼ 如何添加 USB 音频设备

1. **插入 USB 扬声器。**

   主音频设备 /dev/audio 指向 USB 扬声器。

   ```
   % ls -l /dev/audio
   lrwxrwxrwx   1 root     root    10 Feb 13 08:46 /dev/audio -> usb/audio0
   ```

2. **（可选的）移除扬声器。然后，重新插入扬声器。**

   如果移除扬声器，则 /dev/audio 设备将恢复回板载音频。

   ```
   % ls -l /dev/audio
   lrwxrwxrwx   1 root     root    7 Feb 13 08:47 /dev/audio -> sound/0
   ```

3. **添加 USB 麦克风。**

   ```
   % ls -l /dev/audio
   lrwxrwxrwx   1 root     root    10 Feb 13 08:54 /dev/audio -> usb/audio1
   ```

▼ 如何确定系统的主音频设备

此过程假定已连接 USB 音频设备。

- **检查系统的新音频链路。**
  - 使用 ls 命令显示系统的新音频链路。

    例如：

    ```
    % ls -lt /dev/audio*
    lrwxrwxrwx   1 root  root      7 Jul 23 15:46 /dev/audio -> usb/audio0
    lrwxrwxrwx   1 root  root     10 Jul 23 15:46 /dev/audioctl ->
    ```

```
usb/audioctl0/
% ls -lt /dev/sound/*
lrwxrwxrwx   1 root  root     74 Jul 23 15:46 /dev/sound/1 ->
../../devices/pci@1f,4000/usb@5/hub@1/device@3/sound-control@0:...
lrwxrwxrwx   1 root  root     77 Jul 23 15:46 /dev/sound/1ctl ->
../../devices/pci@1f,4000/usb@5/hub@1/device@3/sound-control@0:...
lrwxrwxrwx   1 root  other    66 Jul 23 14:21 /dev/sound/0 ->
../../devices/pci@1f,4000/ebus@1/SUNW,CS4231@14,200000:sound,audio
lrwxrwxrwx   1 root  other    69 Jul 23 14:21 /dev/sound/0ctl ->
../../devices/pci@1f,4000/ebus@1/SUNW,CS4231@14,200000:sound,audioctl
%
```

请注意，主音频设备 `/dev/audio` 指向新插入的 USB 音频设备 `/dev/usb/audio0`。

- 还可以使用 `prtconf` 命令来检查系统的 USB 音频设备，并查找 USB 设备信息。

```
% prtconf
.
.
.
usb, instance #0
   hub, instance #0
        mouse, instance #0
        keyboard, instance #1
        device, instance #0
            sound-control, instance #0
            sound, instance #0
            input, instance #0
.
.
.
```

▼ 如何更改主 USB 音频设备

- **选择以下操作之一可更改主 USB 音频设备。**
  - 如果要使板载音频设备成为主音频设备，请移除 USB 音频设备。然后，`/dev/audio` 链接即会指向 `/dev/sound/0` 项。如果 `/dev/sound/0` 项不是主音频设备，请关闭系统并使用 `boot -r` 命令，或以超级用户身份运行 `devfsadm -i` 命令。
  - 如果要使 USB 音频设备成为主音频设备，只需插入 USB 音频设备并检查设备链接即可。

USB 音频设备问题疑难解答

有时，USB 扬声器不会发出任何声音，即使连接了驱动程序并将音量设置为高也是如此。热插拔该设备可能不会更改该行为。

解决方法是关闭然后再打开 USB 扬声器的电源。

音频设备所有权要点

使用音频设备时，请牢记音频设备所有权的以下要点：

- 插入 USB 音频设备并登录控制台后，控制台即是 /dev/* 项的所有者。此情况表明，只要登录控制台，便可使用音频设备。
- 如果插入 USB 音频设备时未登录控制台，则超级用户会成为该设备的所有者。但是，如果登录控制台并尝试访问 USB 音频设备，则设备所有权将归控制台所有。有关更多信息，请参见 logindevperm(4)。
- 如果使用 rlogin 命令远程进行登录并尝试访问 USB 音频设备，则所有权不会更改。这表明，未经授权的用户无法通过他人拥有的麦克风侦听会话。

使用 cfgadm 命令热插拔 USB 设备（任务列表）

任务	说明	参考
显示 USB 总线信息。	显示有关 USB 设备和总线的信息。	第 144 页中的“如何显示 USB 总线信息 (cfgadm)”
取消配置 USB 设备。	以逻辑方式取消配置仍通过物理方式连接至系统的 USB 设备。	第 145 页中的“如何取消配置 USB 设备”
配置 USB 设备。	配置先前取消配置的 USB 设备。	第 146 页中的“如何配置 USB 设备”
以逻辑方式断开 USB 设备连接。	如果物理位置远离系统，则可以逻辑方式断开 USB 设备连接。	第 146 页中的“如何以逻辑方式断开 USB 设备连接”
以逻辑方式连接 USB 设备。	以逻辑方式连接先前通过逻辑方式断开连接或取消配置的 USB 设备。	第 146 页中的“如何以逻辑方式连接 USB 设备”
断开 USB 设备子树连接。	断开 USB 设备子树连接，该子树是集线器下的设备分层结构（或树）。	第 147 页中的“如何以逻辑方式断开 USB 设备子树连接”
重置 USB 设备。	重置 USB 设备，以便以逻辑方式移除和重新创建设备。	第 147 页中的“如何重置 USB 设备”
更改多重配置 USB 设备的缺省配置。	更改多重配置 USB 设备的缺省配置。	第 148 页中的“如何更改多重配置 USB 设备的缺省配置”

使用 cfgadm 命令热插拔 USB 设备

可以在正在运行的系统中添加或移除 USB 设备，而不必使用 cfgadm 命令。不过，USB 设备也可以**逻辑方式**热插拔，而不必以物理方式移除。在远程位置工作并且需要禁用或重置未正常运行的 USB 设备时，这种方案比较方便。cfgadm 命令还可提供显示 USB 设备树（包括制造商和产品信息）的方法。

cfgadm 命令可以显示有关**附着点**的信息，附着点是指系统中可以执行动态重新配置操作的位置。连接点由以下几项组成：

- 插卡，表示可以配置到系统中的硬件资源，如 USB 设备
- 插座，接受插卡的位置，如 USB 端口

连接点由逻辑连接点 ID 和物理连接点 ID (Ap_Ids) 表示。物理 Ap_Id 是连接点的物理路径名。逻辑 Ap_Id 是物理 Ap_Id 的用户友好替代项。有关 Ap_Id 的更多信息，请参见 cfgadm_usb(1M)。

cfgadm 命令提供了以下 USB 设备状态信息。

插座状态	说明
empty/unconfigured	设备不是通过物理方式连接的。
disconnected/unconfigured	设备以逻辑方式断开连接且不可用，即使仍可以通过物理方式连接设备也是如此。
connected/unconfigured	设备以逻辑方式建立连接但不可用。设备在 prtconf 输出中可见。
connected/configured	设备已连接并且可用。

以下各小节介绍如何使用 cfgadm 命令通过软件热插拔 USB 设备。这些小节中的所有样例 USB 设备信息都已截断，以便集中阐明相关信息。

▼ 如何显示 USB 总线信息 (cfgadm)

有关使用 prtconf 命令显示 USB 配置信息的示例，请参见第 129 页中的 “如何显示 USB 设备信息”。

1 显示 USB 总线信息。

例如：

```
% cfgadm
Ap_Id                  Type         Receptacle   Occupant     Condition
usb0/4.5               usb-hub      connected    configured   ok
usb0/4.5.1             usb-device   connected    configured   ok
usb0/4.5.2             usb-printer  connected    configured   ok
usb0/4.5.3             usb-mouse    connected    configured   ok
usb0/4.5.4             usb-device   connected    configured   ok
usb0/4.5.5             usb-storage  connected    configured   ok
usb0/4.5.6             usb-communi  connected    configured   ok
usb0/4.5.7             unknown      empty        unconfigured ok
```

在前面的示例中，usb0/4.5.1 标识连接至二级外部集线器的端口 1 的设备，二级外部集线器连接至一级外部集线器的端口 5，而一级外部集线器连接至第一个 USB 控制器的根集线器端口 4。

2 **显示特定 USB 设备信息。**

例如：

```
% cfgadm -l -s "cols=ap_id:info"
Ap_Id       Information
 usb0/4.5.1  Mfg: Inside Out Networks Product: Edgeport/421 NConfigs: 1
Config: 0  : ...
 usb0/4.5.2  Mfg: <undef> Product: <undef>   NConfigs: 1 Config: 0 ...
 usb0/4.5.3  Mfg: Mitsumi Product: Apple USB Mouse NConfigs: 1
Config: 0 ...
 usb0/4.5.4  Mfg: NMB  Product: NMB USB KB/PS2 M NConfigs: 1 Config: 0
 usb0/4.5.5  Mfg: Hagiwara Sys-Com  Product: SmartMedia R/W  NConfigs: 1
Config: 0 : ...
 usb0/4.5.6  Mfg: 3Com Inc.  Product: U.S.Robotics 56000 Voice USB Modem
NConfigs: 2 ...
 usb0/4.5.7
```

▼ 如何取消配置 USB 设备

可以取消配置仍通过物理方式连接至系统的 USB 设备。但是，无法再将驱动程序连接至该设备。请注意，`prtconf` 输出中会始终显示 USB 设备，即使已取消配置该设备也是如此。

1 **成为超级用户。**

2 **取消配置 USB 设备。**

例如：

```
# cfgadm -c unconfigure usb0/4.7
Unconfigure the device: /devices/pci@8,700000/usb@5,3/hub@4:4.7
This operation will suspend activity on the USB device
Continue (yes/no)? y
```

3 **验证是否已取消配置该设备。**

例如：

```
# cfgadm
Ap_Id                  Type         Receptacle   Occupant     Condition
usb0/4.5               usb-hub      connected    configured   ok
usb0/4.5.1             usb-device   connected    configured   ok
usb0/4.5.2             usb-printer  connected    configured   ok
usb0/4.5.3             usb-mouse    connected    configured   ok
usb0/4.5.4             usb-device   connected    configured   ok
usb0/4.5.5             usb-storage  connected    configured   ok
usb0/4.5.6             usb-communi  connected    configured   ok
usb0/4.5.7             unknown      empty        unconfigured ok
usb0/4.6               usb-storage  connected    configured   ok
usb0/4.7               usb-storage  connected    unconfigured ok
```

▼ 如何配置 USB 设备

1. **成为超级用户。**
2. **配置 USB 设备。**

   例如：

   ```
   # cfgadm -c configure usb0/4.7
   ```

3. **验证是否已配置 USB 设备。**

   例如：

   ```
   # cfgadm usb0/4.7
   Ap_Id                  Type         Receptacle  Occupant    Condition
   usb0/4.7               usb-storage  connected   configured  ok
   ```

▼ 如何以逻辑方式断开 USB 设备连接

如果要从系统和 prtconf 输出中移除 USB 设备，但物理位置远离系统，则只需以逻辑方式断开 USB 设备连接即可。从物理角度来看，该设备仍然处于连接状态。但是，从逻辑角度来看，对系统而言设备已断开连接、不可用且不可见。

1. **成为超级用户。**
2. **断开 USB 设备连接。**

   例如：

   ```
   # cfgadm -c disconnect -y usb0/4.7
   ```

3. **验证是否已断开该设备的连接。**

   例如：

   ```
   # cfgadm usb0/4.7
   Ap_Id                  Type       Receptacle    Occupant      Condition
   usb0/4.7               unknown    disconnected  unconfigured  ok
   ```

▼ 如何以逻辑方式连接 USB 设备

使用此过程可以逻辑方式连接先前通过逻辑方式断开连接或取消配置的 USB 设备。

1. **成为超级用户。**
2. **连接 USB 设备。**

   例如：

   ```
   # cfgadm -c configure usb0/4.7
   ```

3 **验证是否已连接该设备。**

例如：

```
# cfgadm usb0/4.7
Ap_Id                  Type         Receptacle  Occupant    Condition
usb0/4.7               usb-storage  connected   configured  ok
```

对系统而言设备现在可用且可见。

▼ 如何以逻辑方式断开 USB 设备子树连接

使用此过程可以断开 USB 设备子树连接，该子树是集线器下的设备分层结构（或树）。

1 **成为超级用户。**

2 **移除 USB 设备子树。**

例如：

```
# cfgadm -c disconnect -y usb0/4
```

3 **验证是否已断开 USB 设备子树连接。**

例如：

```
# cfgadm usb0/4
Ap_Id                  Type       Receptacle   Occupant     Condition
usb0/4                 unknown    disconnected unconfigured ok
```

▼ 如何重置 USB 设备

如果 USB 设备行为异常，请使用 `cfgadm` 命令重置该设备，即以逻辑方式移除并重新创建该设备。

1 **成为超级用户。**

2 **确保设备未被使用。**

3 **重置设备。**

例如：

```
# cfgadm -x usb_reset -y usb0/4.7
```

4 **验证是否已连接该设备。**

例如：

```
# cfgadm usb0/4.7
Ap_Id                  Type         Receptacle   Occupant     Condition
usb0/4.7               usb-storage  connected    configured   ok
```

▼ 如何更改多重配置 USB 设备的缺省配置

使用多重配置 USB 设备时，请牢记以下要点：

- USB 设备配置定义了设备向操作系统显示其本身的方式。此方法不同于其他 `cfgadm` 部分中介绍的系统设备配置。
- 某些 USB 设备支持多种配置，但一次只能有一种配置处于活动状态。
- 通过检查 `cfgadm -lv` 输出，可以确定多重配置设备。`Nconfigs` 将大于 1。
- 缺省 USB 配置为配置 1。当前配置在 `cfgadm -lv` 输出中反映为 `Config`。
- 只要将设备重新连接至同一个端口，对缺省配置所做的更改在重新引导、热移除以及重新配置设备期间就将始终保持不变。

1 确保设备未被使用。

2 更改缺省 USB 配置。

例如：

```
# cfgadm -x usb_config -o config=2 usb0/4
   Setting the device: /devices/pci@1f,0/usb@c,3:4
   to USB configuration 2
   This operation will suspend activity on the USB device
   Continue (yes/no)? yes
```

3 验证设备是否已更改。

例如：

```
# cfgadm -lv usb0/4
Ap_Id  Receptacle   Occupant     Condition  Information When  Type
     Busy     Phys_Id
usb0/4 connected    unconfigured ok         Mfg: Sun  2000
Product: USB-B0B0 aka Robotech
With 6 EPPS High Clk Mode   NConfigs: 7  Config: 2  : EVAL Board Setup
unavailable
usb-device   n        /devices/pci@1f,0/usb@c,3:4
```

请注意，`Config:` 现在显示 2。

◆ ◆ ◆ 第 9 章

使用 InfiniBand 设备（概述/任务）

本章提供有关在网络中使用 InfiniBand (IB) 设备的一般概述和逐步说明。

以下是本章中概述信息的列表。

- 第 149 页中的 “InfiniBand 设备概述”
- 第 153 页中的 “动态重新配置 IB 设备 (cfgadm)”

有关与使用 IB 设备关联的过程的信息，请参见以下内容：

- 第 151 页中的 “动态重新配置 IB 设备（任务列表）”
- 第 161 页中的 “将 uDAPL 应用程序接口用于 InfiniBand 设备”
- 第 164 页中的 “管理 IPoIB 设备 (dladm)”
- 第 166 页中的 “监视并排除 IB 设备故障”

有关动态重新配置和热插拔的常规信息，请参见第 6 章，动态配置设备（任务）。

InfiniBand 设备概述

InfiniBand (IB) 是一项新的基于交换光纤结构的 I/O 技术。它为将 I/O 设备连接到主机以及主机到主机通信提供了高带宽、低延迟的互连。IB 设备由 Solaris IB 结点驱动程序来管理。

Oracle Solaris 11 中支持以下设备：

- 基于 IB 的 IP (IP over IB, IPoIB) 设备－能够通过 IB 连接传输 IP 包的功能，此功能由 ibp(7D) 驱动程序实现
- 插槽直接协议 (Socket Direct Protocol, SDP)－支持基于 IB 的插槽
- 可靠数据报服务第 3 版 (Reliable Datagram Service version 3, RDSv3) 和可靠数据报服务 (RDS)
- 基于 RDMA 的 NFS (NFS over RDMA, NFSoRDMA)-提供使用 RDMA 的基于 IB 的 NFS 服务

- 针对 RDMA 的 iSCSI 扩展 (iSCSI Extension for RDMA, iSER) — 为 iSCSI 协议提供 RDMA 数据传输功能
- 用户直接访问编程语言 (User Direct Access Programming Language, uDAPL)
- 开放光纤网络用户组件 (Open Fabric User Verb, OFUV)
- 基于 IB 的以太网 (Ethernet over IB, EoIB)

IB 结点驱动程序可查询 Solaris IB 设备管理器 (IB Device Manager, IBDM) 中的服务（在本指南中称为**通信服务**），以枚举 IB 端口设备、HCA_SVC 设备和 IB VPPA 设备。

IB 分区链路代表数据链路的一个新**部件**类别，并且此对象通过使用新的 `dladm` 子命令进行管理。IB 分区链路可以在 IB 物理链路的顶部创建，与端口上的 `P_Key` 一一对应。分区链路用于数据传输。

端口设备会将通信服务绑定至主机通道适配器 (Host Channel Adapter, HCA) 的给定 `port#`。而 VPPA 设备会将通信服务绑定至 `port#, p_key#` 组合。HCA_SVC 设备会将通信服务绑定至给定的 HCA。请注意，端口设备和 HCA_SVC 设备始终使用其值为零的 `p_key`（分区密钥）。端口、HCA_SVC 和 VPPA 设备都是 HCA 的子设备，并且通过 `ib.conf` 文件进行枚举。有关更多信息，请参见 ib(7D)。

IOC 设备是 IB 结点驱动程序的子程序，同时也是 I/O 单元的一部分。伪设备也是 IB 结点驱动程序的子程序，并且指代提供各自要枚举的配置文件的其他所有设备。有关更多信息，请参见 ib(4)。

下表列出了可能的 IB 设备树路径名。

IOC 设备	`/ib/ioc@1730000007F510C,173000007F50`
IB 伪设备	`/ib/<driver>@<unit-address>`
IB VPPA 设备	`/pci@1f,2000/pci@1/pci15b3,5a44@0/ibport@<port#>, <P_key>,<service>`
IB HCA_SVC 设备	`/pci@1f,2000/pci@1/pci15bc,5a44@0/ibport@0,0,<service>`
IB 端口设备	`/pci@1f,2000/pci@1/pci15b3,5a44@0/ibport@<port#>,0, <service>`
HCA	`/pci@1f,2000/pci@1/pci15b3,5a44@0`

请注意，IB HCA_SVC 设备使用零作为 `port#` 和 `p_key` 的值。

下面介绍了上表中的 IB 组件：

<services> 通信服务。例如，`ipib` 是 `ibd` 内核客户机驱动程序所使用的通信服务。

<P_key> 使用的分区链路密钥值。

<port> 端口号。

<unit-address> 是指以其 `driver.conf` 文件中指定的名称命名的 IB 内核客户机驱动程序的属性。有关更多信息，请参见`driver.conf(4)`。

有关使用 IB 诊断命令和实用程序的信息，请参见第 166 页中的 “监视并排除 IB 设备故障”。

InfiniBand 软件包

与 IB 相关的软件包如下：

- `driver/infiniband/connectx` – Mellanox ConnectX Family InfiniBand HCA 与 10GbE NIC 驱动程序
- `driver/infiniband/tavor` – Tavor HCA 驱动程序
- `system/io/infiniband/ethernet-over-ib` – 实现基于 InfiniBand 的以太网的 InfiniBand 设备驱动程序
- `system/io/infiniband/ib-device-mgt-agent` – InfiniBand 设备管理器代理
- `system/io/infiniband/ib-sockets-direct` – InfiniBand 分层插槽直接协议
- `system/io/infiniband/ip-over-ib` – 支持基于 InfiniBand 的 IP (IP over InfiniBand, IPoIB) 协议的网络设备驱动程序
- `system/io/infiniband/open-fabrics` – 开放光纤网络内核组件
- `system/io/infiniband/reliable-datagram-sockets-v3` – 可靠数据报插槽 (RDSv3)
- `system/io/infiniband/reliable-datagram-sockets` – 可靠数据报插槽
- `system/io/infiniband/rpc-over-rdma` – 基于 RDMA 的 InfiniBand RPC 驱动程序
- `system/io/infiniband/udapl` – UDAPL 库和命令
- `system/io/infiniband` – InfiniBand 框架

动态重新配置 IB 设备（任务列表）

任务	说明	参考
显示 IB 设备信息。	显示有关系统中的 IB 设备的信息。	第 154 页中的 “如何显示 IB 设备信息”
配置或取消配置端口或 VPPA 设备。	选择以下操作之一：	

任务	说明	参考
	取消配置端口或 VPPA 设备。	第 155 页中的 “如何取消配置 IB Port、HCA_SVC 或 VPPA 设备”
	配置端口或 VPPA 设备。	第 156 页中的 “如何配置 IB 端口、HCA_SVC 或 VPPA 设备”
配置或取消配置 IB 伪设备。	选择以下操作之一：	
	取消配置 IB 伪设备。	第 157 页中的 “如何取消配置 IB 伪设备”
	配置IB 伪设备。	第 157 页中的 “如何配置 IB 伪设备”
显示 HCA 的内核 IB 客户机。	您可能需要显示有关 HCA 的内核 IB 客户机的信息，特别是要取消配置 HCA 时。	第 157 页中的 “如何显示 HCA 的内核 IB 客户机”
配置或取消配置 IB HCA 或 EoIB 接口。	选择以下操作之一：	
	取消配置与 HCA 连接的 IB 设备。	
	当 EoIB 设备活动时动态重新配置 HCA。	第 158 页中的 “如何在有活动 EoIB 设备的情况下动态重新配置 HCA”
	在有活动 EoIB 接口的情况下取消配置非最后一个 IB HCA。	第 159 页中的 “如何在热移除之后重新配置并恢复 EoIB 接口”
	配置与 HCA 连接的 IB 设备。	第 160 页中的 “配置 IB HCA”
更新 IB p_key 表。	如果 HCA 端口的 p_key 表信息发生更改，则需要通知 IBTF 和 IBDM，以便更新其内部 p_key 数据库。	第 160 页中的 “如何更新 IB p_key 表”
显示 IB 通信服务	显示当前由 IBTF 使用的 IB 通信服务。	第 160 页中的 “如何显示 IB 通信服务”
添加或删除 VPPA 通信服务。	选择以下操作之一：	
	添加 VPPA 通信服务。	第 160 页中的 “如何添加 VPPA 通信服务”
	删除 VPPA 通信服务。	第 161 页中的 “如何删除现有 IB 端口、HCA_SVC 或 VPPA 通信服务”

动态重新配置 IB 设备 (cfgadm)

只能使用 cfgadm CLI 配置或取消配置正在运行的系统中的 IB 设备。此命令还提供了显示 IB 光纤结构、管理通信服务以及更新 p_key 表数据库的方法。有关更多信息，请参见 cfgadm_ib(1M)。

cfgadm CLI 可以管理被主机视为整个 IB 光纤结构的动态重新配置（在本指南中称为 DR）。cfgadm 操作在端口、VPPA、HCA_SVC、IOC 和伪设备等所有 IB 设备上都受支持。

cfgadm 命令可以显示有关连接点 (Ap_Id) 的信息，连接点是指系统中可以进行 DR 操作的位置。有关 cfgadm 支持的 Ap_Id 的详细信息，请参见 cfgadm_ib.1M。请注意，所有 IB Ap_Id 都显示为 connected。

cfgadm 命令提供了以下 IB 设备状态信息。

插座状态	说明
connected/configured/ok	设备已连接并且可用。存在 devinfo 节点。
connected/unconfigured/unknown	设备不可用，并且不存在与设备对应的 devinfo 节点或设备驱动程序。或者，设备从未配置为供 ib 节点驱动程序使用。设备可能会由 IB 设备管理器识别。

本节介绍的所有过程需要管理特权，一般不授予用户帐户。有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。

以下各节介绍如何使用 cfgadm 命令动态重新配置 (DR) IB 设备。这些小节中的所有样例 IB 设备的信息都已截断，以便集中介绍相关信息。

取消配置 IB 设备考虑因素

HCA 的实际动态重新配置 (DR) 超出了 IB cfgadm 插件的范围。尽管如此，仍可以使用底层总线的插件来实现 HCA 的 DR。例如，基于 PCI 的 HCA 可以使用 cfgadm_pci 命令。有关更多信息，请参见 cfgadm_pci(1M)。

注 – RDSv3 不支持取消配置 HCA。如果系统在动态重新配置 (DR) 时已经装有 RDSv3 驱动程序，取消配置 HCA 会失败，显示如下。

```
# cfgadm -c unconfigure ib::rdsv3,0
This operation will suspend activity on the IB device
Continue (yes/no)? yes
cfgadm: Hardware specific failure: unconfigure operation failed ap_id: /devices/ib:fabric::rdsv3,0

# cfgadm -c unconfigure PCI-EM0
cfgadm: Component system is busy, try again: unconfigure failed
```

解决方法：

在生产系统上，HCA 动态重新配置操作和重新引导系统之前，必须删除 RDSv3 驱动程序。

```
# rem_drv rdsv3
Device busy
Cannot unload module: rdsv3
Will be unloaded upon reboot.

# init 6
```

▼ 如何显示 IB 设备信息

可以使用 `prtconf` 命令来显示有关 IB 设备的常规信息。例如：

```
$ prtconf
.
.
.
    ib, instance #0
        rpcib, instance #0
        rdsib, instance #0
        daplt, instance #0
        rdsv3, instance #0
        sdpib, instance #0
        eibnx, instance #0
        sol_umad, instance #0
        sol_uverbs, instance #0
        iser, instance #0
.
.
.
    pci15b3,673c, instance #0
                 ibport, instance #0
                 ibport, instance #1
```

在上面的示例中，`pci15b3,673c` 是指 IB HCA。

执行以下步骤可显示特定 IB 设备的信息。

1. **成为管理员。**

2. **显示 IB 光纤结构信息。**

   例如：

```
# cfgadm -al
Ap_Id                          Type         Receptacle     Occupant     Condition
ib                             IB-Fabric    connected      configured   ok
hca:21280001A0A478             IB-HCA       connected      configured   ok
ib::21280001A0A47A,0,ipib      IB-PORT      connected      configured   ok
ib::21280001A0A479,0,ipib      IB-PORT      connected      configured   ok
ib::1730000008070,0,hnfs       IB-HCA_SVC   connected      configured   ok
ib::daplt,0                    IB-PSEUDO    connected      configured   ok
ib::eibnx,0                    IB-PSEUDO    connected      configured   ok
ib::iser,0                     IB-PSEUDO    connected      configured   ok
ib::rdsib,0                    IB-PSEUDO    connected      configured   ok
ib::rdsv3,0                    IB-PSEUDO    connected      configured   ok
ib::rpcib,0                    IB-PSEUDO    connected      configured   ok
ib::sdpib,0                    IB-PSEUDO    connected      configured   ok
ib::sol_umad,0                 IB-PSEUDO    connected      configured   ok
ib::sol_uverbs,0               IB-PSEUDO    connected      configured   ok
```

   在上面的示例输出中，组件的描述如下：

`Ap_Id ib::21280001A0A47A,0,ipib`	标识与端口 GUID 连接并且绑定至 `ipib` 服务的 IB 端口设备。
`Ap_ID ib::sdpib,0`	标识伪设备。
`Ap_ID hca:21280001A0A478`	标识 HCA 设备。
`Ap_ID ib::1730000008070,0,hnfs`	标识绑定至 `hnfs` 服务的 IB HCA_SVC 设备。
`Ap_Id ib::ibgen,0`	标识伪设备。

3. **显示特定 IB 设备的信息。**

   例如，对于 IB 端口设备，将显示以下信息：

```
# cfgadm -al -s "cols=ap_id:info" ib::21280001A0A47A,0,ipib
Ap_Id                          Information
ib::21280001A0A47A,0,ipib      ipib
```

   例如，对于 IB HCA 设备，将显示以下信息：

```
# cfgadm -al -s "cols=ap_id:info" hca::1730000008070
Ap_Id                          Information
hca::1730000008070             VID: 0x15b3, PID: 0x5a44, #ports: 0x2,
port1 GUID: 0x1730000008071, port2 GUID: 0x1730000008072
```

   前面的输出显示了端口号和端口的 GUID。

▼ 如何取消配置 IB Port、HCA_SVC 或 VPPA 设备

如果要从系统中删除 IB 端口、HCA_SVC 或 VPPA 设备，请执行以下步骤。

以下示例说明如何取消配置 IB 端口设备，该过程也同样适用于 VPPA 和 HCA_SVC 设备。

1 **成为管理员。**

2 **取消配置虚拟 IB 端口设备。**

例如：

```
# cfgadm -c unconfigure ib::1730000007F51,*0*,ipib
Unconfigure the device: /devices/ib:fabric::1730000007F51,*0*,ipib
This operation will suspend activity on the IB device
Continue (yes/no)? Y
```

3 **验证是否已断开该设备的连接。**

例如：

```
# cfgadm -a ib::1730000007F51,*0*,ipib
Ap_Id                       Type    Receptacle Occupant     Condition
ib::1730000007F51,*0*,ipib IB-VPPA  connected  unconfigured unknown
```

▼ 如何配置 IB 端口、HCA_SVC 或 VPPA 设备

如果要在系统中配置 IB 端口、HCA_SVC 或 VPPA 设备，请执行以下步骤。

以下示例说明如何配置 VPPA 设备，类似的步骤也可用于 IB 端口设备和 HCA_SVC 设备。

1 **成为管理员。**

2 **配置虚拟 IB 端口设备。**

例如：

```
# cfgadm -c configure ib::1730000007F51,*0*,ipib
```

3 **验证是否已连接该设备。**

例如：

```
# cfgadm -a ib::1730000007F51,*0*,ipib
Ap_Id                       Type      Receptacle Occupant   Condition
ib::1730000007F51,*0*,ipib  IB-VPPA   connected  configured ok
```

注 – IB 端口设备和 HCA_SVC 设备的基于 `cfgadm` 的配置或取消配置操作与前面的 IB VPPA 设备示例类似。

▼ 如何取消配置 IB 伪设备

如果要从系统中删除 IB 伪设备，请执行以下步骤。

1. **成为管理员。**

2. **取消配置 IB 伪设备。**

   例如：

```
# cfgadm -c unconfigure ib::ibgen,0
Unconfigure the device: /devices/ib:fabric::ibgen,0
This operation will suspend activity on the IB device
Continue (yes/no)? Y
```

3. **验证是否已断开该设备的连接。**

```
# cfgadm -a ib::ibgen,0
Ap_Id                  Type      Receptacle Occupant     Condition
ib::ibgen,0            IB-PSEUDO connected  unconfigured unknown
```

▼ 如何配置 IB 伪设备

执行以下步骤可配置 IB 伪设备。

1. **成为管理员。**

2. **配置 IB 伪设备。**

   例如：

```
# cfgadm -yc configure ib::ibgen,0
```

3. **验证是否已连接该设备。**

   例如：

```
# cfgadm -a ib::ibgen,0
Ap_Id                  Type       Receptacle Occupant   Condition
ib::ibgen,0            IB-PSEUDO  connected  configured ok
```

▼ 如何显示 HCA 的内核 IB 客户机

可以调用以下 IB `cfgadm` 插件命令来列出使用此 HCA 的内核 IB 客户机。请注意，如果内核 IB 客户机使用其他 HCA，则最后一列会显示 "yes"（是）。不使用 HCA 的 IB 管理器和内核客户机以 `Ap_Id` "-" 显示。

- **显示 HCA 的内核 IB 客户机。**

  例如：

```
$ cfgadm -x list_clients hca:173000007F50
Ap_Id                        IB Client         Alternate HCA
ib::1730000007F51D0             ibgen              no
ib::1730000007F51D1             ibgen              no
ib::1730000007F51,0,ipib        ibd                no
ib::ibgen,0                     ibgen              no
-                               ibdm               no
-                               ibmf               no
-                               nfs/ib             no
```

▼ 如何在有活动 EoIB 设备的情况下动态重新配置 HCA

尝试在系统中执行动态重新配置 (DR) 操作时，如果有 Connect-X 系列 IB HCA 用于活动 EoIB 接口（例如 EoIB 数据链路是活动的 (plumbed) 或 VNIC 是基于 HCA 的数据链路创建的），则在满足以下条件之一时，DR 操作可成功执行：

- 该 Connect-X 系列 IB HCA 是系统中唯一的 Connect-X 系列 HCA。
- 系统中有多个 Connect-X HCA，但该适配器是要取消配置的最后一个。

在有活动 EoIB 接口的情况下，如果不满足以上条件，对 IB HCA 的动态重新配置 (DR) 操作将失败，并显示 `cfgadm` 错误消息。

如果 DR 操作失败，必须对活动的 EoIB 接口进行取消激活 (unplumb) 并删除数据链路上的 VNIC，然后重试 DR 操作。此外，在这种情况下，如果在 DR 操作后更换了同一插槽中的 Connect-X 系列 HCA，并使用 `cfgadm` 或 `hotplug` 命令对其进行了配置，必须重新激活 (replumb) 之前取消激活 (unplumb) 的 EoIB 数据链路并重新创建之前删除的 VNIC。

请注意，在使用的 EoIB 接口要求在 Sun Network QDR Infiniband 网关交换机的相应配置文件中列出 HCA 端口 GUID。因此，在热移除之后和重新插入之前，需要以新的 HCA 端口 GUID 信息更新网关交换机配置文件。有关如何更新配置的信息，请参见《Sun Network QDR Infiniband Gateway Switch Administration Guide》。

1. **在有多个 IB HCA 的系统中成为管理员。**

2. **尝试取消配置与 EoIB 数据链路相关的连接点。**

```
# cfgadm -c unconfigure PCI-EM0
cfgadm: Component system is busy, try again: unconfigure failed
```

3. **确定阻碍取消配置操作的 EoIB 接口。**

```
# dmesg | tail | grep 'failing HCA detach'
Aug 23 12:37:20  eoib: [ID 530795 kern.warning] WARNING: eoib0 still in use,
failing HCA detach
```

4 确定是否由于存在基于 **eoib0** 的 IP 接口而导致 eoib0 在使用。

```
# ipadm show-if eoib0
ipadm: cannot get information for interface(s): No such interface
```

5 如果不存在基于 **eoib0** 的 IP 接口，检查是否存在任何导致 **eoib0** 忙的 VNIC。

```
# dladm show-vnic
    LINK                OVER         SPEED  MACADDRESS        MACADDRTYPE       VID
    evnic0              eoib0        10000  2:8:20:e5:56:99   random            0
```

6 删除基于 **eoib0** 的 vnic。

```
# dladm delete-vnic evnic0
```

7 重试取消配置操作。

```
# cfgadm -c unconfigure PCI-EM0
```

▼ 如何在热移除之后重新配置并恢复 EoIB 接口

1 成为 Sun Network QDR 网关交换机的管理员。

2 编辑 Sun Network QDR 网关交换机上的 BXM 配置文件并使用新 HCA 的端口 GUID 替换所有旧的 HCA 端口 GUID。

3 重新启动 BXM。

4 成为 Oracle Solaris 系统的管理员。

5 在 Solaris 系统上热插入新 HCA 并使用 **cfgadm** 命令重新配置原来与 EoIB 数据链路关联的连接点。

6 确认原来的 EoIB 数据链路是否已恢复。

```
# dladm show-link | grep eoib0
eoib0               phys      1500   unknown       --
```

7 如果在之前的取消配置操作中显式删除了任何 VNIC，现在重新创建这些 VNIC。

```
# dladm create-vnic -l eoib0 evnic0
```

8 确认是否存在基于 **eoib0** 的 VNIC。

```
# dladm show-vnic
LINK         OVER     SPEED  MACADDRESS        MACADDRTYPE       VID
evnic0       eoib0    10000  2:8:20:e5:56:99   random              0
```

9 如果在之前的取消配置操作中显式删除了任何基于 **eoib0** 的 IP 地址，使用 **ipadm** 命令重新创建这些地址。

配置 IB HCA

调用总线特定的 cfgadm 插件来配置 HCA。确切的详细信息超出了本章的范围。

▼ 如何更新 IB p_key 表

如果 HCA 端口的 p_key 表信息发生更改（例如，启用或禁用了其他 p_key），则需要通知 InfiniBand 传输框架 (InfiniBand Transport Framework, IBTF) 和 IBDM，以便更新其内部 p_key 数据库。cfgadm 命令可帮助更新 IBTF 和 IBDM 的 p_key 数据库。有关更多信息，请参见 ibtl(7D) 和 ibdm(7D)。

1 **成为管理员。**

2 **更新 p_key 表。**

例如：

```
# cfgadm -x update_pkey_tbls -y ib
```

▼ 如何显示 IB 通信服务

执行以下步骤可显示当前由 IBTF 使用的通信服务。

1 **成为管理员。**

2 **显示 IB 通信服务。**

例如：

```
# cfgadm -x list_services ib
Port communication services:
                srp
VPPA communication services:
                ibd
HCA_SVC communication services:
                hnfs
```

▼ 如何添加 VPPA 通信服务

执行以下步骤可添加新的 VPPA 通信服务。

可以使用类似的步骤来添加新的 HCA_SVC 或端口通信服务。

1 **成为管理员。**

2 **添加新的 VPPA 通信服务。**

例如：

```
# cfgadm -o comm=vppa,service=new -x add_service ib
```

3 **验证是否已添加新的服务。**

例如：

```
# cfgadm -x list_services ib
Port communication services:
                srp
VPPA communication services:
                ibd
                new
HCA_SVC communication services:
                nfs_service
```

▼ 如何删除现有 IB 端口、HCA_SVC 或 VPPA 通信服务

执行以下步骤可删除现有 IB 端口、HCA_SVC 或 VPPA 通信服务。

1 **成为管理员。**

2 **删除 VPPA 通信服务。**

例如：

```
# cfgadm -o comm=vppa,service=new -x delete_service ib
```

3 **验证是否已删除该通信服务。**

例如：

```
# cfgadm -x list_services ib
Port communication services:
                srp
VPPA communication services:
                ibd
HCA_SVC communication services:
                hnfs
```

将 uDAPL 应用程序接口用于 InfiniBand 设备

用户直接访问编程库 (User Direct Access Programming Library, uDAPL) 是标准的 API，可以通过远程直接内存访问 (Remote Direct Memory Access, RDMA) 功能的互连（如 InfiniBand）来提高数据中心应用程序数据消息传送性能、伸缩性和可靠性。uDAPL 接口由 DAT collaborative 定义。有关 DAT collaborative 的信息，请转至以下站点：

http://www.datcollaborative.org

Oracle Solaris 发行版提供了以下 uDAPL 功能：

- 标准 DAT 注册表库 `libdat`。有关更多信息，请参见 `libdat(3LIB)`。
- 标准服务提供者注册文件 `dat.conf`。有关更多信息，请参见 `dat.conf(4)`。
- 支持多个服务提供商，以便每个提供商都在各自的 `service_provider.conf` 文件中指定各自的 uDAPL 库路径、版本号等。有关更多信息，请参见 `service_provider.conf(4)`。
- 管理工具，即 `datadm` 命令，用于配置 `dat.conf`。有关更多信息，请参见 `datadm(1M)`。
- 新的资源控制属性 `project.max-device-locked-memory`，用于控制锁定的物理内存量。
- 使用 IPv4 或 IPv6 地址的命名方案，这些地址利用 IP 基础结构（如 IPv4 中的 ARP 和 IPv6 中的相邻节点搜索）来进行地址解析。Solaris uDAPL 接口适配器可直接映射到 IPoIB 设备实例。
- 支持 DAT collaborative 团体使用的标准地址转换方案。
- uDAPL 服务提供者库，用于支持 tavor 或 hermon 主机通道适配器，该适配器具有向 `dat.conf` 注册文件自动注册的功能。有关更多信息，请参见 `tavor(7D)` 或 `hermon(7D)`。
- 支持 SPARC 平台和 x86 平台。

▼ 如何启用 uDAPL

1 成为管理员。

2 确认是否已安装以下软件包。也可以根据需要安装这些软件包。

- `driver/infiniband/connectx` － InfiniBand 框架
- `driver/infiniband/tavor` － HCA 驱动程序
- `system/io/infiniband/ip-over-ib` － 基于 InfiniBand 的 IP
- `system/io/infiniband/udapl` － 软件包服务提供者

3 创建 IPoIB 接口。

例如：

```
# ipadm create-ip ibd1
# ipadm create-addr -T static -a 192.168.0.1/24 ibd1/ipv4
# datadm -a /usr/share/dat/ABCudaplt.conf
```

更新 DAT 静态注册表

可以使用 datadm 命令来维护 DAT 静态注册表，即 dat.conf 文件。有关更多信息，请参见 dat.conf(4)。

datadm 命令还可用于向 dat.conf 文件注册或注销服务提供者。有关更多信息，请参见 datadm(1M)。

添加或删除 IPoIB 接口适配器时，请运行 datadm 命令来更新 dat.conf 文件，以反映系统的当前状态。此时将重新产生当前安装的所有服务提供商的一组新的接口适配器。

▼ 如何更新 DAT 静态注册表

1. 成为管理员。
2. 在系统中添加或删除 IBoIP 接口适配器后，更新 DAT 静态注册表。

   ```
   # datadm -u
   ```

3. 显示更新的 DAT 静态注册表。

   ```
   # datadm
   ```

▼ 如何在 DAT 静态注册表中注册服务提供者

1. 成为管理员。
2. 添加供应商的 Mellanox 主机通道适配器的服务提供者后，更新 DAT 静态注册表。

   ```
   # datadm -a /usr/share/dat/ABCudaplt.conf
   ```

3. 显示更新的 DAT 静态注册表。

   ```
   # datadm -v
   ```

▼ 如何从 DAT 静态注册表中注销服务提供者

1. 成为管理员。
2. 在系统中删除供应商的 Mellanox 主机适配器的服务提供者后，更新 DAT 静态注册表。

   ```
   # datadm -r /usr/share/dat/ABCudaplt.conf
   ```

3. 显示更新的 DAT 静态注册表。

   ```
   # datadm -v
   ```

管理 IPoIB 设备 (dladm)

缺省情况下，每个 HCA 的各个端口会创建一个物理数据链路。该物理链路可用作管理和观察数据点。可以基于物理数据链路创建 IB 分区链路，与基于 NIC 创建 VNIC 类似。请记住，物理数据链路并非用于数据传输，因此这些链路不支持激活并分配 IB 地址。数据在分区数据链路上传输。

有关配置网络组件的信息，请参见《Oracle Solaris 管理：IP 服务》。

▼ 如何显示物理数据链路信息

使用 `dladm show-phys` 命令在系统中显示物理数据链路信息。物理链路状态与 IB HCA 端口状态直接对应。

1 **成为管理员。**

2 **显示物理数据链路信息。**

例如，显示有关 `ibp0` 一 `ibp3` 数据链路的信息。

```
# dladm show-phys
LINK     MEDIA        STATE      SPEED     DUPLEX     DEVICE
ibp0     Infiniband     up       8000      unknown    ibp0
ibp1     Infiniband    down      8000      unknown    ibp1
ibp2     Infiniband    down      8000      unknown    ibp2
ibp3     Infiniband     up       8000      unknown    ibp3
```

例如，使用 `show-ib` 子命令只会显示命令运行时端口上当前的物理链路、端口 GUID、端口号 HCA GUID 和 `P_Key`。

```
# dladm show-ib
LINK    HCAGUID         PORTGUID          PORT    STATE    P_Key
ibp0    2C9020040041C   2C9020040041D      1        up     FFFF, 8001
ibp1    2C9020040041C   2C9020040041E      2       down    FFFF
ibp2    3BA0001004E14   3BA0001004E15      1       down    FFFF
ibp3    3BA0001004E14   3BA0001004E16      2        up     FFFF, 8001
```

▼ 如何创建 IB 分区链路

IB 分区数据链路可以在 IB 物理链路的顶部创建，与端口上的 `P_Key` 一一对应。分区数据链路用于数据传输。

1 **成为管理员。**

2 **创建新的 IB 分区链路。**

例如，针对 P_Key 0x8001 的 IB 数据链路创建于 ibp0 物理数据链路顶部。

```
# dladm create-part -l ibp0 -P 0x8001 p8001.ibp0
```

如果端口启用且 P_Key 当前存在于该端口，那么上述命令将会成功并且 IPoIB 也会成功初始化。

例如，针对 P_Key 0x9000 的 IB 分区链路创建于 ibp2 顶部。

```
# dladm create-part -f  -l ibp2 -P 0x9000 p9000.ibp2
```

注 – 该端口上没有配置 P_Key 0x9000。强制选项 (-f) 允许您创建 IB 分区链路，即使当前没有 P_Key 或者端口**关闭**。链路状态标记为关闭状态。当 P_Key 添加至该端口且端口激活时，链路状态会转为**开启**。

---

3 **显示 IB 分区链路信息。**

例如：

```
# dladm show-part
LINK            P_Key      OVER       STATE      FLAGS
p8000.ibp0        8001       ibp0       unknown     ----
p9000.ibp2        9000       ibp2       unknown     f---
```

当链路在激活后可以使用时，IB 分区链路状态为**开启**。在以下条件下，分区链路状态为**关闭**：

- HCA 端口处于关闭状态
- P_Key 不存在
- 广播组不存在

4 **激活 IB 地址并指定给 IB 分区链路。**

例如：

```
# ipadm create-ip p9000.ibp2
# ipadm create-addr -T static -a 1.1.1.1 p9000.ibp2/ipv4
# ipadm show-addr
```

▼ 如何显示 IB 分区链路信息

1 **成为管理员。**

2 **显示 IB 分区链路信息。**

例如：

```
# dladm show-part
LINK            P_Key      OVER       STATE      FLAGS
p8000.ibp0        8001       ibp0       unknown     ----
```

```
p9000.ibp2       9000      ibp2      unknown     f---
```

▼ 如何删除 IB 分区链路

1. **成为管理员。**
2. **删除 IB 分区链路。**

   本示例中，分区链路 `p8001.ibp0` 被删除。

   ```
   # dladm delete-part p8001.ibp0
   ```

3. **查看分区链路信息。**

   ```
   # dldam show-part
   LINK        P_Key      OVER     STATE      FLAGS
   p9000.ibp2   9000      ibp2      down       f---
   ```

监视并排除 IB 设备故障

在 Oracle Solaris 11 发行版中，新的命令和实用程序允许您更高效地管理 IB 光纤网络。这些命令涵盖于 `system/io/infiniband/open-fabrics` 软件包中，并且在 `open-fabrics` 软件包安装时会自动安装手册页。例如：

```
% man rping
Reformatting page.  Please Wait... done

librdmacm                                                  RPING(1)

NAME
     rping - RDMA CM connection and RDMA ping-pong test.

SYNOPSIS
     rping -s [-v] [-V] [-d] [-P] [-a address] [-p port]
               [-C message_count] [-S message_size]
     rping -c [-v] [-V] [-d] -a address [-p port]
               [-C message_count] [-S message_size]
.
.
.
```

以下新命令和实用程序使您能够列举并查询 IB 设备，诊断并排除 IB 光纤网络故障，以及测量 IB 性能。

表 9–1 常规 IB 监视命令

命令	说明
`ibv_asyncwatch`	监视 InfiniBand 异步事件

表 9–1 常规 IB 监视命令 （续）

命令	说明
`ibv_devices or ibv_devinfo`	列举 InfiniBand 设备或设备信息
`ibv_rc_pingpong`、`ibv_srq_pingpong` 或 `ibv_ud_pingpong`	使用 RC 连接、SRQ 或 UD 连接测试节点之间的连通性
`mckey`	测试 RDMA CM 多播设置和简单数据传输
`rping`	测试 RDMA CM 连接并尝试 RDMA ping-pong
`ucmatose`	测试 RDMA CM 连接并尝试简单 ping-pong
`udaddy`	测试 RDMA CM 数据报设置并尝试简单 ping-pong

表 9–2 常规 IB 性能测试命令

命令	说明
`rdma_bw or rdma_lat`	测试 RDMA 写处理确定流带宽或等待时间
`ib_read_bw` 或 `ib_read_lat`	测试 RDMA 读处理确定带宽或等待时间
`ib_send_bw` 或 `ib_send_lat`	测试 RDMA 发送处理确定带宽或等待时间
`ib_write_bw` 或 `ib_write_bw_postlist`	测试 RDMA 写处理，确定一次显示一个 I/O 请求的带宽或显示一系列 I/O 请求的发布列表带宽
`ib_write_lat`	测试 RDMA 写处理确定等待时间
`ib_clock_test`	测试系统时钟准确性
`qperf`	测量插槽与 RDMA 性能

表 9–3 RDS 监视与测试工具

命令	说明
`rds-info`	显示 RDS 内核模块信息
`rds-ping`	确定基于 RDS 的远程节点是否可访问
`rds-stress`	在基于 RDS 插槽的进程间发送消息

表 9–4 光纤网络诊断工具

命令	说明
`ibdiagnet`	执行整个光纤网络诊断检查
`ibaddr`	查询 InfiniBand 的一个地址或多个地址
`ibnetdiscover`	搜索远程 InfiniBand 拓扑

表 9–4 光纤网络诊断工具 （续）

命令	说明
ibping	验证 IB 节点之间的连通性
ibportstate	查询 IB 端口的物理端口状态和链接速度
ibroute	显示 InfiniBand 交换机转发表
ibstat 或 ibsysstat	查询 InfiniBand 设备状态或 IB 地址上的系统状态
ibtracert	跟踪 IB 路径
perfquery or saquery	查询 IB 端口计数器或 sIB 子网管理属性
sminfo	查询 IB SMInfo 属性
smpquery 或 smpdump	查询或转储 IB 子网管理属性
ibcheckerrors 或 ibcheckerrs	验证 IB 端口（或节点）或 IB 子网并报告错误
ibchecknet, ibchecknode, 或 ibcheckport	验证 IB 子网、节点或端口并报告错误
ibcheckportstate, ibcheckportwidth, ibcheckstate, or ibcheckwidth	验证已链接但不活动的 IB 端口、面向 1x (2.0 Gbps) 链路带宽的端口、IB 子网中已链接但不活动的端口或 IB 子网中的 lx 链路
ibclearcounters or ibclearerrors	对 IB 子网中的端口计数器或错误计数器进行清零
ibdatacounters or ibdatacounts	查询 IB 子网中的数据计数器或 IB 端口数据计数器
ibdiscover.pl	注释并比较 IB 拓扑
ibhosts	显示拓扑中的 IB 主机节点
iblinkinfo.pl 或 iblinkinfo	显示光纤网络中所有链路的链路信息
ibnodes	显示拓扑中的 IB 节点
ibprintca.pl	显示来自 ibnetdiscover 输出的特定 CA 或 CA 列表
ibprintrt.pl	显示来自 ibnetdiscover 输出的特定路由器或路由器列表
ibprintswitch.pl	显示来自 ibnetdiscover 输出的特定交换机或交换机列表
ibqueryerrors.pl	查询并报告非零 IB 端口计数器
ibrouters	显示拓扑中的 IB 路由器节点
ibstatus	查询 IB 设备的基本状态
ibswitches	显示拓扑中的 IB 交换机节点
ibswportwatch.pl	轮询特定交换机或端口上的计数器并报告更改信息速率

表 9–4 光纤网络诊断工具 （续）

命令	说明
set_nodedesc.sh	设置或显示针对 IB 主机控制器适配器 (HCA) 的节点描述字符串
dump2psl.pl	转储基于 opensm 输出文件的 PSL 文件，该输出文件用于信用循环检查
dump2slvl.pl	转储基于 opensm 输出文件的 SLVL 文件，该输出文件用于信用循环检查
ibis	针对 IB 管理带内服务的扩展 TCL shell

◆ ◆ ◆ 第 10 章

10 管理磁盘（概述）

本章提供有关 Oracle Solaris 磁盘分片的概述信息并介绍 `format` 实用程序。

以下是本章中概述信息的列表。

- 第 171 页中的 “在磁盘管理方面的新增功能”
- 第 177 页中的 “有关磁盘管理任务的参考信息”
- 第 177 页中的 “磁盘管理概述”
- 第 177 页中的 “磁盘术语”
- 第 178 页中的 “关于磁盘标签”
- 第 181 页中的 “关于磁盘分片”
- 第 182 页中的 “`format` 实用程序”
- 第 184 页中的 “对磁盘分区”

有关如何将磁盘添加到系统的说明，请参见第 12 章，SPARC：设置磁盘（任务）或第 13 章，x86：设置磁盘（任务）。

在磁盘管理方面的新增功能

本节介绍 Oracle Solaris 发行版中新增的磁盘管理功能。

- 第 172 页中的 “通过物理位置标识设备”
- 第 175 页中的 “支持在 2 TB 磁盘上安装和引导 Oracle Solaris OS”
- 第 175 页中的 “Solaris iSCSI 目标和启动器中的 iSNS 支持”
- 第 176 页中的 “Solaris COMSTAR iSCSI 支持”
- 第 176 页中的 “x86: GRUB 引导环境中的磁盘管理”
- 第 176 页中的 “支持容量超过 2 TB 的 SCSI 磁盘”

通过物理位置标识设备

Oracle Solaris 11：在该发行版中，/dev/chassis 目录提供包括物理位置的设备名称。可以使用该信息帮您确定设备需要更换或改变时它们的物理位置如何。可以使用以下命令按机箱、插口和插卡的值显示系统上设备的信息：

- diskinfo－使用此命令显示物理磁盘位置的一般信息
- format－使用该命令显示查看分区表或重新设置标签时磁盘的物理磁盘位置信息
- prtconf -l－使用该命令显示包括物理磁盘位置信息的系统配置信息
- zpool status -l－使用该命令显示池设备的物理磁盘位置信息

此外，可以使用 fmadm add-alias 命令来引入磁盘别名，磁盘别名可帮助您识别磁盘在您的环境中的物理位置。例如：

```
# fmadm add-alias SUN-Storage-J4200.0912QAJ001 J4200@RACK10:U26-27
# fmadm add-alias SUN-Storage-J4200.0905QAJ00E J4200@RACK10:U24-25
```

然后使用上述命令显示这些别名，从而显示物理磁盘位置信息。例如：

```
$ diskinfo
D:devchassis-path                                    c:occupant-compdev
---------------------------------------------------  ------------------
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__0/disk   c1t13d0
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__1/disk   c1t14d0
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__2/disk   c1t2d0
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__3/disk   c1t3d0
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__4/disk   c1t15d0
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__5/disk   c1t16d0
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__6/disk   c1t6d0
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__7/disk   c1t7d0
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__8/disk   c1t17d0
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__9/disk   c1t18d0
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__10/disk  c1t10d0
```

确定特定磁盘的位置：

```
% diskinfo -c c6t11d0
D:devchassis-path                                          c:occupant-compdev
---------------------------------------------------------  ------------------
/dev/chassis/SUN-Storage-J4400/SCSI_Device__11/disk        c6t11d0
```

在此示例中，/dev/chassis 磁盘名称包括了一个别名，这可帮助您确定设备在您的环境中的位置。

以下 diskinfo 示例说明了如何显示特定磁盘的物理位置。

```
$ diskinfo -c c6t11d0 -o cp
c:occupant-compdev  p:occupant-paths
------------------  ----------------------------------------------------
c6t11d0             /devices/pci@0,0/pci8086,3604@1/pci1000,3150@0/sd@b,0
```

如果想要确定系统中特定类型的磁盘数，可以使用类似于以下内容的 diskinfo 语法：

```
$ diskinfo -n SEAGATE ST31000N-SU0B-931.51GB -o Dcf
D:devchassis-path                                  t:occupant-type  c:occupant-compdev
-------------------------------------------------  ---------------  ------------------
/dev/chassis/colab5@RACK10_24-25/SCSI_Device__0/disk   disk             c0t13d0
/dev/chassis/colab5@RACK10_24-25/SCSI_Device__1/disk   disk             c0t14d0
/dev/chassis/colab5@RACK10_24-25/SCSI_Device__2/disk   disk             c0t2d0
/dev/chassis/colab5@RACK10_24-25/SCSI_Device__3/disk   disk             c0t1d0
/dev/chassis/colab5@RACK10_24-25/SCSI_Device__4/disk   disk             c0t15d0
/dev/chassis/colab5@RACK10_24-25/SCSI_Device__5/disk   disk             c0t16d0
/dev/chassis/colab5@RACK10_24-25/SCSI_Device__6/disk   disk             c0t6d0
/dev/chassis/colab5@RACK10_24-25/SCSI_Device__7/disk   disk             c0t7d0
/dev/chassis/colab5@RACK10_24-25/SCSI_Device__8/disk   disk             c0t17d0
/dev/chassis/colab5@RACK10_24-25/SCSI_Device__9/disk   disk             c0t18d0
/dev/chassis/colab5@RACK10_24-25/SCSI_Device__10/disk  disk             c0t10d0
```

注 – `diskinfo` 命令要求机箱支持 SES 诊断页 0xa（其他元素状态）并且必须将元素当前索引 (Element Index Present, EIP) 位设置为 1。不满足此条件的附件将无法完全枚举，因此也无法正确地表示。

`format` 命令已得到更新，可提供物理设备位置信息。例如：

```
# format
.
.
.
       18. c0t4d0 <SEAGATE-ST345056SSUN450G-081C-419.19GB>
           /pci@0,600000/pci@0/pci@9/LSILogic,sas@0/sd@4,0
           /dev/chassis/colab5@RACK10_26-27/SCSI_Device__6/disk
       19. c0t27d0 <ATA-SEAGATE ST35000N-3AZQ-465.76GB>
           /pci@0,600000/pci@0/pci@9/LSILogic,sas@0/sd@1b,0
           /dev/chassis/colab5@RACK10_26-27/SCSI_Device__7/disk
       20. c0t23d0 <ATA-SEAGATE ST31000N-SU0B-931.51GB>
           /pci@0,600000/pci@0/pci@9/LSILogic,sas@0/sd@17,0
           /dev/chassis/colab5@RACK10_26-27/SCSI_Device__8/disk
       21. c0t24d0 <ATA-SEAGATE ST31000N-SU0B-931.51GB>
           /pci@0,600000/pci@0/pci@9/LSILogic,sas@0/sd@18,0
           /dev/chassis/colab5@RACK10_26-27/SCSI_Device__9/disk
```

使用 `prtconf -l` 显示物理设备位置信息。例如：

```
$ prtconf -l | more
System Configuration:  Oracle Corporation  sun4v
Memory size: 32640 Megabytes
System Peripherals (Software Nodes):

SUNW,SPARC-Enterprise-T5220 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__0/disk
.
.
.
pci, instance #15 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__0/disk
     LSILogic,sas, instance #1 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__0/disk
        smp, instance #0 (driver not attached)
        sd, instance #2 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__0/disk
        sd, instance #4 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__1/disk
```

```
        sd, instance #5 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__2/disk
        sd, instance #6 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__3/disk
        sd, instance #7 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__4/disk
        sd, instance #8 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__5/disk
        sd, instance #9 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__6/disk
        sd, instance #10 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__7/disk
        sd, instance #11 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__8/disk
        sd, instance #12 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__9/disk
        sd, instance #13 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__10/disk
        sd, instance #14 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__11/disk
```

使用 `zpool status -l` 选项显示物理设备位置信息。例如：

```
% zpool status -l export
  pool: export
 state: ONLINE
 scan: resilvered 379G in 8h31m with 0 errors on Thu Jan 27 23:10:20 2011
config:

        NAME                                                STATE     READ WRITE CKSUM
        export                                              ONLINE       0     0     0
          mirror-0                                          ONLINE       0     0     0
            /dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__2/disk   ONLINE       0     0     0
            /dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__3/disk   ONLINE       0     0     0
          mirror-1                                          ONLINE       0     0     0
            /dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__4/disk   ONLINE       0     0     0
            /dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__5/disk   ONLINE       0     0     0
          mirror-2                                          ONLINE       0     0     0
            /dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__6/disk   ONLINE       0     0     0
            /dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__7/disk   ONLINE       0     0     0
          mirror-3                                          ONLINE       0     0     0
            /dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__8/disk   ONLINE       0     0     0
            /dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__9/disk   ONLINE       0     0     0
          mirror-4                                          ONLINE       0     0     0
            /dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__10/disk  ONLINE       0     0     0
            /dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__11/disk  ONLINE       0     0     0
        spares
          /dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__0/disk     AVAIL
          /dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__1/disk     AVAIL

errors: No known data errors
```

多磁盘扇区容量支持

Oracle Solaris 11：之前的 Solaris 发行版支持 512 字节的磁盘扇区容量。在该发行版中，支持 512 字节、1024 字节、2048 字节或 4096 字节的扇区容量的磁盘。具有更大扇区的磁盘提供更大的磁盘容量、更高的可靠性以及更高效的数据传送和更快的驱动程序维护。

在此 Oracle Solaris 发行版中，可在大扇区磁盘上使用的唯一受支持文件系统是非根 ZFS 文件系统。有关使用大扇区磁盘作为 COMSTAR 目标的更多信息，请参见第 14 章，使用 COMSTAR 配置存储设备。

目前不支持从大扇区磁盘引导和安装的功能。

支持在 2 TB 磁盘上安装和引导 Oracle Solaris OS

Oracle Solaris 11：在之前的 Solaris 发行版中，无法从容量超过 1 TB 的磁盘中安装和引导 Solaris OS。在本 Solaris 发行版中，可以从容量高达 2 TB 的磁盘中安装和引导 Solaris OS。在之前的发行版中，还必须对超过 1 TB 的磁盘使用 EFI 标签。在本发行版中，可以在任何容量的磁盘上使用 VTOC 标签，但是 VTOC 的可寻址空间限制在 2 TB 内。

Oracle Solaris 磁盘驱动器和磁盘实用程序已得到更新，可提供以下支持：

- 在 2 TB 磁盘上安装和引导 Oracle Solaris OS 必须连接至内存至少为 1.5 GB 的系统。
- 可以使用 `format -e` 实用程序给任何容量的磁盘设置 VTOC 标签，但是可寻址空间限制在 2 TB 以内。
- `format` 实用程序以及容量小于 2 TB 的磁盘的安装软件使用的缺省标签为 VTOC 标签。
- 可以在 x86 系统中容量超过 1 TB 的磁盘上使用 `fdisk` 实用程序。还增加对非 EFI 分区类型的 MBR 中高达 2 TB 分区的支持。该支持表明 Solaris 分区可高达 2 TB。其他非 EFI 分区可能受到不同分区类型的限制。

  `fdisk` 实用程序运行在容量超过 2 TB 的磁盘上时，将显示一条警告消息，指出您无法创建超过 2 TB 的非 EFI 分区。

请记住，您无法将超过 1 TB 的带有传统 MBR 或传统 VTOC 的磁盘移动至先前的 Solaris 发行版。带有 EFI 标签的磁盘仍能运行在先前的 Solaris 发行版中。

有关该发行版中 EFI 标签变更的更多信息，请参见第 178 页中的 “EFI 磁盘标签”。

Solaris iSCSI 目标和启动器中的 iSNS 支持

Oracle Solaris 11：该 Solaris 发行版支持 Solaris iSCSI 目标和启动器软件中的 Internet 存储名称服务 (Internet Storage Name Service, iSNS) 协议。iSNS 协议允许自动搜索、管理和配置 TCP/IP 网络中的 iSCSI 设备。

请参见以下资源的逐步说明：

- 有关配置 Oracle Solaris 目标以使用第三方 iSNS 服务器的信息，请参见第 14 章，使用 COMSTAR 配置存储设备。
- 有关配置具有 Solaris iSNS 服务器的 Solaris iSCSI 目标的信息，请参见第 15 章，配置和管理 Oracle Solaris Internet 存储名称服务 (Internet Storage Name Service, iSNS)和`isnsadm(1M)`。

Solaris COMSTAR iSCSI 支持

Oracle Solaris 11：iSCSI 是一个基于 Internet 协议 (Internet Protocol, IP) 的存储联网标准，用于链接数据存储子系统。通过跨 IP 网络传输 SCSI 命令，iSCSI 协议允许您通过网络将磁盘设备挂载到本地系统上。在本地系统上，可以将这些设备作为块设备来使用。

常见多协议 SCSI 目标或 COMSTAR（一种软件框架）使您能将任何 Oracle Solaris 11 主机转换成 SCSI 目标设备，该设备可以通过启动器主机在存储网络上进行访问。

有关更多信息，请参见第 14 章，使用 COMSTAR 配置存储设备。

x86: GRUB 引导环境中的磁盘管理

Oracle Solaris 11：GRUB 引导菜单替换了以前的 x86 系统引导方法。在磁盘管理方面，从备用设备引导以替换系统磁盘或安装引导块时，可以使用 GRUB 界面。

GRUB 引导环境提供了以下功能：

- **网络引导**－通过在 BIOS 配置阶段按 F12 键从网络进行引导。
- **单用户引导**－通过从 Solaris 故障安全引导菜单中选择此选项将系统引导至单用户模式：

  ```
  kernel /platform/i86pc/multiboot
  ```

  然后，使用 e（编辑）选项添加 -s 单用户选项。例如：

  ```
  kernel /platform/i86pc/multiboot -s
  ```

  按返回键，然后按 b 键来引导系统。按 Ctrl-D 组合键，将系统引导回多用户模式。

如果运行的是 x86 系统，则在 GRUB 环境中不能使用 `fmthard` 命令自动安装引导块。必须单独安装引导块。

有关详细的功能信息以及在 x86 系统上使用基于 GRUB 的新引导的说明，请参见《在 x86 平台上引导和关闭 Oracle Solaris》。

此功能在 SPARC 系统上不可用。

支持容量超过 2 TB 的 SCSI 磁盘

Oracle Solaris 11：SCSI 驱动器、`ssd` 或 `sd` 支持 2 TB 以及以上的容量。SCSI 驱动器、`ssd` 或 `sd` 在之前的 Solaris 发行版中限制在 2 TB 以内。

可以使用 `format` 实用程序对这些较大的磁盘进行设置标签、配置和分区。有关在大磁盘上使用 EFI 磁盘标签以及使用 `fdisk` 实用程序的限制的信息，请参见第 179 页中的 “EFI 磁盘标签限制”。

有关磁盘管理任务的参考信息

使用这些参考信息可以查找有关管理磁盘的逐步说明。

磁盘管理任务	更多信息
格式化磁盘并检查磁盘标签。	第 11 章，管理磁盘（任务）
向 SPARC 系统中添加新磁盘。	第 12 章，SPARC：设置磁盘（任务）
向 x86 系统中添加新磁盘。	第 13 章，x86：设置磁盘（任务）
热插拔 SCSI 或 PCI 磁盘。	第 6 章，动态配置设备（任务）

磁盘管理概述

在 Oracle Solaris OS 中管理磁盘通常涉及设置系统并运行 Oracle Solaris 安装程序，以创建适当的磁盘分片和文件系统并安装 Oracle Solaris OS。有时，可能需要使用 `format` 实用程序添加新磁盘驱动器或更换有故障的磁盘驱动器。

磁盘术语

您应熟悉基本的磁盘体系结构，这样才能有效地使用本节中的信息。需要特别指出的是，应熟悉以下术语：

磁盘术语	说明
Track（磁轨）	磁盘上的同心环，在磁盘旋转时在单个固定磁头下穿过。
Cylinder（柱面）	距磁盘旋转所围绕的轴的距离相同的一组磁轨。
Sector（扇区）	每个磁盘片的段。一个扇区能容纳 512 字节。
Block（块）	磁盘上的数据存储区域。磁盘块为 512 字节。
Disk controller（磁盘控制器）	用于控制磁盘驱动器的芯片及其关联的电路。
Disk label（磁盘标签）	磁盘中包含磁盘几何参数和分区信息的第一个扇区。
Device driver（设备驱动程序）	控制硬件或虚拟设备的内核模块。

有关其他信息，请参见磁盘制造商提供的产品信息。

关于磁盘标签

专门为每个磁盘保留的一个特殊区域，用于存储有关磁盘的控制器、几何参数和分片的信息。此信息称为磁盘的**标签**。用于描述磁盘标签的另一个术语是带有 VTOC 标签的磁盘上的 *VTOC*（**卷目录**）。为磁盘**设置标签**意味着将分片信息写入磁盘。更改磁盘分片后，通常要为磁盘设置标签。

Oracle Solaris 发行版支持以下两种磁盘标签：

- SMI－用于容量小于 2 TB 的磁盘的传统 VTOC 标签。
- EFI－支持容量超过 2 TB 的磁盘。可扩展固件接口 GUID 分区表 (Extensible Firmware Interface GUID Partition Table, EFI GPT) 磁盘标签还适用于小于 2 TB 的磁盘。

如果创建分片后未能为磁盘设置标签，则无法使用分片，因为 OS 不能识别这些分片。

EFI 磁盘标签

EFI 标签支持物理磁盘和容量超过 2 TB 的虚拟磁盘。该发行版还包括用于管理容量超过 2 TB 磁盘的更新磁盘实用程序。

以下文件系统产品支持容量超过 1 TB 的文件系统：

- Oracle Solaris ZFS 文件系统支持容量超过 1 TB 的文件系统。

VTOC 标签仍适用于容量小于 2 TB 的磁盘。如果仅在系统中使用小于 2 TB 的磁盘，则如先前的 Solaris 发行版中所述管理磁盘。另外，可以使用 `format -e` 命令给容量小于等于 2 TB 的磁盘设置 EFI 标签。有关更多信息，请参见示例 11–5。

如果系统正在运行适当的 Oracle Solaris 发行版，则可以使用 `format -e` 命令给磁盘设置 EFI 标签。然而，应在尝试设置 EFI 标签之前，查看第 179 页中的 “EFI 磁盘标签限制”中的重要信息。

如果不再需要 EFI 标签，还可以使用 `format -e` 命令重新设置 VTOC 标签。例如：

```
# format -e
Specify disk (enter its number): 2
selecting c0t5d0
[disk formatted]
.
.
.
format> label
[0] SMI Label
[1] EFI Label
Specify Label type[1]: 0
Warning: This disk has an EFI label. Changing to SMI label will erase all
current partitions.
Continue? yes
Auto configuration via format.dat[no]?
Auto configuration via generic SCSI-2[no]?
format> quit
```

注意 – 请记住，改变磁盘标签会损坏磁盘中的所有数据。

比较 EFI 标签与 VTOC 标签

在以下方面 EFI 磁盘标签不同于 VTOC 磁盘标签：

- 支持容量超过 2 TB 的磁盘。
- 提供可用的分片 0–6，其中仅分片 2 是不同的分片。
- 分区（或分片）不能与主标签或备份标签重叠，也不能与任何其他分区重叠。EFI 标签的大小通常为 34 个扇区，因此分区通常从第 34 个扇区开始。此特性意味着任何分区都不能从扇区零 (0) 开始。
- EFI 标签中不存储柱面信息。大小以块数进行报告。
- 在备用柱面区域（磁盘的最后两个柱面）中存储的信息现在存储在分片 8 中。
- 如果使用 `format` 实用程序来更改分区大小，则 `unassigned` 分区标记将被指定给大小等于零的分区。缺省情况下，`format` 实用程序将 `usr` 分区标记指定给大小大于零的任何分区。更改分区后，可以使用分区更改菜单重新指定分区标记。但是，不能将大小不为零的分区更改为 `unassigned` 分区标记。

EFI 磁盘标签限制

确定使用容量超过 1 TB 的磁盘是否适合于环境时，请记住以下限制：

- 如果系统的磁盘使用 VTOC 标签，则专门为该系统设计的分层软件产品可能无法访问使用 EFI 磁盘标签的磁盘。
- 运行以前的 Solaris 发行版的系统不能识别带有 EFI 标签的磁盘。
- 无法从带有 EFI 磁盘标签的磁盘进行引导。
- 基于 x86 的系统中，在容量大于 2 TB 且带有 EFI 标签的磁盘上，可以使用 `fdisk` 命令。
- 使用 `format` 实用程序对带有 EFI 标签的磁盘进行分区。
- EFI 规范禁止重叠分片。整个磁盘以 *cxtydz* 表示。
- EFI 磁盘标签以扇区和块（而不是柱面和磁头）为单位提供磁盘或分区大小的信息。
- 在带有 EFI 标签的磁盘上不支持或不适用以下 `format` 选项：
  - 不支持 `save` 选项，因为带有 EFI 标签的磁盘不需要在 `format.dat` 文件中包含项。
  - 不适用 `backup` 选项，因为磁盘驱动器会发现主标签，并将其写回磁盘。

支持 x86 系统上带有 EFI 标签的磁盘

在 x86 系统中提供对 EFI 磁盘标签的 Oracle Solaris 支持。在 x86 系统上可以使用以下命令添加 EFI 标签：

```
# format -e
> [0] SMI Label
> [1] EFI Label
> Specify Label type[0]: 1
> WARNING: converting this device to EFI labels will erase all current
> fdisk partition information. Continue? yes
```

以前的标签信息不会转换为 EFI 磁盘标签。

必须使用 `format` 命令手动重新创建标签的分区信息。在容量为 2 TB 且带有 EFI 标签的磁盘上，不能使用 `fdisk` 命令。如果 `fdisk` 命令运行在容量超过 2 TB 的磁盘上以创建 Solaris 分区，则 Solaris 分区的大小限制在 2 TB 以内。有关 EFI 磁盘标签的更多信息，请参见上一节。

使用带有 EFI 标签的磁盘安装系统

Oracle Solaris 根池磁盘必须具有 SMI 标签。Oracle Solaris 11 安装实用程序自动对选为具有 SMI 标签的根池磁盘的磁盘重新设置标签。

管理带有 EFI 磁盘标签的磁盘

使用下表查找有关管理带有 EFI 磁盘标签的磁盘的信息。

任务	更多信息
如果已安装系统，请将磁盘连接至系统。	第 205 页中的 “SPARC: 为 ZFS 文件系统设置磁盘（任务列表）”或第 213 页中的 “x86: 为 ZFS 文件系统设置磁盘（任务列表）”
创建 ZFS 文件系统。	第 206 页中的 “SPARC: 为 ZFS 文件系统设置磁盘”或第 214 页中的 “x86: 为 ZFS 文件系统设置磁盘”

EFI 磁盘标签问题的疑难解答

使用以下错误消息和解决方法解决带有 EFI 标签的磁盘问题。

解决方法

使用容量大于 1 TB 的磁盘引导运行 SPARC 或 x86 内核的系统。

错误消息

```
Dec  3 09:12:17 holoship scsi: WARNING: /sbus@a,0/SUNW,socal@d,10000/
sf@1,0/ssd@w50020f23000002a4,0 (ssd1):
Dec  3 09:12:17 holoship corrupt label - wrong magic number
```

原因
　您尝试向运行早期 Solaris 发行版的系统中添加磁盘。

解决方法
　向运行 Solaris 发行版且支持 EFI 磁盘标签的系统中添加磁盘。

关于磁盘分片

文件系统中包含存储在磁盘上的文件。磁盘上的每个文件系统都被指定给**分片**，分片是专门保留供该文件系统使用的一组扇区。对于 Oracle Solaris OS（和系统管理员）而言，每个磁盘分片看起来就如同单独的磁盘驱动器一样。

有关文件系统的信息，请参见第 17 章，管理文件系统（概述）。

注 – 分片有时称为**分区**。某些界面（如 `format` 实用程序）将分片称为分区。

设置分片时，请记住以下规则：

- 每个磁盘分片仅容纳一个文件系统。
- 任何文件系统都不能跨多个分片。

分片在 SPARC 和 x86 平台上的设置方式略有不同。下表总结了这些差异。

表 10–1 SPARC 和 x86 平台上的分片差异

SPARC 平台	x86 平台
整个磁盘供 Oracle Solaris OS 专用。	磁盘被分成 `fdisk` 分区，每个操作系统都有一个 `fdisk` 分区。
VTOC－磁盘被分成 8 个分片，编号为 0–7。 **EFI**－磁盘被分成 7 个分片，编号为 0–6。	**VTOC**－Solaris `fdisk` 分区被分成 10 个分片，编号为 0–9。 **EFI**－磁盘被分成 7 个分片，编号为 0–6。

磁盘分片

在具有 ZFS 文件系统的系统上，只有含有 ZFS 根文件系统的 ZFS 根池才需要磁盘分片。一般而言，ZFS 根池包含在分片 0 中。由于长久引导限制，ZFS 根池必须位于分片或镜像分片中。对于非根存储池，可以使用整个磁盘。ZFS 文件系统不对应特定的磁盘分片。管理整个磁盘比管理带有分片的磁盘简单。

在 x86 系统上：

- 磁盘被分成 `fdisk` 分区。`fdisk` 分区是为特定操作系统（如 Oracle Solaris OS）保留的磁盘段。
- Oracle Solaris OS 将十个分片（编号为 0–9）置于 Solaris `fdisk` 分区上。

注 – 在带有 VTOC 标签的磁盘上，不得修改分片或使用分片 2 存储文件系统。如果分片 2 以任何形式进行修改，`installgrub` 命令不能正确运行。

使用原始数据分片

磁盘标签存储在每个磁盘的块 0 中。因此，用于创建原始数据分片的第三方数据库应用程序决不能从块 0 启动。否则，将覆盖磁盘标签，且磁盘上的数据将无法访问。

请勿对原始数据分片使用以下磁盘区域，这些原始数据分片有时是由第三方数据库应用程序创建的：

- 存储磁盘标签的块 0
- 分片 2，表示带有 VTOC 标签的整个磁盘

format 实用程序

继续学习操作方法或参考部分之前，请先阅读以下 `format` 实用程序及其用法的概述。

`format` 实用程序是系统管理工具，用于准备要在 Oracle Solaris 系统上使用的硬盘驱动器。

下表说明了 `format` 实用程序提供的功能和关联优势。

表 10–2 format 实用程序的功能和优势

功能	优势
搜索系统中所有连接的磁盘驱动器	报告以下内容： ■ 目标位置 ■ 磁盘几何参数 ■ 磁盘是否已格式化 ■ 磁盘是否具有已挂载的分区
检索磁盘标签	便于修复操作
修复有故障的扇区	允许管理员修复具有可恢复错误的磁盘驱动器，而不用将驱动器发回给制造商
格式化和分析磁盘	在磁盘上创建扇区并验证每个扇区
对磁盘分区	将磁盘分成分片。ZFS 文件系统不对应磁盘分片，ZFS 根池除外。
为磁盘设置标签	将磁盘名称和配置信息写入磁盘，以备将来检索（通常用于修复操作）

`format` 实用程序选项在第 16 章，`format` 实用程序（参考）予以介绍。

何时使用 format 实用程序

安装 Oracle Solaris 发行版时，Oracle Solaris 安装实用程序会对磁盘驱动器进行分区并设置标签。可以使用 format 实用程序执行以下操作：

- 显示分片信息
- 对磁盘分区
- 向现有系统添加磁盘驱动器
- 格式化磁盘驱动器
- 为磁盘设置标签
- 修复磁盘驱动器
- 分析磁盘错误

系统管理员使用 format 实用程序的主要目的是对磁盘分区。这些步骤将在第 12 章，SPARC：设置磁盘（任务）和第 13 章，x86：设置磁盘（任务）中介绍。

有关使用 format 实用程序的指南，请参见下一节。

format 实用程序使用指南

表 10–3 format 实用程序指南

任务	指南	更多信息
格式化磁盘。	■ 重新格式化磁盘时，任何现有数据都将被销毁。 ■ 随着越来越多的制造商提供经过格式化和分区的磁盘驱动器，对磁盘驱动器格式化的需求有所下降。在现有系统中添加或更换磁盘驱动器时，可能不需要使用 format 实用程序。 ■ 如果磁盘已重定位且显示许多磁盘错误，则可以尝试重新为该磁盘设置标签。	第 193 页中的 “如何格式化磁盘”或第 196 页中的 “如何为磁盘设置标签”
设置含有 ZFS 根文件系统的磁盘。	在非冗余配置中，受损磁盘中的 ZFS 根文件系统数据必须从备份介质中进行存储。否则，必须使用安装实用程序重新安装系统。	第 206 页中的 “SPARC: 如何为 ZFS 根文件系统设置磁盘”或第 214 页中的 “x86: 如何为 ZFS 根文件系统设置磁盘”，或者（如果必须重新安装系统）《安装 Oracle Solaris 11 系统》
为根池创建磁盘分片。	■ 使用 ZFS 存储池的最佳方式是创建具有整个磁盘的池。 ■ 如果磁盘预期用于根池，则必须创建磁盘分片。这是长久引导限制。	第 207 页中的 “SPARC: 如何为 ZFS 根文件系统创建磁盘分片”或第 215 页中的 “x86: 如何为 ZFS 根文件系统创建磁盘分片”

表 10–3 format 实用程序指南 （续）

任务	指南	更多信息
设置含有 ZFS 文件系统的磁盘。	用于非根 ZFS 文件系统的磁盘通常含有用户或数据文件的空间。为了获得更多磁盘空间，可以向根池或非根池附加或添加另一个磁盘。	第 211 页中的 “SPARC: 如何为 ZFS 文件系统设置磁盘”或 第 220 页中的 “x86: 如何为 ZFS 文件系统设置磁盘”

格式化磁盘

在大多数情况下，磁盘由制造商或转售商来格式化。因此，安装驱动器时，不需要重新格式化磁盘。要确定磁盘是否已格式化，请使用 `format` 实用程序。有关更多信息，请参见第 192 页中的 “如何确定磁盘是否已格式化”。

如果确定磁盘未格式化，请使用 `format` 实用程序来格式化磁盘。

格式化磁盘时，需要完成两个步骤：

- 准备好磁盘介质以便随时使用。
- 根据表面分析编译磁盘缺陷列表。

注意 – 格式化磁盘是一个破坏性过程，因为它覆盖磁盘上的数据。由于此原因，磁盘通常仅由制造商或转售商进行格式化。如果认为磁盘缺陷是导致问题反复出现的原因，则可以使用 `format` 实用程序执行表面分析。但是，应非常谨慎地仅使用不会破坏数据的命令。有关详细信息，请参见第 193 页中的 “如何格式化磁盘”。

用于存储数据的总磁盘空间中有一小部分用于存储缺陷和格式化信息。此部分的百分比随磁盘几何参数的变化而变化，而且会因磁盘使用年限增加，产生的缺陷增多，而不断增大。

格式化磁盘可能需要几分钟到几小时时间，具体取决于磁盘类型和大小。

对磁盘分区

系统管理员经常使用 `format` 实用程序来对磁盘分区。步骤如下：

- 确定所需的分片
- 确定每个分片或分区的大小
- 使用 `format` 实用程序对磁盘分区
- 使用新的分区信息为磁盘设置标签
- 为每个分区创建文件系统

最简单的磁盘分区方法是使用 `format` 实用程序分区菜单中的 `modify` 命令。借助 `modify` 命令，通过指定每个分区的大小可以创建分区，而不必跟踪起始柱面边界。`modify` 命令还可用于跟踪“浮动 (free hog)”分片中的任何磁盘空间。

分区表术语

磁盘标签的一个重要部分是**分区表**。分区表标识磁盘分片、分片边界（在柱面中）以及分片的总大小。可以使用 format 实用程序来显示磁盘的分区表。下面介绍了分区表术语。

表 10–4 分区表术语

分区术语	值	说明
数字	0–7	**VTOC**－分区或分片，编号为 0–7。 **EFI**－分区或分片，编号为 0–6。
Tag（标记）	0=UNASSIGNED 1=BOOT 2=ROOT 3=SWAP 4=USR 5=BACKUP 7=VAR 8=HOME 11=RESERVED	通常用于描述此分区中挂载的文件系统的数值。
Flag（标志）	wm	分区是可写且可挂载的。
	wu rm	分区可写，但不能取消挂载。此状态是专用于交换区域的分区的缺省状态。（但是，mount 命令不会检查“不可挂载”标志。）
	rm	分区是只读且可挂载的。

分区标志和标记按约定指定且不需要进行维护。

有关显示分区表的更多信息，请参见以下参考信息：

- 第 185 页中的 “显示分区表信息”
- 第 194 页中的 “如何显示磁盘分片信息”
- 第 200 页中的 “如何检查磁盘标签”

显示分区表信息

以下 format 实用程序输出给出了带有 VTOC 标签的 74 GB 磁盘中的分区表示例：

```
Total disk cylinders available: 38756 + 2 (reserved cylinders)

Part      Tag    Flag     Cylinders         Size            Blocks
  0       root    wm       3 -  2083        4.00GB    (2081/0/0)    8390592
  1       swap    wu    2084 -  3124        2.00GB    (1041/0/0)    4197312
  2     backup    wm       0 - 38755       74.51GB    (38756/0/0) 156264192
  3 unassigned    wm       0                0         (0/0/0)             0
  4 unassigned    wm       0                0         (0/0/0)             0
  5 unassigned    wm       0                0         (0/0/0)             0
  6 unassigned    wm       0                0         (0/0/0)             0
  7       home    wm    3125 - 38755       68.50GB    (35631/0/0) 143664192
  8       boot    wu       0 -     0        1.97MB    (1/0/0)          4032
```

```
  9 alternates     wu       1 -     2        3.94MB    (2/0/0)          8064

partition>
```

通过 format 实用程序显示的分区表包含以下信息。

列名	说明
Part	分区编号或分片编号。有关此列的说明，请参见表 10–4。
Tag	分区标记。有关此列的说明，请参见表 10–4。
Flag	分区标志。有关此列的说明，请参见表 10–4。
Cylinders	分片的起始和结束柱面编号。不显示在带有 EFI 标签的磁盘上。
Size	分片大小 (MB)。
Blocks	每个分片的柱面总数和扇区总数。不显示在带有 EFI 标签的磁盘上。
First Sector	**EFI**－起始块编号。不显示在带有 VTOC 标签的磁盘上。
Last Sector	**EFI**－结束块编号。不显示在带有 VTOC 标签的磁盘上。

以下是使用 prtvtoc 命令显示的 EFI 磁盘标签的示例。

```
# prtvtoc /dev/rdsk/c4t1d0s0
* /dev/rdsk/c4t1d0s0 partition map
*
* Dimensions:
*     512 bytes/sector
* 2576941056 sectors
* 2576940989 accessible sectors
*
* Flags:
*   1: unmountable
*  10: read-only
*
*                           First     Sector    Last
* Partition  Tag  Flags     Sector     Count    Sector   Mount Directory
       0      2    00          34   629145600  629145633
       1      4    00   629145634   629145600 1258291233
       6      4    00  1258291234  1318633404 2576924637
       8     11    00  2576924638       16384 2576941021
```

prtvtoc 命令的输出按以下三个部分提供信息：

- 尺寸
- Flag（标志）
- 分区表

prtvtoc 列名	说明
Partition	分区编号或分片编号。有关此列的说明，请参见表 10–4。
Tag	分区标记。有关此列的说明，请参见表 10–4。
Flags	分区标志。有关此列的说明，请参见表 10–4。
First Sector	分片的第一个扇区。
Sector Count	分片中的扇区总数。
Last Sector	分片的最后一个扇区。
Mount Directory	文件系统的最后一个挂载点目录。

使用浮动分片 (free hog slice)

使用 format 实用程序更改一个或多个磁盘分片的大小时，可以指定为适应大小调整操作而伸缩的临时分片。

扩展分片时，此临时分片将提供或“释放”空间；收缩分片时，此临时分片将接收或“占据”放弃的空间。出于此原因，提供分片有时称为**浮动分片** *(free hog slice)*。

浮动分片 (free hog slice) 仅在安装期间或运行 format 实用程序时存在。日常操作期间没有永久的浮动分片 (free hog slice)。

有关使用浮动分片 (free hog slice) 的信息，请参见第 207 页中的 “SPARC: 如何为 ZFS 根文件系统创建磁盘分片”或第 215 页中的 “x86: 如何为 ZFS 根文件系统创建磁盘分片”。

◆ ◆ ◆ 第 11 章

11 管理磁盘（任务）

本章介绍管理磁盘的过程。如果您已经熟悉如何在运行 Oracle Solaris OS 的系统中管理磁盘，则本章介绍的许多过程是可选的。

有关与管理磁盘关联的过程的信息，请参见第 189 页中的 “管理磁盘（任务列表）”。

有关磁盘管理的概述信息，请参见第 10 章，管理磁盘（概述）。

管理磁盘（任务列表）

任务	说明	参考
确定系统中的磁盘。	如果不能确定系统中的磁盘类型，请使用 `format` 实用程序确定磁盘类型。	第 190 页中的 “如何确定系统中的磁盘”
格式化磁盘。	使用 `format` 实用程序确定磁盘是否已格式化。 大多数情况下，磁盘已经格式化。如果需要格式化磁盘，请使用 `format` 实用程序。	第 192 页中的 “如何确定磁盘是否已格式化” 第 193 页中的 “如何格式化磁盘”
显示分片信息。	使用 `format` 实用程序显示分片信息。	第 194 页中的 “如何显示磁盘分片信息”
为磁盘设置标签。	使用 `format` 实用程序创建磁盘标签。	第 196 页中的 “如何为磁盘设置标签”
检查磁盘标签。	使用 `prtvtoc` 命令检查磁盘标签。	第 200 页中的 “如何检查磁盘标签”
恢复损坏的磁盘标签。	可以尝试恢复因系统或电源故障而损坏的磁盘标签。	第 202 页中的 “如何恢复损坏的磁盘标签”

确定系统中的磁盘

使用 format 实用程序搜索已连接到系统的磁盘的类型。还可以使用 format 实用程序验证磁盘是否是系统可识别的。有关使用 format 实用程序的详细信息，请参见第 16 章，format 实用程序（参考）。

▼ 如何确定系统中的磁盘

1. **成为管理员。**

   有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。

2. **使用 format 实用程序确定系统可识别的磁盘。**

   ```
   # format
   ```

   format 实用程序在 AVAILABLE DISK SELECTIONS 下显示它所识别的磁盘的列表。

示例 11–1 确定系统中的磁盘

以下示例显示 format 命令的输出。

```
# format
AVAILABLE DISK SELECTIONS:
       0. c2t0d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
          /pci@1c,600000/scsi@2/sd@0,0
       1. c2t1d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
          /pci@1c,600000/scsi@2/sd@1,0
       2. c2t2d0 <SUN146G cyl 14087 alt 2 hd 24 sec 848>
          /pci@1c,600000/scsi@2/sd@2,0
       3. c2t3d0 <SUN146G cyl 14087 alt 2 hd 24 sec 848>
          /pci@1c,600000/scsi@2/sd@3,0
Specify disk (enter its number):
```

该输出将磁盘的物理设备名称和逻辑设备名称与磁盘的市场名称（位于尖括号 <> 中）相关联。请参见下面的示例。使用此方法可以轻松地确定哪些逻辑设备名称表示已连接到系统的磁盘。有关逻辑设备名称和物理设备名称的说明，请参见第 70 页中的“设备命名约定”。

以下示例使用通配符显示已连接到控制器 0 的四个磁盘。

```
# format /dev/rdsk/c0t6*
AVAILABLE DISK SELECTIONS:
       0. /dev/rdsk/c0t600A0B800022024E000054AC4970A629d0p0 <...>
          /scsi_vhci/disk@g600a0b800022024e000054ac4970a629
       1. /dev/rdsk/c0t600A0B800022024E000054AE4970A711d0p0 <...>
          /scsi_vhci/disk@g600a0b800022024e000054ae4970a711
       2. /dev/rdsk/c0t600A0B800022028A000050444970A834d0p0 <....>
          /scsi_vhci/disk@g600a0b800022028a000050444970a834
       3. /dev/rdsk/c0t600A0B800022028A000050454970A8EAd0p0 <...>
          /scsi_vhci/disk@g600a0b800022028a000050454970a8ea
Specify disk (enter its number):
```

以下示例说明如何确定磁盘信息。

```
# format
0. c2t0d0 <SUN36G cyl 24620 alt 2 hd 27 sec 107>
          /pci@1c,600000/scsi@2/sd@0,0
```

该输出表明磁盘 0（目标 0）已连接到第二个 SCSI 主机适配器 (scsi@2)，该适配器已连接到第二个 PCI 接口 (/pci@1c,600000/...)。该输出还将物理设备名称和逻辑设备名称与磁盘的市场名称 SUN36G 相关联。

某些磁盘不具备市场名称。如果 format 输出不通过其市场名称确定磁盘，那么则可以按照以下步骤中的说明使用 format 实用程序的 type 和 label 功能来包含磁盘的市场名称。

磁盘或系统不活动时，必须执行以下步骤，这表示从安装 DVD 或网络引导，除非该磁盘当前未使用，并且不包含 Oracle Solaris 发行版。另外，最后一步是重新为磁盘设置标签，这可以删除任何现有分区信息或数据。

```
# format
Searching for disks...done

AVAILABLE DISK SELECTIONS:
       0. c2t0d0 <SUN36G cyl 24620 alt 2 hd 27 sec 107>
          /pci@1c,600000/scsi@2/sd@0,0
       1. c2t1d0 <SEAGATE-ST336607LSUN36G-0307-33.92GB>
          /pci@1c,600000/scsi@2/sd@1,0
       2. c2t2d0 <SEAGATE-ST336607LSUN36G-0507-33.92GB>
          /pci@1c,600000/scsi@2/sd@2,0
       3. c2t3d0 <drive type unknown>
          /pci@1c,600000/scsi@2/sd@3,0
Specify disk (enter its number): 3
selecting c2t3d0
[disk formatted]
format> type
AVAILABLE DRIVE TYPES:
        0. Auto configure
        1. other
Specify disk type (enter its number): 0
c2t3d0: configured with capacity of 33.92GB
<SEAGATE-ST336607LSUN36G-0507-33.92GB>
[disk formatted]
format> label
Ready to label disk, continue? yes
format> quit
```

更多信息 如果 format 实用程序无法识别磁盘...

- 请转至第 12 章，SPARC：设置磁盘（任务）或第 13 章，x86：设置磁盘（任务）。
- 请转至第 196 页中的“如何为磁盘设置标签”。
- 请通过参阅磁盘硬件文档将磁盘连接到系统。

格式化磁盘

磁盘通常由制造商或转售商进行格式化。在安装驱动器时，通常无需重新格式化磁盘。

必须已对磁盘进行格式化后才能执行以下操作：

- 将数据写入磁盘。不过，大多数磁盘都已经格式化。
- 使用 Oracle Solaris 安装实用程序安装系统。

注意 – 格式化磁盘是一个破坏性过程，因为它覆盖磁盘上的数据。由于此原因，磁盘通常仅由制造商或转售商进行格式化。如果认为磁盘缺陷是导致问题反复出现的原因，则可以使用 `format` 实用程序执行表面分析。但是，应非常谨慎地仅使用不会破坏数据的命令。

▼ 如何确定磁盘是否已格式化

1. **成为管理员。**

   有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。

2. **调用 `format` 实用程序。**

   ```
   # format
   ```

   显示磁盘的编号列表。

3. **键入要检查的磁盘的编号。**

   ```
   Specify disk (enter its number): 0
   ```

4. **通过查看以下消息，验证所选磁盘是否已格式化：**

   ```
   [disk formatted]
   ```

示例 11–2 确定磁盘是否已格式化

以下示例说明磁盘 `c2t3d0` 已格式化。

```
# format
AVAILABLE DISK SELECTIONS:
       0. c2t0d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
          /pci@1c,600000/scsi@2/sd@0,0
       1. c2t1d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
          /pci@1c,600000/scsi@2/sd@1,0
       2. c2t2d0 <SUN146G cyl 14087 alt 2 hd 24 sec 848>
          /pci@1c,600000/scsi@2/sd@2,0
       3. c2t3d0 <SUN146G cyl 14087 alt 2 hd 24 sec 848>
          /pci@1c,600000/scsi@2/sd@3,0
```

```
Specify disk (enter its number): 3
selecting c2t3d0
[disk formatted]
```

▼ 如何格式化磁盘

磁盘由制造商进行格式化。很少出现重新格式化磁盘。该过程很耗费时间，并且会删除磁盘上的所有数据。

1 成为管理员。

有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。

2 调用 format 实用程序。

```
# format
```

显示磁盘的编号列表。

3 键入要格式化的磁盘的编号。

```
Specify disk (enter its number): 3
```

注意 – 请勿选择包含根文件系统的磁盘。如果格式化根池磁盘，则会删除该磁盘上的 OS 和任何数据。

4 要开始格式化磁盘，请在 format> 提示符下键入 format。通过键入 y 确认该命令。

```
format> format
The protection information is not enabled
The disk will be formatted with protection type 0

Ready to format.  Formatting cannot be interrupted
and takes 169 minutes (estimated). Continue? yes
```

5 通过查看以下消息，验证磁盘格式化是否已成功：

```
Beginning format. The current time is Fri Apr 1 ...

Formatting...
done

Verifying media...
        pass 0 - pattern = 0xc6dec6de
14086/23/734

        pass 1 - pattern = 0x6db6db6d
   14086/23/734

Total of 0 defective blocks repaired.
```

6 退出 format 实用程序。

```
format> quit
```

显示磁盘分片

创建 ZFS 存储池的最佳方法是使用整个磁盘，而非磁盘分片，因为整个磁盘更易于管理。仅在磁盘用于 ZFS 根池时需要使用磁盘分片。这是长期的引导限制。针对非根池，请使用整个磁盘。当使用整个磁盘创建池时，将应用 EFI 标签。请参见以下 EFI 磁盘标签示例。

如果需要准备磁盘以使用根池磁盘，请按照下面的 VTOC 磁盘标签示例中的说明创建包含整个磁盘容量的分片 0。

有关设置结合 ZFS 存储池使用的磁盘的信息，请参见第 12 章，SPARC：设置磁盘（任务）或第 13 章，x86：设置磁盘（任务）。

注 – `format` 实用程序使用术语 *partition* 而不是 *slice*。

▼ 如何显示磁盘分片信息

如果磁盘打算用于 ZFS 根池，则可能需要显示磁盘分片信息。还必须包括 SMI 标签。

1. **成为管理员。**
   有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。

2. **调用 `format` 实用程序。**

   ```
   # format
   ```

   显示磁盘的编号列表。

3. **键入要显示其分片信息的磁盘的编号。**

   ```
   Specify disk (enter its number):1
   ```

4. **选择 `partition` 菜单。**

   ```
   format> partition
   ```

5. **显示选定磁盘的分片信息。**

   ```
   partition> print
   ```

6. **退出 `format` 实用程序。**

   ```
   partition> q
   format> q
   ```

7. **通过确定特定的分片标记和分片，验证所显示的分片信息。**
   如果屏幕输出说明未指定分片大小，则磁盘很可能没有分片。

示例 11–3 显示磁盘分片信息

以下示例显示标签为 VTOC 的磁盘的分片信息。

```
# format
Searching for disks...done
Specify disk (enter its number):3
Selecting c2t3d0
format> partition
partition> print
Current partition table (c2t3d0):
Total disk cylinders available: 14087 + 2 (reserved cylinders)

Part      Tag    Flag     Cylinders         Size            Blocks
  0       root    wm       0 - 14086      136.71GB    (14087/0/0) 286698624
  1       swap    wu       0                0         (0/0/0)             0
  2     backup    wu       0 - 14086      136.71GB    (14087/0/0) 286698624
  3 unassigned    wm       0                0         (0/0/0)             0
  4 unassigned    wm       0                0         (0/0/0)             0
  5 unassigned    wm       0                0         (0/0/0)             0
  6        usr    wm       0                0         (0/0/0)             0
  7 unassigned    wm       0                0         (0/0/0)             0
partition> q
format> q
```

有关这些示例中分片信息的详细说明，请参见第 10 章，管理磁盘（概述）。

以下示例显示标签为 EFI 的磁盘的分片信息。

```
# format
Searching for disks...done
Specify disk (enter its number): 3
selecting c2t3d0
[disk formatted]
format> partition
partition> print
Current partition table (default):
Total disk sectors available: 286722878 + 16384 (reserved sectors)

Part      Tag    Flag     First Sector         Size         Last Sector
  0        usr    wm                34      136.72GB          286722911
  1 unassigned    wm                 0           0               0
  2 unassigned    wm                 0           0               0
  3 unassigned    wm                 0           0               0
  4 unassigned    wm                 0           0               0
  5 unassigned    wm                 0           0               0
  6 unassigned    wm                 0           0               0
  7 unassigned    wm                 0           0               0
  8   reserved    wm         286722912        8.00MB          286739295
partition> q
format> q
```

创建和检查磁盘标签

为磁盘设置标签通常是在系统安装过程中或者使用新磁盘时进行的。如果磁盘标签已损坏，则可能需要重新为磁盘设置标签。例如，磁盘标签因电源故障而损坏。

format 实用程序会尝试自动配置任何不带标签的 SCSI 磁盘。如果 format 实用程序能够自动配置不带标签的磁盘，则它将显示与以下内容类似的消息：

```
c2t3d0: configured with capacity of 136.73GB
```

▼ 如何为磁盘设置标签

可以使用接下来的过程执行以下操作：

- 用标签 VTOC 或 EFI 为 2 TB 大小的磁盘设置标签。
- 用标签 EFI 为大于 2 TB 的磁盘设置标签。

如果要用标签 EFI 为小于 2 TB 的磁盘设置标签，请参见示例 11–5。

1 成为管理员。

有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。

2 调用 format 实用程序。

```
# format
```

显示磁盘的编号列表。

3 键入要设置标签的磁盘的编号。

```
Specify disk (enter its number):1
```

如果 format 实用程序识别该磁盘类型，则下一步是搜索备份标签以为磁盘设置标签。使用备份标签为磁盘设置标签时，会以正确的分区信息、磁盘类型和磁盘几何参数为磁盘设置标签。

4 选择以下内容之一以为磁盘设置标签：

- 如果磁盘是不带标签的但已成功配置，请转至步骤 5 为磁盘设置标签。

  format 实用程序将询问您是否要为磁盘设置标签。

- 如果已为磁盘设置标签但您要更改磁盘类型，或者如果 format 实用程序无法自动配置磁盘，请继续执行步骤 6 设置磁盘类型并为磁盘设置标签。

5 通过在 Label it now? 提示符下键入 y 来为磁盘设置标签。

```
Disk not labeled. Label it now? y
```

现在已为磁盘设置标签。转至步骤 10 以退出 format 实用程序。

6 **在 format> 提示符下输入 type。**

```
format> type
```

将显示 "Available Drive Types"（可用的驱动器类型）菜单。

7 **从可能的磁盘类型列表中选择一种磁盘类型。**

```
Specify disk type (enter its number)[12]: 12
```

或者，选择 0 自动配置 SCSI-2 磁盘。

8 **为磁盘设置标签。如果磁盘未设置标签，则显示以下消息。**

```
Disk not labeled. Label it now? y
```

否则，将显示以下消息提示您：

```
Ready to label disk, continue? y
```

9 **验证磁盘标签。**

```
format> verify
```

10 **退出 format 实用程序。**

```
format> q
#
```

示例 11–4 为磁盘设置标签

以下示例说明如何自动配置并为 36 GB 磁盘设置标签。

```
# format
AVAILABLE DISK SELECTIONS:
       0. c0t0d0 <SUN36G cyl 24620 alt 2 hd 27 sec 107>
          /pci@1c,600000/scsi@2/sd@0,0
       1. c0t1d0 <SEAGATE-ST336607LSUN36G-0307-33.92GB>
          /pci@1c,600000/scsi@2/sd@1,0
       2. c0t2d0 <SEAGATE-ST336607LSUN36G-0507-33.92GB>
          /pci@1c,600000/scsi@2/sd@2,0
       3. c0t3d0 <SEAGATE-ST336607LSUN36G-0507-33.92GB>
          /pci@1c,600000/scsi@2/sd@3,0
Specify disk (enter its number): 3
c0t3d0: configured with capacity of 33.92GB
Disk not labeled.  Label it now?  yes
format> verify
format> q
```

示例 11–5 用标签 EFI 为磁盘设置标签

以下示例说明如何使用 `format -e` 命令以标签 EFI 为磁盘设置标签。请记住验证分层软件产品是否继续在包含标签 EFI 的磁盘的系统上起作用。有关 EFI 标签限制的常规信息，请参见第 179 页中的 “EFI 磁盘标签限制”。

```
# format -e
Searching for disks...done
AVAILABLE DISK SELECTIONS:
       0. c2t0d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
          /pci@1c,600000/scsi@2/sd@0,0
       1. c2t1d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
          /pci@1c,600000/scsi@2/sd@1,0
       2. c2t2d0 <FUJITSU-MAP3147N SUN146G-0501-136.73GB>
          /pci@1c,600000/scsi@2/sd@2,0
       3. c2t3d0 <FUJITSU-MAP3147N SUN146G-0501-136.73GB>
          /pci@1c,600000/scsi@2/sd@3,0
Specify disk (enter its number): 3
selecting c2t3d0
[disk formatted]
format> label
[0] SMI Label
[1] EFI Label
Specify Label type[0]: 1
Ready to label disk, continue? yes
format> quit
```

示例 11–6 将以 EFI 设置标签的磁盘更改为以 SMI 设置标签的磁盘

以下示例说明如何使用 `format - e` 命令将以 EFI 设置标签的磁盘更改为可以用于 ZFS 根池的以 SMI 设置标签的磁盘。

在 x86 系统中，首先必须将 EFI fdisk 分区更改为 Solaris 分区。例如：

```
# format -e
select disk ...
format> fdisk
FORMAT MENU:
        disk       - select a disk
        type       - select (define) a disk type
             Total disk size is 17833 cylinders
             Cylinder size is 16065 (512 byte) blocks

                                                Cylinders
      Partition   Status    Type          Start   End   Length    %
      =========   ======    ============  =====   ===   ======   ===
          1                 EFI               0  17833    17834    100



SELECT ONE OF THE FOLLOWING:
   1. Create a partition
   2. Specify the active partition
   3. Delete a partition
   4. Change between Solaris and Solaris2 Partition IDs
   5. Edit/View extended partitions
   6. Exit (update disk configuration and exit)
   7. Cancel (exit without updating disk configuration)
Enter Selection: 3
Specify the partition number to delete (or enter 0 to exit): 1
This will make all files and
```

```
programs in this partition inaccessible (type "y" or "n"). y
Enter Selection: 1
Select the partition type to create:
   1=SOLARIS2   2=UNIX      3=PCIXOS     4=Other        5=DOS12
   6=DOS16      7=DOSEXT    8=DOSBIG     9=DOS16LBA     A=x86 Boot
   B=Diagnostic C=FAT32     D=FAT32LBA   E=DOSEXTLBA    F=EFI (Protective)
   G=EFI_SYS    0=Exit? 1
Specify the percentage of disk to use for this partition
(or type "c" to specify the size in cylinders). 100
Should this become the active partition? If yes, it  will be activated
each time the computer is reset or turned on.
Please type "y" or "n". y
Enter Selection: 6
Partition 1 is now the active partition.
```

在 SPARC 系统中，请按照以下步骤。

```
# format -e
Searching for disks...done
AVAILABLE DISK SELECTIONS:
       0. c2t0d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
          /pci@1c,600000/scsi@2/sd@0,0
       1. c2t1d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
          /pci@1c,600000/scsi@2/sd@1,0
       2. c2t2d0 <FUJITSU-MAP3147N SUN146G-0501-136.73GB>
          /pci@1c,600000/scsi@2/sd@2,0
       3. c2t3d0 <FUJITSU-MAP3147N SUN146G-0501-136.73GB>
          /pci@1c,600000/scsi@2/sd@3,0
Specify disk (enter its number): 3
selecting c2t0d0
[disk formatted]
format> label
[0] SMI Label
[1] EFI Label
Specify Label type[1]: 0
Ready to label disk, continue? yes
```

还应该检查缺省分区表，以确保其对于根池分片是最佳的，这表示大多数磁盘空间位于分片 0 中。请参见下面的步骤增加分片 0 的大小。

```
format> partition
partition> print
Current partition table (default):
Total disk cylinders available: 14085 + 2 (reserved cylinders)

Part      Tag    Flag     Cylinders         Size            Blocks
  0       root    wm       1 -    13      129.19MB    (13/0/0)       264576
  1       swap    wu      14 -    26      129.19MB    (13/0/0)       264576
  2     backup    wu       0 - 14086      136.71GB    (14087/0/0) 286698624
  3 unassigned    wm       0               0         (0/0/0)             0
  4 unassigned    wm       0               0         (0/0/0)             0
  5 unassigned    wm       0               0         (0/0/0)             0
  6        usr    wm      27 - 14084      136.43GB    (14058/0/0) 286108416
  7 unassigned    wm       0               0         (0/0/0)             0
  8       boot    wu       0 -     0        9.94MB    (1/0/0)         20352
  9 alternates    wm       0               0         (0/0/0)             0
```

```
partition> modify
Select partitioning base:
        0. Current partition table (default)
        1. All Free Hog
Choose base (enter number) [0]? 1

Part      Tag    Flag     Cylinders         Size            Blocks
  0       root    wm       0               0         (0/0/0)             0
  1       swap    wu       0               0         (0/0/0)             0
  2     backup    wu       0 - 14084      136.69GB    (14085/0/0) 286657920
  3 unassigned    wm       0               0         (0/0/0)             0
  4 unassigned    wm       0               0         (0/0/0)             0
  5 unassigned    wm       0               0         (0/0/0)             0
  6        usr    wm       0               0         (0/0/0)             0
  7 unassigned    wm       0               0         (0/0/0)             0
  8       boot    wu       0 -     0       9.94MB    (1/0/0)         20352
  9 alternates    wm       0               0         (0/0/0)             0

Do you wish to continue creating a new partition
table based on above table[yes]? yes
Free Hog partition[6]? 0
Enter size of partition '1' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '3' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '4' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '5' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '6' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '7' [0b, 0c, 0.00mb, 0.00gb]:

Part      Tag    Flag     Cylinders         Size            Blocks
  0       root    wm       1 - 14084      136.68GB    (14084/0/0) 286637568
  1       swap    wu       0               0         (0/0/0)             0
  2     backup    wu       0 - 14084      136.69GB    (14085/0/0) 286657920
  3 unassigned    wm       0               0         (0/0/0)             0
  4 unassigned    wm       0               0         (0/0/0)             0
  5 unassigned    wm       0               0         (0/0/0)             0
  6        usr    wm       0               0         (0/0/0)             0
  7 unassigned    wm       0               0         (0/0/0)             0
  8       boot    wu       0 -     0       9.94MB    (1/0/0)         20352
  9 alternates    wm       0               0         (0/0/0)             0

Okay to make this the current partition table[yes]? yes
Enter table name (remember quotes): "c2t0d0"
Ready to label disk, continue? yes
partition> quit
format> quit
```

▼ 如何检查磁盘标签

使用 `prtvtoc` 命令检查磁盘标签信息。有关磁盘标签的详细说明和 `prtvtoc` 命令所显示的信息，请参见第 10 章，管理磁盘（概述）。

1 成为管理员。

有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。

2 **显示磁盘标签信息。**

```
# prtvtoc /dev/rdsk/device-name
```

其中 *device-name* 是要检查的原始磁盘设备。

示例 11–7 检查磁盘标签

以下示例说明标签为 VTOC 的磁盘的磁盘标签信息。

```
# prtvtoc /dev/rdsk/c2t3d0s0
* /dev/rdsk/c2t3d0s0 partition map
*
* Dimensions:
*     512 bytes/sector
*     848 sectors/track
*      24 tracks/cylinder
*   20352 sectors/cylinder
*   14089 cylinders
*   14087 accessible cylinders
*
* Flags:
*   1: unmountable
*  10: read-only
*
*                          First     Sector    Last
* Partition  Tag  Flags    Sector     Count    Sector  Mount Directory
       0      2    00          0 286698624 286698623
       2      5    01          0 286698624 286698623
```

以下示例说明标签为 EFI 的磁盘的磁盘标签信息。

```
# prtvtoc /dev/rdsk/c2t3d0s0
* /dev/rdsk/c2t3d0s0 partition map
*
* Dimensions:
*     512 bytes/sector
*     848 sectors/track
*      24 tracks/cylinder
*   20352 sectors/cylinder
*   14089 cylinders
*   14087 accessible cylinders
*
* Flags:
*   1: unmountable
*  10: read-only
*
*                          First     Sector    Last
* Partition  Tag  Flags    Sector     Count    Sector  Mount Directory
       0      2    00          0 286698624 286698623
       2      5    01          0 286698624 286698623
```

恢复损坏的磁盘标签

有时，电源或系统故障会导致磁盘标签变得无法识别。损坏的磁盘标签并不总是意味着，必须重新创建或恢复分片信息或磁盘数据。

要恢复损坏的磁盘标签，首先应使用正确的几何参数和磁盘类型信息为磁盘设置标签。可以通过常规的磁盘设置标签方法完成该步骤，即使用自动配置或手动指定磁盘类型。

如果 `format` 实用程序识别该磁盘类型，则下一步是搜索备份标签以为磁盘设置标签。使用备份标签为磁盘设置标签时，会以正确的分区信息、磁盘类型和磁盘几何参数为磁盘设置标签。

▼ 如何恢复损坏的磁盘标签

1 将系统引导到单用户模式。

如有必要，请在单用户模式下从本地 Oracle Solaris DVD 或网络引导系统以访问磁盘。

有关引导系统的信息，请参见《在 SPARC 平台上引导和关闭 Oracle Solaris》或《在 x86 平台上引导和关闭 Oracle Solaris》。

2 重新为磁盘设置标签。

```
# format
```

`format` 实用程序会尝试自动配置任何不带标签的 SCSI 磁盘。如果 `format` 实用程序能够配置不带标签的已损坏磁盘，则它将显示以下消息：

cwtxdy`: configured with capacity of` *abc*`MB`

`format` 实用程序然后显示系统中磁盘的编号列表。

3 键入需要恢复的磁盘的编号。

```
Specify disk (enter its number): 1
```

4 选择以下内容之一以确定如何为磁盘设置标签。

- 如果已成功配置磁盘，则按照步骤 5 和 6 进行操作。然后转至步骤 12。
- 如果磁盘未成功配置，则按照步骤 7–11 操作。然后转至步骤 12。

5 搜索备份标签。

```
format> verify
Warning: Could not read primary label.
Warning: Check the current partitioning and 'label' the disk or
use the 'backup' command.
Backup label contents:
Volume name = <        >
```

```
ascii name  = <FUJITSU-MAP3147N SUN146G-0501>
pcyl        = 14089
ncyl        = 14087
acyl        =    2
nhead       =   24
nsect       =  848
Part      Tag    Flag     Cylinders         Size            Blocks
  0       root    wm       0 - 14086      136.71GB    (14087/0/0) 286698624
  1       swap    wu       0                0         (0/0/0)             0
  2     backup    wu       0 - 14086      136.71GB    (14087/0/0) 286698624
  3 unassigned    wm       0                0         (0/0/0)             0
  4 unassigned    wm       0                0         (0/0/0)             0
  5 unassigned    wm       0                0         (0/0/0)             0
  6        usr    wm       0                0         (0/0/0)             0
  7 unassigned    wm       0                0         (0/0/0)             0
```

6 **如果 `format` 实用程序找到备份标签，而且备份标签的内容符合要求，请使用 `backup` 命令以备份标签为磁盘设置标签。**

```
format> backup
Disk has a primary label, still continue? y

Searching for backup labels...found.
Restoring primary label
```

磁盘标签已恢复。转至步骤 12。

7 **如果 `format` 实用程序无法自动配置磁盘，请使用 `type` 命令指定磁盘类型。**

```
format> type
```

将显示 "Available Drive Types"（可用的驱动器类型）菜单。

8 **选择 `0` 以自动配置磁盘。或者，从可能的磁盘类型列表中选择一种磁盘类型。**

```
Specify disk type (enter its number)[12]: 12
```

9 **如果成功配置磁盘，则在 `format` 实用程序询问您是否要为磁盘设置标签时应答 `no`。**

```
Disk not labeled.  Label it now?  no
```

10 **使用 `verify` 命令搜索备份标签。**

```
format> verify
Warning: Could not read primary label.
Warning: Check the current partitioning and 'label' the disk
or use the 'backup' command.
.
.
.
```

11 **如果 `format` 实用程序找到备份标签，而且备份标签的内容符合要求，请使用 `backup` 命令以备份标签为磁盘设置标签。**

```
format> backup
Disk has a primary label, still continue? y
Searching for backup labels...found.
Restoring primary label
```

磁盘标签已恢复。

12 **退出 format 实用程序。**

```
format> q
```

13 **验证已恢复磁盘上的文件系统。**

有关将 zpool scrub 命令用于 ZFS 文件系统的信息，请参见zpool(1M)。

添加第三方磁盘

Oracle Solaris OS 支持许多第三方磁盘。但是，为了识别磁盘，您可能需要提供设备驱动程序。用于添加磁盘的其他选项如下：

- 如果要添加 SCSI 磁盘，则可能尝试使用 format 实用程序的自动配置功能。
- 可能尝试热插拔 PCI、SCSI 或 USB 磁盘。有关更多信息，请参见第 5 章，管理设备（概述/任务）。

注 – Oracle 无法保证其 format 实用程序将与所有第三方磁盘驱动程序兼容。如果磁盘驱动程序与 Solaris format 实用程序不兼容，则磁盘驱动器供应商应该为您提供定制磁盘格式化程序。

通常，在调用 format 实用程序时如果发现未识别出磁盘类型，则说明缺少软件支持。

请参阅第 12 章，SPARC：设置磁盘（任务）或第 13 章，x86：设置磁盘（任务）中添加系统磁盘或辅助磁盘的相应配置过程。

◆ ◆ ◆ 第 12 章

12 SPARC：设置磁盘（任务）

本章介绍如何在 SPARC 系统中设置磁盘。

有关与在 SPARC 系统中设置磁盘关联的过程的信息，请参见第 205 页中的 “SPARC: 为 ZFS 文件系统设置磁盘（任务列表）”：

有关磁盘管理的概述信息，请参见第 10 章，管理磁盘（概述）。有关在基于 x86 的系统中设置磁盘的逐步说明，请参见第 13 章，x86：设置磁盘（任务）。

SPARC: 为 ZFS 文件系统设置磁盘（任务列表）

以下任务列表介绍了为 ZFS 根文件系统设置 ZFS 根池磁盘以及为基于 SPARC 的系统设置 ZFS 非根池磁盘的过程。

任务	说明	参考
1. 为 ZFS 根文件系统设置磁盘。	*ZFS* **根文件系统的磁盘** 连接新磁盘或替换现有根池磁盘，并从本地或远程 Oracle Solaris DVD 进行引导。	第 206 页中的 “SPARC: 如何为 ZFS 根文件系统设置磁盘”
2. 为 ZFS 根文件系统创建磁盘分片。	为用于 ZFS 根池的磁盘创建磁盘分片。这是长期的引导限制。	第 207 页中的 “SPARC: 如何为 ZFS 根文件系统创建磁盘分片”
3. 为 ZFS 根文件系统安装引导块（如有必要）。	如果用 `zpool replace` 命令替换用于根池的磁盘，那么必须手动安装引导块，以使系统可以从替换磁盘进行引导。	第 211 页中的 “SPARC: 如何为 ZFS 根文件系统安装引导块”
4. 为 ZFS 文件系统设置磁盘。	*ZFS* **文件系统的磁盘** 为 ZFS 文件系统设置磁盘。	第 211 页中的 “SPARC: 如何为 ZFS 文件系统设置磁盘”

SPARC: 为 ZFS 文件系统设置磁盘

虽然说明如何设置磁盘的过程可以结合 ZFS 文件系统使用，但是 ZFS 文件系统不直接映射到磁盘或磁盘分片。必须创建 ZFS 存储池才能创建 ZFS 文件系统。有关更多信息，请参见《Oracle Solaris 管理： ZFS 文件系统》 。

根池包含用于引导 Oracle Solaris OS 的根文件系统。如果根池磁盘已损坏，且根池未进行镜像配置，则系统可能不会引导。如果根池磁盘已损坏，有两种可以恢复的方法：

- 可以重新安装整个 Oracle Solaris OS。
- 或者，可以替换根池磁盘并从快照或备份介质中恢复文件系统。可以通过创建冗余根池减少因硬件故障而导致的系统停机时间。唯一受支持的冗余根池配置是镜像根池。

用于非根池中的磁盘通常包含用户或数据文件的空间。为了获得更多磁盘空间，可以向根池或非根池附加或添加另一个磁盘。或者，可以使用以下方法替换池中损坏的磁盘。

- 如果所有的设备当前为 ONLINE（联机），则可以在非冗余池中替换磁盘。
- 如果其他设备中存在足够的冗余，则可以在冗余池中替换磁盘。
- 在镜像根池中，可以替换磁盘或附加磁盘，然后拆离故障磁盘或较小的磁盘，以增加池的大小。

通常情况下，在系统中设置磁盘取决于硬件，所以当在系统中添加或替换磁盘时，请查看硬件文档。如果需要将磁盘添加到现有的控制器中，那么这可能只是将磁盘插入空槽（如果系统支持热插拔）中的问题而已。如果需要配置新的控制器，请参见第 75 页中的 “动态重新配置和热插拔”。

▼ SPARC: 如何为 ZFS 根文件系统设置磁盘

有关替换磁盘的信息，请参阅硬件安装指南。

1. **断开损坏的磁盘与系统之间的连接（ 如有必要 ）。**
2. **将替换磁盘连接至系统，并检查磁盘的物理连接（ 如有必要 ）。**
3. **根据从本地 Oracle Solaris DVD 还是从网络中的远程 Oracle Solaris DVD 引导，按照下表中的说明进行操作。**

引导类型	操作
从本地驱动器中的 Oracle Solaris DVD	1. 确保 Oracle Solaris DVD 位于驱动器中。 2. 从介质引导到单用户模式： ok **boot cdrom -s**
从网络	从网络引导到单用户模式： ok **boot net -s**

几分钟后，将显示根提示符 (#)。

更多信息 **为 ZFS 根文件系统设置磁盘之后...**

磁盘已连接或已替换之后，可以创建分片并更新磁盘标签。请转至第 207 页中的 “SPARC: 如何为 ZFS 根文件系统创建磁盘分片”。

SPARC: 为 ZFS 根文件系统创建磁盘分片

必须为用于 ZFS 根池的磁盘创建磁盘分片。这是长期的引导限制。查看以下根池磁盘要求：

- 必须包含磁盘分片和标签 SMI (VTOC)。
- 根池磁盘不支持标签 EFI。
- 必须是单个磁盘或镜像配置的一部分。根池既不支持非冗余配置，也不支持 RAIDZ 配置。
- 作为 OS 映像一部分的根文件系统的所有子目录（/var 除外）必须与根文件系统处于同一数据集。
- 除交换和转储设备外，所有其他 Solaris OS 组件必须驻留在根池中。

通常情况下，应该创建在分片 0 中具有大量磁盘空间的磁盘分片。不推荐尝试使用磁盘上不同的分片并在不同的操作系统中共享该磁盘，或与不同的 ZFS 存储池或存储池组件一同使用。

▼ SPARC: 如何为 ZFS 根文件系统创建磁盘分片

通常情况下，安装系统时，会自动安装根池磁盘。如果需要替换根池磁盘或将新磁盘附加为镜像根池磁盘，请参见以下步骤。

1 **成为管理员。**

2 **使故障磁盘脱机并取消其配置（如有必要）。**

有些硬件要求您在尝试通过 zpool replace 操作替换故障磁盘之前，使磁盘脱机并取消其配置。例如：

```
# zpool offline rpool c2t1d0s0
# cfgadm -c unconfigure c2::dsk/c2t1d0
```

3 **以物理形式将新磁盘或替换磁盘连接至系统（如有必要）。**

a. **以物理形式删除故障磁盘。**

b. **以物理形式插入替换磁盘。**

c. **配置替换磁盘（如有必要）。例如：**

```
# cfgadm -c configure c2::dsk/c2t1d0
```

对于某些硬件，插入替换磁盘后不必对其进行重新配置。

4 **通过查看格式化输出确认磁盘可存取。**

例如，format 命令显示 4 个连接至该系统的磁盘。

```
# format -e
AVAILABLE DISK SELECTIONS:
       0. c2t0d0 <SUN36G cyl 24620 alt 2 hd 27 sec 107>
          /pci@1c,600000/scsi@2/sd@0,0
       1. c2t1d0 <SEAGATE-ST336607LSUN36G-0307-33.92GB>
          /pci@1c,600000/scsi@2/sd@1,0
       2. c2t2d0 <SEAGATE-ST336607LSUN36G-0507-33.92GB>
          /pci@1c,600000/scsi@2/sd@2,0
       3. c2t3d0 <SEAGATE-ST336607LSUN36G-0507-33.92GB>
          /pci@1c,600000/scsi@2/sd@3,0
```

5 **选择用于 ZFS 根池的磁盘。**

6 **通过显示分区（分片）信息确认磁盘具有标签 SMI。**

例如，c2t1d0 的分区（分片）输出显示该磁盘具有标签 EFI，因为其标识第一个和最后一个扇区。

```
Specify disk (enter its number): 1
selecting c2t1d0
[disk formatted]
format> p
PARTITION MENU:
        0      - change ‘0’ partition
        1      - change ‘1’ partition
        2      - change ‘2’ partition
        3      - change ‘3’ partition
        4      - change ‘4’ partition
        5      - change ‘5’ partition
        6      - change ‘6’ partition
        expand - expand label to use whole disk
        select - select a predefined table
        modify - modify a predefined partition table
```

```
        name   - name the current table
        print  - display the current table
        label  - write partition map and label to the disk
        !<cmd> - execute <cmd>, then return
        quit
partition> p
Current partition table (original):
Total disk sectors available: 71116508 + 16384 (reserved sectors)

Part      Tag    Flag     First Sector         Size         Last Sector
  0        usr    wm               256      33.91GB          71116541
  1 unassigned    wm                 0          0              0
  2 unassigned    wm                 0          0              0
  3 unassigned    wm                 0          0              0
  4 unassigned    wm                 0          0              0
  5 unassigned    wm                 0          0              0
  6 unassigned    wm                 0          0              0
  8   reserved    wm          71116542       8.00MB         71132925

partition>
```

7 如果磁盘包含标签 EFI，则使用标签 SMI 重新为该磁盘设置标签。

例如，已使用标签 SMI 重新为 `c2t1d0` 磁盘设置标签，但是缺省分区表并不提供最佳分片配置。

```
partition> label
[0] SMI Label
[1] EFI Label
Specify Label type[1]: 0
Auto configuration via format.dat[no]?
Auto configuration via generic SCSI-2[no]?
partition> p
Current partition table (default):
Total disk cylinders available: 24620 + 2 (reserved cylinders)

Part      Tag    Flag     Cylinders         Size            Blocks
  0       root    wm       0 -    90      128.37MB    (91/0/0)      262899
  1       swap    wu      91 -   181      128.37MB    (91/0/0)      262899
  2     backup    wu       0 - 24619       33.92GB    (24620/0/0) 71127180
  3 unassigned    wm       0                  0         (0/0/0)           0
  4 unassigned    wm       0                  0         (0/0/0)           0
  5 unassigned    wm       0                  0         (0/0/0)           0
  6        usr    wm     182 - 24619       33.67GB    (24438/0/0) 70601382
  7 unassigned    wm       0                  0         (0/0/0)           0

partition>
```

8 为 ZFS 根池磁盘创建最佳的分片配置。

设置浮动分区，以使所有未分配的磁盘空间集中在分片 0 中。然后，按下 "return"（返回）键通过分片大小字段来创建一个大的分片 0。

```
partition> modify
Select partitioning base:
        0. Current partition table (default)
        1. All Free Hog
Choose base (enter number) [0]? 1
```

```
Part      Tag    Flag     Cylinders         Size            Blocks
  0       root    wm       0                 0         (0/0/0)            0
  1       swap    wu       0                 0         (0/0/0)            0
  2     backup    wu       0 - 24619       33.92GB    (24620/0/0) 71127180
  3 unassigned    wm       0                 0         (0/0/0)            0
  4 unassigned    wm       0                 0         (0/0/0)            0
  5 unassigned    wm       0                 0         (0/0/0)            0
  6        usr    wm       0                 0         (0/0/0)            0
  7 unassigned    wm       0                 0         (0/0/0)            0

Do you wish to continue creating a new partition
table based on above table[yes]?
Free Hog partition[6]? 0
Enter size of partition '1' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '3' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '4' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '5' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '6' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '7' [0b, 0c, 0.00mb, 0.00gb]:

Part      Tag    Flag     Cylinders         Size            Blocks
  0       root    wm       0 - 24619       33.92GB    (24620/0/0) 71127180
  1       swap    wu       0                 0         (0/0/0)            0
  2     backup    wu       0 - 24619       33.92GB    (24620/0/0) 71127180
  3 unassigned    wm       0                 0         (0/0/0)            0
  4 unassigned    wm       0                 0         (0/0/0)            0
  5 unassigned    wm       0                 0         (0/0/0)            0
  6        usr    wm       0                 0         (0/0/0)            0
  7 unassigned    wm       0                 0         (0/0/0)            0

Okay to make this the current partition table[yes]?
Enter table name (remember quotes): "c2t1d0"

Ready to label disk, continue? yes
partition> quit
format> quit
```

9 使 ZFS 分辨出故障磁盘已替换。

```
# zpool replace rpool c2t1d0s0
# zpool online rpool c2t1d0s0
```

对于某些硬件，插入替换磁盘后不必使其联机。

如果正在附加新磁盘来创建镜像根池或附加较大的磁盘来替换较小的磁盘，请使用与以下语法类似的语法：

```
# zpool attach rpool c0t0d0s0 c1t0d0s0
```

10 如果使用新磁盘替换根池磁盘，请在重新同步新磁盘或替换磁盘之后应用引导块。

例如：

```
# zpool status rpool
installboot -F zfs /usr/platform/'uname -i'/lib/fs/zfs/bootblk /dev/rdsk/c2t1d0s0
```

11 验证您是否可以从新磁盘引导。

12 **如果系统从新磁盘引导，则分离旧磁盘。**

仅在附加新磁盘来替换故障磁盘或较小磁盘时，本步骤才是必要步骤。

```
# zpool detach rpool c0t0d0s0
```

13 **通过使用 SPARC 引导 PROM 中的 eeprom 命令或 setenv 命令，将系统设置为自动从新磁盘引导。**

更多信息 **为 ZFS 根文件系统创建磁盘分片之后...**

为 ZFS 根文件系统创建磁盘分片之后，需要恢复根池快照来恢复根池，请参见《Oracle Solaris 管理：ZFS 文件系统》中的“如何替换 ZFS 根池中的磁盘”。

▼ SPARC: 如何为 ZFS 根文件系统安装引导块

1 **成为管理员。**

2 **为 ZFS 根文件系统安装引导块。**

```
# installboot -F zfs /usr/platform/‘uname -i‘/lib/fs/zfs/bootblk
/dev/rdsk/cwtxdys0
```

有关更多信息，请参见`installboot(1M)`。

3 **通过将系统重新引导至运行 3 级，验证是否安装了引导块。**

```
# init 6
```

示例 12–1 SPARC: 为 ZFS 根文件系统安装引导块

如果以物理形式替换用于根池的磁盘，并且之后重新安装了 Oracle Solaris OS，或者为根池附加新磁盘，则会自动安装引导块。如果用 `zpool replace` 命令替换用于根池的磁盘，那么必须手动安装引导块，以使系统可以从替换磁盘引导。

以下示例说明如何为 ZFS 根文件系统安装引导块。

```
# installboot -F zfs /usr/platform/‘uname -i‘/lib/fs/zfs/bootblk /dev/rdsk/c0t1d0s0
```

▼ SPARC: 如何为 ZFS 文件系统设置磁盘

如果设置结合 ZFS 非根文件系统使用的磁盘，在创建池或将磁盘添加到池中时，会自动重新为该磁盘设置标签。如果使用整个磁盘创建池，或者在将整个磁盘添加到 ZFS 存储池时，则会应用标签 EFI。有关磁盘标签 EFI 的更多信息，请参见第 178 页中的“EFI 磁盘标签”。

通常，大多数现代的总线类型都支持热插拔。这说明可以将磁盘插入空槽中，而且系统识别该磁盘。有关热插拔设备的更多信息，请参见第 6 章，动态配置设备（任务）。

1 **成为管理员。**

2 **将磁盘连接至系统，并检查磁盘的物理连接。**

有关详细信息，请参阅磁盘的硬件安装指南。

3 **使故障磁盘脱机并取消其配置（如有必要）。**

有些硬件要求您在尝试通过 `zpool replace` 操作替换故障磁盘之前，使磁盘脱机并取消其配置。例如：

```
# zpool offline tank c1t1d0
# cfgadm -c unconfigure c1::dsk/c1t1d0
<Physically remove failed disk c1t1d0>
<Physically insert replacement disk c1t1d0>
# cfgadm -c configure c1::dsk/c1t1d0
```

对于某些硬件，插入替换磁盘后不必对其进行重新配置。

4 **确认新磁盘可识别。**

检查 `format` 实用程序的输出来查看该磁盘是否列在 `AVAILABLE DISK SELECTIONS`（可用的磁盘选择）之下。然后，退出 `format` 实用程序。

```
# format
```

5 **使 ZFS 分辨出故障磁盘已替换（如有必要）。**

```
# zpool replace tank c1t1d0
# zpool online tank c1t1d0
```

确认新磁盘可重新同步。

```
# zpool status tank
```

6 **将新磁盘附加至现有的 ZFS 存储池（如有必要）。**

例如：

```
# zpool attach tank mirror c1t0d0 c2t0d0
```

确认新磁盘可重新同步。

```
# zpool status tank
```

有关更多信息，请参见《Oracle Solaris 管理：ZFS 文件系统》中的第 4 章 “管理 Oracle Solaris ZFS 存储池”。

◆ ◆ ◆ 第 13 章

13 x86：设置磁盘（任务）

本章介绍如何在基于 x86 的系统中设置磁盘。

有关与在基于 x86 的系统中设置磁盘关联的过程的信息，请参见以下各节：

- 第 205 页中的 “SPARC: 为 ZFS 文件系统设置磁盘（任务列表）”
- 第 221 页中的 “创建和更改 Solaris fdisk 分区”

有关磁盘管理的概述信息，请参见第 10 章，管理磁盘（概述）。有关在基于 SPARC 的系统中设置磁盘的逐步说明，请参见第 12 章，SPARC：设置磁盘（任务）。

x86: 为 ZFS 文件系统设置磁盘（任务列表）

以下任务列表说明了在基于 x86 的系统中为 ZFS 根文件系统设置 ZFS 根池磁盘的过程。

任务	说明	参考
1. 为 ZFS 根文件系统设置磁盘。	*ZFS* **根文件系统的磁盘** 连接新磁盘或替换现有根池磁盘，并从本地或远程 Oracle Solaris DVD 进行引导。	第 214 页中的 “x86: 如何为 ZFS 根文件系统设置磁盘”
2. 创建或更改 `fdisk` 分区（如有必要）。	磁盘必须包含有效的 Solaris fdisk 分区。	第 221 页中的 “创建和更改 Solaris fdisk 分区”
3. 为 ZFS 根文件系统创建磁盘分片。	为用于 ZFS 根池的磁盘创建磁盘分片。这是长期的引导限制。	第 221 页中的 “x86: 如何创建 Solaris `fdisk` 分区”和第 215 页中的 “x86: 如何为 ZFS 根文件系统创建磁盘分片”

任务	说明	参考
4. 为 ZFS 根文件系统安装引导块。	如果用 `zpool replace` 命令替换用于根池的磁盘，那么必须手动安装引导块，以使系统可以从替换磁盘进行引导。	第 219 页中的 “x86: 如何为 ZFS 根文件系统安装引导块”
5. 为 ZFS 文件系统设置磁盘。	*ZFS* **文件系统的磁盘** 连接磁盘。	第 220 页中的 “x86: 如何为 ZFS 文件系统设置磁盘”

x86: 为 ZFS 文件系统设置磁盘

虽然说明如何设置磁盘和创建 `fdisk` 分区的过程可以结合 ZFS 文件系统使用，但是 ZFS 文件系统不直接映射到磁盘或磁盘分片。必须创建 ZFS 存储池才能创建 ZFS 文件系统。有关更多信息，请参见《Oracle Solaris 管理：ZFS 文件系统》。

根池包含用于引导 Oracle Solaris OS 的根文件系统。如果根池磁盘已损坏，且根池未进行镜像配置，则系统可能不会引导。如果根池磁盘已损坏，有两种可以恢复的方法：

- 可以重新安装整个 Oracle Solaris OS。
- 或者，可以替换根池磁盘并从快照或备份介质中恢复文件系统。可以通过创建冗余根池减少因硬件故障而导致的系统停机时间。唯一受支持的冗余根池配置是镜像根池。

用于非根池中的磁盘通常包含用户或数据文件的空间。为了获得更多磁盘空间，可以向根池或非根池附加或添加另一个磁盘。或者，可以使用以下方法替换池中损坏的磁盘：

- 如果所有的设备当前为 ONLINE（联机），则可以在非冗余池中替换磁盘。
- 如果其他设备中存在足够的冗余，则可以在冗余池中替换磁盘。
- 在镜像根池中，可以替换磁盘或附加磁盘，然后拆离故障磁盘或较小的磁盘，以增加池的大小。

在通常情况下，为系统设置磁盘取决于硬件，所以当在系统中添加或替换磁盘时，请查看硬件文档。如果需要将磁盘添加到现有的控制器中，那么这可能只是将磁盘插入空槽（如果系统支持热插拔）中的问题而已。如果需要配置新的控制器，请参见第 75 页中的 “动态重新配置和热插拔”。

▼ x86: 如何为 ZFS 根文件系统设置磁盘

有关替换磁盘的信息，请参阅硬件安装指南。

1. **断开损坏的磁盘与系统之间的连接（如有必要）。**
2. **将替换磁盘连接至系统，并检查磁盘的物理连接。**

3 **根据从本地 Oracle Solaris DVD 还是从网络中的远程 Oracle Solaris DVD 引导，按照下表中的说明进行操作。**

引导类型	操作
从本地驱动器中的 Oracle Solaris DVD	1. 确保 Oracle Solaris DVD 位于驱动器中。 2. 选择从介质引导的选项。
从网络	3. 选择从网络引导的选项。

更多信息 **为 ZFS 根文件系统设置磁盘之后...**

在连接或替换磁盘之后，创建 `fdisk` 分区。请转至第 221 页中的 “x86: 如何创建 Solaris `fdisk` 分区”。

x86: 为 ZFS 根文件系统创建磁盘分片

必须为用于 ZFS 根池的磁盘创建磁盘分片。这是长期的引导限制。查看以下根池磁盘要求：

- 必须包含磁盘分片和标签 SMI (VTOC)。
- 根池磁盘不支持标签 EFI。
- x86 系统中的根池磁盘必须包含 `fdisk` 分区。
- 必须是单个磁盘或镜像配置的一部分。根池既不支持非冗余配置，也不支持 RAIDZ 配置。
- 作为 OS 映像一部分的根文件系统的所有子目录（`/var` 除外）必须与根文件系统处于同一数据集。
- 除交换和转储设备外，所有其他 Solaris OS 组件必须驻留在根池中。

对于基于 x86 的系统，必须首先创建 `fdisk` 分区。然后创建在分片 0 中具有大量的磁盘空间的磁盘分片。

不推荐尝试使用磁盘上不同的分片并在不同的操作系统中共享该磁盘，或与不同的 ZFS 存储池或存储池组件一同使用。

▼ x86: 如何为 ZFS 根文件系统创建磁盘分片

通常情况下，安装系统时，会自动安装根池磁盘。如果需要替换根池磁盘或将新磁盘附加作为镜像根池磁盘，请参见以下步骤。

有关 `fdisk` 分区的完整说明，请参见第 221 页中的 “x86: 创建 `fdisk` 分区指南”。

1 **成为管理员。**

2 **使故障磁盘脱机并取消其配置（如有必要）。**

有些硬件要求您在尝试通过 zpool replace 操作替换故障磁盘之前，使磁盘脱机并取消其配置。例如：

```
# zpool offline rpool c2t1d0s0
# cfgadm -c unconfigure c2::dsk/c2t1d0
```

3 **以物理形式将新磁盘或替换磁盘连接至系统（如有必要）。**

a. **以物理形式删除故障磁盘。**

b. **以物理形式插入替换磁盘。**

c. **配置替换磁盘（如有必要）。例如：**

```
# cfgadm -c configure c2::dsk/c2t1d0
```

对于某些硬件，插入替换磁盘后不必对其进行重新配置。

4 **通过查看格式化输出确认磁盘可存取。**

例如，format 命令显示 4 个连接至该系统的磁盘。

```
# format -e
AVAILABLE DISK SELECTIONS:
       1. c8t0d0 <Sun-STK RAID INT-V1.0 cyl 17830 alt 2 hd 255 sec 63>
          /pci@0,0/pci10de,375@f/pci108e,286@0/disk@0,0
       2. c8t1d0 <Sun-STK RAID INT-V1.0-136.61GB>
          /pci@0,0/pci10de,375@f/pci108e,286@0/disk@1,0
       3. c8t2d0 <Sun-STK RAID INT-V1.0-136.61GB>
          /pci@0,0/pci10de,375@f/pci108e,286@0/disk@2,0
       4. c8t3d0 <Sun-STK RAID INT-V1.0-136.61GB>
          /pci@0,0/pci10de,375@f/pci108e,286@0/disk@3,0
```

5 **选择用于 ZFS 根池的磁盘。**

```
Specify disk (enter its number): 1
selecting c8t0d0
[disk formatted]
.
.
.
format>
```

6 **查看 fdisk 分区的状态。**

- 如果磁盘不具有 fdisk 分区，您将看到类似于以下内容的消息：

```
format> fdisk
No Solaris fdisk partition found.
```

如果这样，请转至步骤 4 创建 fdisk 分区。

- 如果磁盘具有 EFI fdisk 或某些其他分区类型，请转至步骤 5 创建 Solaris fdisk 分区。
- 如果磁盘具有 Solaris fdisk 分区，请转至步骤 6 为根池创建磁盘分片。

7 如有必要，请通过选择 fdisk 选项创建 Solaris fdisk 分区。

```
format> fdisk
No fdisk table exists. The default partition for the disk is:

  a 100% "SOLARIS System" partition

Type "y" to accept the default partition,  otherwise type "n" to edit the
 partition table. y
```

然后请转至步骤 6 为根池创建磁盘分片。

8 如果磁盘具有 EFI fdisk 分区，那么将需要创建 Solaris fdisk 分区。

如果使用 format 实用程序打印磁盘的分区表，并且看见该分区表指向第一个扇区和大小，那么这就是一个 EFI 分区。将需要按照以下操作创建 Solaris fdisk 分区：

- 从格式化选项中选择 fdisk。

```
# format -e c8t0d0
selecting c8t0d0
[disk formatted]
format> fdisk
```

- 选择选项 3 删除现有 EFI 分区，删除分区。

```
Enter Selection: 3
Specify the partition number to delete (or enter 0 to exit): 1
Are you sure you want to delete partition 1? This will make all files and
programs in this partition inaccessible (type "y" or "n"). y


Partition 1 has been deleted.
```

- 选择选项 1 创建新的 Solaris 分区，创建分区。

```
Enter Selection: 1
Select the partition type to create: 1
Specify the percentage of disk to use for this partition
(or type "c" to specify the size in cylinders). 100
Should this become the active partition? If yes, it  will be activated
each time the computer is reset or turned on.
Please type "y" or "n". y
Partition 1 is now the active partition.
```

- 更新磁盘配置并退出。

```
Enter Selection: 6
format>
```

- 显示 SMI 分区表。如果已应用缺省分区表，那么分片 0 的大小可能为 0，或者可能太小。请参见下一步。

```
format> partition
partition> print
```

9 通过显示分区（分片）信息确认磁盘具有标签 SMI，并查看分片 0 的大小信息。

设置浮动分区，以使所有未分配的磁盘空间集中在分片 0 中。然后，按下 "return"（返回）键通过分片大小字段来创建一个大的分片 0。

```
partition> modify
Select partitioning base:
        0. Current partition table (default)
        1. All Free Hog
Choose base (enter number) [0]? 1
Part      Tag    Flag     Cylinders         Size            Blocks
  0       root    wm       0                 0         (0/0/0)              0
  1       swap    wu       0                 0         (0/0/0)              0
  2     backup    wu       0 - 17829      136.58GB    (17830/0/0) 286438950
  3 unassigned    wm       0                 0         (0/0/0)              0
  4 unassigned    wm       0                 0         (0/0/0)              0
  5 unassigned    wm       0                 0         (0/0/0)              0
  6        usr    wm       0                 0         (0/0/0)              0
  7 unassigned    wm       0                 0         (0/0/0)              0
  8       boot    wu       0 -     0        7.84MB    (1/0/0)          16065
  9 alternates    wm       0                 0         (0/0/0)              0

Do you wish to continue creating a new partition
table based on above table[yes]?
Free Hog partition[6]? 0
Enter size of partition '1' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '3' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '4' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '5' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '6' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '7' [0b, 0c, 0.00mb, 0.00gb]:

Part      Tag    Flag     Cylinders         Size            Blocks
  0       root    wm       1 - 17829      136.58GB    (17829/0/0) 286422885
  1       swap    wu       0                 0         (0/0/0)              0
  2     backup    wu       0 - 17829      136.58GB    (17830/0/0) 286438950
  3 unassigned    wm       0                 0         (0/0/0)              0
  4 unassigned    wm       0                 0         (0/0/0)              0
  5 unassigned    wm       0                 0         (0/0/0)              0
  6        usr    wm       0                 0         (0/0/0)              0
  7 unassigned    wm       0                 0         (0/0/0)              0
  8       boot    wu       0 -     0        7.84MB    (1/0/0)          16065
  9 alternates    wm       0                 0         (0/0/0)              0
Do you wish to continue creating a new partition
table based on above table[yes]? yes
Enter table name (remember quotes): "c8t0d0"

Ready to label disk, continue? yes
```

10 使 ZFS 分辨出故障磁盘已替换。

```
# zpool replace rpool c2t1d0s0
# zpool online rpool c2t1d0s0
```

对于某些硬件，插入替换磁盘后不必使其联机。

如果正在附加新磁盘来创建镜像根池或附加较大的磁盘来替换较小的磁盘，请使用与以下语法类似的语法：

```
# zpool attach rpool c0t0d0s0 c1t0d0s0
```

11 **如果使用新磁盘替换根池磁盘，请应用引导块。**

例如：

```
# installgrub /boot/grub/stage1 /boot/grub/stage2 /dev/rdsk/c8t0d0s0
```

12 **验证您是否可以从新磁盘引导。**

13 **如果系统从新磁盘引导，则分离旧磁盘。**

仅在附加新磁盘来替换故障磁盘或较小磁盘时，本步骤才是必要步骤。

```
# zpool detach rpool c0t0d0s0
```

14 **通过重新配置系统的 BIOS，设置系统自动从新磁盘引导。**

更多信息

为 ZFS 根文件系统创建磁盘分片之后...

为 ZFS 根文件系统创建磁盘分片之后，需要恢复根池快照来恢复根池，请参见《Oracle Solaris 管理：ZFS 文件系统》中的“如何替换 ZFS 根池中的磁盘”。

▼ x86: 如何为 ZFS 根文件系统安装引导块

1 **成为管理员。**

2 **在系统磁盘中安装引导块。**

```
# installgrub /boot/grub/stage1 /boot/grub/stage2 /dev/rdsk/cwtxdysz
```

有关更多信息，请参见`installgrub(1M)`。

3 **通过将系统重新引导至运行 3 级，验证是否安装了引导块。**

```
# init 6
```

示例 13–1

x86: 为 ZFS 根文件系统安装引导块

如果以物理形式替换用于根池的磁盘，并且之后重新安装了 Oracle Solaris OS，或者为根池附加新磁盘，则会自动安装引导块。如果用 `zpool replace` 命令替换用于根池的磁盘，那么必须手动安装引导块，以使系统可以从替换磁盘引导。

以下示例说明如何为 ZFS 根文件系统安装引导块。

```
# installgrub /boot/grub/stage1 /boot/grub/stage2 /dev/rdsk/c1d0s0
stage2 written to partition 0, 277 sectors starting at 50 (abs 16115)
stage1 written to partition 0 sector 0 (abs 16065)
```

▼ x86: 如何为 ZFS 文件系统设置磁盘

如果设置结合 ZFS 非根文件系统使用的磁盘，在创建池或将磁盘添加到池中时，会自动重新为该磁盘设置标签。如果使用整个磁盘创建池，或者在将整个磁盘添加到 ZFS 存储池时，则会应用标签 EFI。有关磁盘标签 EFI 的更多信息，请参见第 178 页中的 “EFI 磁盘标签”。

通常，大多数现代的总线类型都支持热插拔。这说明可以将磁盘插入空槽中，而且系统识别该磁盘。有关热插拔设备的更多信息，请参见第 6 章，动态配置设备（任务）。

1. **成为管理员。**

   有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。

2. **将磁盘连接至系统，并检查磁盘的物理连接。**

   有关详细信息，请参阅磁盘的硬件安装指南。

3. **使故障磁盘脱机并取消其配置（如有必要）。**

   有些硬件要求您在尝试通过 `zpool replace` 操作替换故障磁盘之前，使磁盘脱机并取消其配置。例如：

   ```
   # zpool offline tank c1t1d0
   # cfgadm -c unconfigure c1::dsk/c1t1d0
   <Physically remove failed disk c1t1d0>
   <Physically insert replacement disk c1t1d0>
   # cfgadm -c configure c1::dsk/c1t1d0
   ```

   对于某些硬件，插入替换磁盘后不必对其进行重新配置。

4. **确认新磁盘可识别。**

   检查 `format` 实用程序的输出来查看该磁盘是否列在 `AVAILABLE DISK SELECTIONS`（可用的磁盘选择）之下。然后，退出 `format` 实用程序。

   ```
   # format
   ```

5. **使 ZFS 分辨出故障磁盘已替换（如有必要）。**

   ```
   # zpool replace tank c1t1d0
   # zpool online tank c1t1d0
   ```

   确认新磁盘可重新同步。

   ```
   # zpool status tank
   ```

6. **将新磁盘附加至现有的 ZFS 存储池（如有必要）。**

   例如：

   ```
   # zpool attach tank mirror c1t0d0 c2t0d0
   ```

确认新磁盘可重新同步。

```
# zpool status tank
```

有关更多信息，请参见《Oracle Solaris 管理：ZFS 文件系统》中的第 4 章 “管理 Oracle Solaris ZFS 存储池”。

创建和更改 Solaris fdisk 分区

有关创建或更改 Solaris `fdisk` 分区的指南和示例，请查看以下各节。

x86: 创建 fdisk 分区指南

当设置一个或多个 `fdisk` 分区时，请遵循这些指南。

- 磁盘最多可以分成四个 `fdisk` 分区。其中之一必须是 Solaris 分区。
- Solaris 分区必须在磁盘中为活动分区。活动分区是指其操作系统在缺省情况下在系统启动时引导的分区。
- Solaris `fdisk` 分区必须在柱面边界开始。
- Solaris `fdisk` 分区必须在第一个磁盘的柱面 1（而不是柱面 0）开始，因为附加引导信息（包括主引导记录）写在扇区 0 中。
- Solaris `fdisk` 分区可以为整个磁盘。或者，可能需要使其更小，以便为 DOS 分区留出空间。可能还需要在磁盘上创建新的 `fdisk` 分区，而不扰乱现有分区（如果有足够的空间），以创建新的分区。

仅适用于 x86 – Solaris 分片也称为分区。某些接口可能将**分片**称为**分区**。

仅在基于 x86 的系统中支持 `fdisk` 分区。为了避免混淆，Oracle Solaris 文档尝试区分 `fdisk` 分区和 Solaris `fdisk` 分区内的实体。这些项可能称为分片或分区。

▼ x86: 如何创建 Solaris fdisk 分区

开始之前 如果需要有关 `fdisk` 分区的信息，请参见第 221 页中的 “x86: 创建 `fdisk` 分区指南”。

1. **成为管理员。**

   有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。

2. **调用 `format` 实用程序。**

   ```
   # format
   ```

   显示磁盘的编号列表。

有关更多信息，请参见format(1M)。

3 **键入创建 Solaris fdisk 分区的磁盘编号。**

```
Specify disk (enter its number): disk-number
```

其中 *disk-number* 是希望创建 Solaris fdisk 分区的磁盘编号。

4 **选择 fdisk 菜单。**

```
format> fdisk
```

所显示的 fdisk 菜单取决于磁盘是否存在现有 fdisk 分区。

5 **通过在提示符下指定 y，创建并激活跨整个磁盘的 Solaris fdisk 分区。然后，转至步骤 13。**

```
No fdisk table exists. The default partition for the disk is:

a 100% "SOLARIS System" partition

Type "y" to accept the default partition, otherwise type "n" to edit the
partition table.
y
```

6 **如果不希望 Solaris fdisk 分区跨整个磁盘，请在提示符下指定 n。**

```
Type "y" to accept the default partition, otherwise type "n" to edit the
 partition table.
n
             Total disk size is 17848 cylinders
             Cylinder size is 16065 (512 byte) blocks

                                               Cylinders
      Partition   Status    Type          Start   End   Length    %
      =========   ======    ============  =====   ===   ======   ===



SELECT ONE OF THE FOLLOWING:
   1. Create a partition
   2. Specify the active partition
   3. Delete a partition
   4. Change between Solaris and Solaris2 Partition IDs
   5. Edit/View extended partitions
   6. Exit (update disk configuration and exit)
   7. Cancel (exit without updating disk configuration)
Enter Selection:
```

7 **选择选项 1 Create a partition 以创建 fdisk 分区。**

```
Enter Selection: 1
```

8 **通过选择 1(=Solaris2) 创建 Solaris fdisk 分区。**

```
Select the partition type to create:
   1=SOLARIS2   2=UNIX      3=PCIXOS     4=Other        5=DOS12
   6=DOS16      7=DOSEXT    8=DOSBIG     9=DOS16LBA     A=x86 Boot
```

```
   B=Diagnostic C=FAT32     D=FAT32LBA    E=DOSEXTLBA    F=EFI (Protective)
   G=EFI_SYS    0=Exit?
```

9 **确定要为 Solaris `fdisk` 分区保留的磁盘百分比。计算该百分比时，请记住任何现有 `fdisk` 分区的大小。**

```
Specify the percentage of disk to use for this partition
(or type "c" to specify the size in cylinders). nn
```

10 **通过在提示符下键入 `y`，激活 Solaris `fdisk` 分区。**

```
Should this become the active partition? If yes, it will be
activated each time the computer is reset or turned on.
Please type "y" or "n". y
```

激活 `fdisk` 分区之后，将显示提示 "`Enter Selection`"（输入选择）。

11 **选择选项 1 `Create a partition` 以创建另一个 `fdisk` 分区。**

有关创建 `fdisk` 分区的说明，请参见步骤 8–10。

12 **更新磁盘配置，然后从 `selection` 菜单中退出 `fdisk` 菜单。**

```
Selection: 6
```

13 **使用 `label` 命令重新为磁盘设置标签。**

```
format> label
Ready to label disk, continue? yes
format>
```

14 **退出 `format` 实用程序。**

```
format> quit
```

示例 13–2 x86: 创建一个跨整个驱动器的 Solaris fdisk 分区

以下示例使用 `format` 实用程序的 `fdisk` 选项创建一个跨整个驱动器的 Solaris `fdisk` 分区。

```
# format
Searching for disks...done
AVAILABLE DISK SELECTIONS:
       0. c0d0 <DEFAULT cyl 2466 alt 2 hd 16 sec 63>
          /pci@0,0/pci-ide@7,1/ide@0/cmdk@0,0
       1. c0d1 <DEFAULT cyl 522 alt 2 hd 32 sec 63>
          /pci@0,0/pci-ide@7,1/ide@0/cmdk@1,0
       2. c1d0 <DEFAULT cyl 13102 alt 2 hd 16 sec 63>
          /pci@0,0/pci-ide@7,1/ide@1/cmdk@0,0
Specify disk (enter its number): 0
selecting c0d0
Controller working list found
[disk formatted]
format> fdisk
No fdisk table exists. The default partitioning for your disk is:
```

```
  a 100% "SOLARIS System" partition.

Type "y" to accept the default partition, otherwise type "n" to edit the
partition table. y

format> label
Ready to label disk, continue? yes
format> quit
```

更多信息 **创建 Solaris fdisk 分区之后...**

在磁盘中创建 Solaris `fdisk` 分区之后，可以将该分区用于根池磁盘或非根池磁盘。有关更多信息，请参见第 214 页中的 “x86: 为 ZFS 文件系统设置磁盘”。

更改 fdisk 分区标识符

x86 系统中的 Solaris `fdisk` 分区标识符已从130 (`0x82`) 更改为 191 (`0xbf`)。所有的 Oracle Solaris 命令、实用程序和驱动程序已更新为适用于任一 `fdisk` 标识符。`fdisk` 功能没有任何更改。

▼ 如何更改 Solaris fdisk 标识符

新的 `fdisk` 菜单选项使您能够在新标识符和旧标识符之间来回切换。甚至当挂载分区中所包含的文件系统时，也可以更改 `fdisk` 标识符。

`fdisk` 菜单中的两个 `type` 值反映旧标识符和新标识符，如下：

- `Solaris` 标识 `0x82`
- `Solaris2` 标识 `0xbf`

1 **成为管理员。**

2 **显示当前的 `fdisk` 标识符。**

例如：

```
          Total disk size is 39890 cylinders
          Cylinder size is 4032 (512 byte) blocks

                                             Cylinders
    Partition   Status    Type          Start   End   Length    %
    =========   ======    ============  =====   ===   ======   ===
        1       Active    x86 Boot          1     6       6      0
        2                 Solaris2          7  39889    39883    100
```

3 **从 `fdisk` 菜单中选择选项 `4`，以将 `fdisk` 分区标识符更改回 `0x82`。**

```
SELECT ONE OF THE FOLLOWING:
  1. Create a partition
  2. Specify the active partition
```

```
  3. Delete a partition
  4. Change between Solaris and Solaris2 Partition IDs
  5. Exit (update disk configuration and exit)
  6. Cancel (exit without updating disk configuration)
Enter Selection: 4
```

4 选择选项 **5**，以更新磁盘配置，然后退出。

5 如有必要，从 **fdisk** 菜单中选择选项 **4**，以将 **fdisk** 分区标识符更改回 **0xbf**。

例如：

```
Total disk size is 39890 cylinders
            Cylinder size is 4032 (512 byte) blocks

                                             Cylinders
     Partition   Status    Type          Start   End   Length    %
     =========   ======    ============  =====   ===   ======   ===
         1       Active    x86 Boot          1     6       6      0
         2                 Solaris           7  39889    39883    100

SELECT ONE OF THE FOLLOWING:
  1. Create a partition
  2. Specify the active partition
  3. Delete a partition
  4. Change between Solaris and Solaris2 Partition IDs
  5. Exit (update disk configuration and exit)
  6. Cancel (exit without updating disk configuration)
Enter Selection: 4
```

6 选择选项 **5**，以更新磁盘配置，然后退出。

◆ ◆ ◆ 第 14 章

使用 COMSTAR 配置存储设备

本章介绍如何配置通用多协议 SCSI 目标 (Common Multiprotocol SCSI TARget, COMSTAR)。COMSTAR 是一种软件框架，可用于将任何 Oracle Solaris 11 主机转换为 SCSI 目标设备，以便启动器主机通过存储网络可访问 SCSI 目标设备。

这意味着可以使系统上的存储设备用于 Linux、Mac OS 或 Windows 客户机系统，就像它们是本地存储设备一样。支持的存储协议为 iSCSI、FC、iSER 和 SRP。

有关 Oracle Solaris 中 iSNS 支持的信息，请参见第 15 章，配置和管理 Oracle Solaris Internet 存储名称服务 (Internet Storage Name Service, iSNS)。

有关如何解决 Oracle Solaris 中 iSCSI 配置问题的信息，请参见第 266 页中的 “iSCSI 配置问题的故障排除”。

COMSTAR 和 iSCSI 技术（概述）

iSCSI 是 Internet SCSI（Small Computer System Interface，小型计算机系统接口）的首字母缩略词，是用于链接数据存储子系统的基于 Internet 协议 (Internet Protocol, IP) 的存储网络标准。

通过在 IP 网络上传输 SCSI 命令，iSCSI 协议可用于通过网络访问块设备，就像这些设备连接至本地系统一样。COMSTAR 提供了一种更为简便的方法管理这些 iSCSI **目标设备**。

COMSTAR 利用一种 SCSI 目标模式框架 (SCSI Target Mode Framework, STMF) 管理具有以下组件的目标存储设备：

- 端口提供者（或插件）－实现协议，如光纤通道 (Fibre Channel, FC) 和 iSCSI。
- 逻辑单元提供者－模仿各种 SCSI 设备，如磁盘和磁带设备。

- `libstmf` 管理库－提供 COMSTAR 管理接口。实现 iSCSI 功能的模块与底层传输不直接交互。同样，实现传输协议的模块无法识别它们所传输的数据包中固有的 SCSI 级别功能。一些传输示例包括光纤通道和 iSCSI。该框架将 SCSI 命令和相关资源的执行和清除分离开来。该分离可简化编写 SCSI 或传输模块的任务。

  使用以下命令可管理这些功能：

  - `itadm` 命令管理 SCSI 目标模式框架内的 Internet SCSI (iSCSI) 节点。
  - `stmfadm` 命令配置 SCSI 目标模式框架内的逻辑单元。
  - `srptadm` 命令管理 SCSI 目标模式框架内的 SCSI RDMA 协议 (SCSI RDMA Protocol, SRP) 目标端口。

如果要使用现有 TCP/IP 网络中的存储设备，则可以使用以下解决方法：

- iSCSI 块设备或磁带－将 SCSI 命令和数据从块级别转换为 IP 数据包。如果需要在一个系统与目标设备（如磁带设备或数据库）之间进行块级别访问，则在网络中使用 iSCSI 很有优势。对块级别设备的访问不会被锁定，因此您可以让多个用户或系统访问块级别设备（如 iSCSI 目标设备）。
- NFS－通过 IP 传输文件数据。在网络中使用 NFS 的优点在于可以在多个系统之间共享文件数据。当很多用户访问 NFS 环境中可用的数据时，对文件数据的访问就会相应地被锁定。

以下是在 Oracle Solaris 中使用 iSCSI 目标和启动器的优点：

- iSCSI 协议可以跨现有以太网运行。
  - 可以使用任何支持的网络接口卡 (network interface card, NIC)、以太网集线器或以太网交换机。
  - 一个 IP 端口可以处理多个 iSCSI 目标设备。
  - 可以对 IP 网络使用现有的基础结构和管理工具。
- 可能具有可连接至客户机的现有光纤通道设备，而无需光纤通道 HBA 成本。此外，具有专用阵列的系统目前可以使用 Oracle Solaris ZFS 或 UFS 文件系统导出已复制的存储。
- 可以使用该协议连接至光纤通道或包含适当硬件的 iSCSI 存储区域网络 (Storage Area Network, SAN) 环境。

以下是在 Oracle Solaris 中使用 iSCSI 启动器软件时目前具有的限制或约束：

- 当前不支持使用 SLP 的 iSCSI 设备。
- 不能将 iSCSI 目标配置为转储设备。
- 在现有网络上传输大量数据可能对性能造成影响。

COMSTAR 软件和硬件要求

- Oracle Solaris 存储软件和设备
- 适用于提供存储设备的系统的 `group/feature/storage-server` 软件包
- 任何支持的 NIC

配置 COMSTAR（任务列表）

任务	说明	参考
确定 COMSTAR 软件和硬件要求。	确定使用 COMSTAR 设置 iSCSI 存储网络的软件和硬件要求。	第 229 页中的 “COMSTAR 软件和硬件要求”
确定 iSCSI 目标搜索方法。	确定最适合您环境的 iSCSI 目标搜索方法。	第 231 页中的 “配置动态或静态目标搜索”
启用 STMF 服务。	启用提供持续目标信息的 STMF 服务。	
创建 SCSI 逻辑单元并使其可用。	创建 SCSI 逻辑单元 (logical unit, LUN) 并使其可用于 iSCSI 或 iSER 配置的所有主机或特定主机。	第 233 页中的 “如何创建 iSCSI LUN”
配置 iSCSI 目标。	配置 iSCSI 存储组件的 iSCSI 目标。	第 235 页中的 “如何创建 iSCSI 目标”
配置 iSCSI 启动器。	配置针对 iSCSI 目标启动 SCSI 请求的一个或多个系统。	第 236 页中的 “如何配置 iSCSI 启动器”
访问 iSCSI 磁盘。	可以使用 `format` 实用程序访问您的 iSCSI 磁盘。还可以在系统重新引导之后使得 iSCSI 磁盘自动可用。	第 240 页中的 “如何访问 iSCSI 磁盘”
限制 LUN 访问选定的系统。	可能需要限制 LUN 访问网络中的特定系统。	第 242 页中的 “如何限制 LUN 访问选中的系统”
配置光纤通道设备。	如果在您的环境中具有 FC 存储阵列，请使用 COMSTAR 配置 FC 设备。	第 243 页中的 “使用 COMSTAR 配置光纤通道设备”
配置 FCoE 设备。	使用 COMSTAR 配置以太网光纤通道 (Fibre Channel over Ethernet, FCoE) 设备。 FCoE 功能是通过以太网接口提供的。FCoE 端口是与以太网接口相关联的逻辑实体。	第 247 页中的 “使用 COMSTAR 配置 FCoE 设备”

任务	说明	参考
配置 SRP 设备。	使用 COMSTAR 配置 SRP 设备。 SRP（SCSI RDMA Protocol，SCSI RDMA 协议）通过将 SCSI 数据传输阶段映射到 Infiniband (IB) 远程直接内存访问 (Remote Direct Memory Access, RDMA) 操作，加速了 SCSI 协议。	第 250 页中的 “使用 COMSTAR 配置 SRP 设备”
（可选的）在 Oracle Solaris iSCSI 配置中设置验证。	确定是否要在 Oracle Solaris iSCSI 配置中使用验证： 考虑使用单向 CHAP 还是双向 CHAP。 考虑使用第三方 RADIUS 服务器简化 CHAP 管理。	 第 252 页中的 “如何配置对 iSCSI 启动器的 CHAP 验证” 第 253 页中的 “如何配置对 iSCSI 目标的 CHAP 验证” 第 254 页中的 “如何为 iSCSI 目标配置 RADIUS 服务器”
监视 iSCSI 配置。	使用 `iscsiadm` 命令监视 iSCSI 配置。	第 259 页中的 “如何显示 iSCSI 配置信息”
（可选的）修改 iSCSI 配置。	可能需要修改 iSCSI 目标参数，如头摘要和数据摘要参数。	第 263 页中的 “如何修改 iSCSI 启动器和目标参数”

配置 COMSTAR

使用 COMSTAR 配置 iSCSI 目标和启动器包含以下任务：

- 确定硬件和软件要求
- 配置 IP 网络
- 连接并设置 iSCSI 目标设备
- 配置启动器
- 配置 iSCSI 目标搜索方法
- 在 iSCSI 磁盘上创建文件系统
- （可选的）在 iSCSI 启动器与 iSCSI 目标之间配置 iSCSI 验证
- 监视 iSCSI 配置

iSCSI 配置信息存储在 `/etc/iscsi` 目录中，但是此信息不需要进行手动管理。

COMSTAR 术语

配置 iSCSI 目标和启动器之前，请查看以下术语。

术语	说明
搜索	搜索是指为启动器提供可用目标列表的过程。
搜索方法	指可找到 iSCSI 目标的方法。当前有三种可用方法： ■ Internet 存储名称服务 (Internet Storage Name Service, iSNS) － 通过与一个或多个 iSNS 服务器进行交互来搜索潜在目标。 ■ SendTargets － 通过使用 *discovery-address* 来搜索潜在目标。 ■ 静态 － 配置了静态目标寻址。
启动器	针对 iSCSI 目标启动 SCSI 请求的驱动程序。
启动器组	一组启动器。当启动器组与 LUN 相关联时，只有该组的启动器可以访问 LUN。
`iqn` 或 `eui` 地址格式	`iqn`（iSCSI qualified name，iSCSI 限定名）地址是 iSCSI 网络中的设备的唯一标识符，采用 *iqn.date.authority:uniqueid* 形式。当初始化 iSCSI 启动器或目标时，将自动为 iSCSI 启动器或目标指定一个 IQN 名称。 `eui`（extended unique identifier，扩展唯一标识符）地址由 16 个十六进制数字组成，可用于标识一类既可在 SCSI 标准中又可在 InfiniBand 标准中使用的 GUID。SRP 设备使用 `eui` 地址格式。
逻辑单元	存储系统中的唯一编号组件。也称为 LUN。当 LUN 与一个或多个 SCSI 目标相关联时，该目标可以供一个或多个 SCSI 启动器访问。
目标设备	iSCSI 存储组件。
目标组	一组目标。LUN 可用于一个目标组中的所有目标。
目标门户组	确定特定 iSCSI 目标将侦听哪些接口的 IP 地址列表。TPG 包含 IP 地址和 TCP 端口号。

配置动态或静态目标搜索

确定是要配置其中一种动态设备搜索方法，还是要使用静态 iSCSI 启动器目标来执行设备搜索。

- **动态设备搜索** － 有两种可用的动态设备搜索方法：
  - SendTargets － 如果 iSCSI 节点显示了大量目标，如 iSCSI 到光纤通道网桥，则可以提供 iSCSI 节点 IP 地址/端口组合，并允许 iSCSI 启动器使用 SendTargets 功能执行设备搜索。
  - iSNS － Internet 存储名称服务 (Internet Storage Name Service, iSNS) 允许 iSCSI 启动器搜索其有权访问的使用尽可能少的配置信息的目标。另外，还提供状态更改通知功能，在存储节点的操作状态发生更改时通知 iSCSI 启动器。要使用 iSNS 搜索选项，可以提供 iSNS 服务器地址/端口组合，并允许 iSCSI 启动器查询为执行设备搜索而指定的 iSNS 服务器。iSNS 服务器的缺省端口为 `3205`。有关 iSNS 的更多信息，请参见 RFC 4171：

http://www.ietf.org/rfc/rfc4171.txt

iSNS 搜索服务提供一种管理模型，用来搜索网络上的所有目标。

有关设置 Oracle Solaris 中 iSNS 支持的更多信息，请参见第 15 章，配置和管理 Oracle Solaris Internet 存储名称服务 (Internet Storage Name Service, iSNS)。

- **静态设备搜索** – 如果 iSCSI 节点具有较少目标，或者如果要限制启动器尝试访问的目标，则可以使用以下静态目标地址命名约定以静态方式配置 *target-name* :

  target,*target-address*[: *port-number*]

  可以使用数组的管理工具来确定静态目标地址。

注 – 请勿将 iSCSI 目标配置为同时使用静态和动态设备搜索方法进行搜索。使用冗余搜索方法的后果是可能会在与 iSCSI 目标设备进行通信时减缓系统性能。

使用 COMSTAR 配置 iSCSI 设备

可以设置并配置一个 COMSTAR Internet SCSI (iSCSI) 目标并使其可用于网络。iSCSI 功能可使用标准的 iSCSI 协议完成正常的 Internet 连接（如以太网）。iSCSI 协议还提供命名和搜索服务、使用 CHAP 和 RADIUS 的验证服务以及通过 iSNS 的集中管理。

如果系统至少有一个 InfiniBand (IB) 主机通道适配器 (Host Channel Adapter, HCA) 和连通性存在于启动器和目标之间，则 iSCSI 连接使用 RDMA (iSER) iSCSI Extensions 以获得增强的数据吞吐量。与 IB HCA 一起使用的 iSER 提供高带宽、低 CPU 利用率和多协议可共享的单一网络连接。

iSER 目标和启动器均使用称为 iSCSI Data Mover 的组件，提供 iSCSI 连接服务。不论何时启动器和目标均被配置为使用具备 IB 功能设备对应的 IP 地址，iSER 的使用都是自动的。

▼ 如何启用 STMF 服务

COMSTAR 使用 SMF 存储其当前和持久配置，如逻辑单元映射、主机组定义和目标组定义。当在引导过程中启用服务或当使用 `svcadm` 命令时，它会清除内核框架内任何过时的配置数据，然后将配置从 SMF 系统信息库中重新载入到驱动程序。配置加载完成后，对配置所做的任何更改将自动更新到驱动程序数据库内，以及 SMF 系统信息库内。例如，通过 `stmfadm` 命令所做的任何更改将自动更新到以上两个区域内。

COMSTAR 目标模式框架作为 `stmf` 服务运行。缺省情况下禁用此服务。您必须启用此服务以使用 COMSTAR 功能。可以使用 `svcs` 命令识别此服务。如果您自安装 `group/feature/storage-server` 软件包之后还未重新引导服务器，则此服务可能未正确启用。

1 **安装 COMSTAR 存储服务器软件。**

```
target# pkg install group/feature/storage-server
              Packages to install:    75
          Create boot environment:    No
              Services to restart:     7
DOWNLOAD                                  PKGS       FILES    XFER (MB)
Completed                                 75/75   9555/9555  105.7/105.7

PHASE                                       ACTIONS
Install Phase                             13347/13347

PHASE                                         ITEMS
Package State Update Phase                    75/75
Image State Update Phase                        2/2
Loading smf(5) service descriptions: 17/17
Loading smf(5) service descriptions: 3/3

PHASE                                         ITEMS
Reading Existing Index                          8/8
Indexing Packages                             75/75
Indexing Packages                             75/75
Optimizing Index...

PHASE                                         ITEMS
Indexing Packages                           573/573
```

2 **重新引导系统或启用 `stmf` 服务。**

```
target# svcadm enable stmf
# svcs stmf
STATE          STIME    FMRI
online         09:42:32 svc:/system/stmf:default
```

▼ 如何备份和存储 COMSTAR 配置

完成 COMSTAR 配置之后，复制一份可恢复的副本（如果需要）。

1 **成为管理员。**

2 **导出当前的 COMSTAR 配置。**

```
# svccfg export -a stmf > COMSTAR.backup
```

3 **如有必要，恢复已导出的配置。**

```
# svccfg import COMSTAR.backup
```

▼ 如何创建 iSCSI LUN

创建磁盘类型 LUN 的逻辑单元提供程序被称为 `sbd`。但是，必须初始化逻辑单元存储，才可以共享磁盘类型 LUN。

服务器提供的磁盘卷被称为**目标**。当 LUN 与 iSCSI 目标相关联时，可由 iSCSI 启动器访问。

创建 SCSI LUN 的过程如下：

- 初始化 LUN 存储，又称为**后备存储**。
- 使用后备存储创建 SCSI LUN。

创建 LUN 时，为其分配全球唯一标识符 (global unique identifier, GUID)，例如 `600144F0B5418B0000004DDAC7C10001`。GUID 用来在随后的任务中引用 LUN，如映射 LUN 来选择主机。

以下步骤在提供存储设备的系统上完成。

1 **创建 ZFS 存储池。**

```
target# zpool create sanpool mirror c2t3d0 c2t4d0
```

2 **创建 ZFS 卷作为 SCSI LUN 使用。**

```
target# zfs create -V 2g sanpool/vol1
```

3 **为 ZFS 卷创建 LUN。**

```
target# stmfadm create-lu /dev/zvol/rdsk/sanpool/vol1
Logical unit created: 600144F0B5418B0000004DDAC7C10001
```

可以在 `/dev/zvol/rdsk/`*pool-name*`/` 目录中找到 ZFS 卷的设备路径。

4 **确认已创建 LUN。**

```
target# stmfadm list-lu
LU Name: 600144F0B5418B0000004DDAC7C10001
```

5 **添加 LUN 视图。**

此命令使 LUN 可访问所有系统。

```
target# stmfadm add-view 600144F0B5418B0000004DDAC7C10001
```

如果您想限制到特定系统上的 LUN 视图，请参见第 242 页中的 “如何限制 LUN 访问选中的系统”。

6 **验证 LUN 配置。**

```
target# stmfadm list-view -l 600144F0B5418B0000004DDAC7C10001
View Entry: 0
    Host group   : All
    Target group : All
    LUN          : 0
```

▼ 如何创建 iSCSI 目标

此过程假定登录到本地系统将包含 iSCSI 目标。

1 启用 iSCSI 目标服务。

```
target# svcadm enable -r svc:/network/iscsi/target:default
```

确认服务已启用。

```
target# svcs -l iscsi/target
fmri         svc:/network/iscsi/target:default
name         iscsi target
enabled      true
state        online
next_state   none
state_time   Mon May 23 14:48:59 2011
logfile      /var/svc/log/network-iscsi-target:default.log
restarter    svc:/system/svc/restarter:default
dependency   require_any/error svc:/milestone/network (online)
dependency   require_all/none svc:/system/stmf:default (online)
```

2 创建 iSCSI 目标。

```
target# itadm create-target
Target iqn.1986-03.com.sun:02:73d12edc-9bb9-cb44-efc4-c3b36c039405 successfully created
```

3 显示 iSCSI 目标信息。

```
target# itadm list-target -v
TARGET NAME                                                  STATE    SESSIONS
iqn.1986-03.com.sun:02:73d12edc-9bb9-cb44-efc4-c3b36c039405  online   0
        alias:                  -
        auth:                   none (defaults)
        targetchapuser:         -
        targetchapsecret:       unset
        tpg-tags:               default
```

▼ 如何为 iSER 配置 IB HCA

InfiniBand (IB) 主机通道适配器 (Host Channel Adapter, HCA) 需要充分利用 iSCSI Extensions 以获得 RDMA (iSER) 功能。要使用 iSER，必须同时配置 HCA 的目标和启动器。

1 成为主机（启动器）系统的管理员。

2 将 HCA 连接到 IB 交换机。

详细信息请参见供应商文档。

3 **为 HCA 配置目标和启动器。**

目标和启动器必须位于同一子网。本示例使用 `ibd0` 作为驱动程序。

```
# ipadm create-addr ibd0
```

4 **为 HCA 配置 IP 地址和端口组合。**

```
# ipadm create-addr -T static -a local=10.1.190.141/24 ibd0/v4addr
```

5 **验证接口配置。**

```
# ipadm show-addr
ADDROBJ           TYPE     STATE        ADDR
lo0/v4            static   ok           127.0.0.1/8
e1000g0/_b        dhcp     ok           10.1.190.141/24
lo0/v6            static   ok           ::1/128
e1000g0/_a        addrconf ok           fe80::214:4fff:fe27:360c/10
```

6 **成为目标系统的管理员，并针对网络上的所有其他 HCA 主机重复步骤 3-5。**

7 **验证目标和启动器的连通性。**

```
target# ping initiator-ip
initiator# ping target-ip
```

▼ 如何配置 iSCSI 启动器

启动器配置过程的一部分是确定 iSCSI 目标搜索方法，提供了具有可用目标列表的启动器。您可以通过静态、SendTargets 或 iSNS 动态搜索来配置 iSCSI 目标。使用 SendTargets 选项的动态搜索是访问大量目标的 iSCSI 启动器的最佳配置，如通过 iSCSI 到光纤通道网桥。SendTargets 动态搜索需要 iSCSI 目标的 IP 地址和端口组合，以使 iSCSI 启动器执行目标搜索。最通用的搜索方法是 SendTargets。

配置目标搜索方法时，必须提供以下信息，这取决于选择哪一种方法：

- SendTargets－目标 IP 地址
- iSNS－iSNS 服务器地址
- 静态－目标 IP 地址和目标名称

有关配置目标搜索方法的更多信息，请参见第 231 页中的 “配置动态或静态目标搜索”。

1 **启用 iSCSI 启动器服务。**

```
initiator# svcadm enable network/iscsi/initiator
```

2 **登录到提供目标的服务器的同时，验证目标名称和 IP 地址。**

```
target# ipadm show-addr
ADDROBJ           TYPE     STATE        ADDR
lo0/v4            static   ok           127.0.0.1/8
e1000g0/_b        dhcp     ok           10.80.227.189/24
lo0/v6            static   ok           ::1/128
```

```
e1000g0/_a        addrconf ok           fe80::214:4fff:fe27:360c/10
target# itadm list-target -v
TARGET NAME                                                  STATE    SESSIONS
iqn.1986-03.com.sun:02:73d12edc-9bb9-cb44-efc4-c3b36c039405  online   0
        alias:                  -
        auth:                   none (defaults)
        targetchapuser:         -
        targetchapsecret:       unset
        tpg-tags:               default
```

3 **配置要静态搜索的目标。**

```
initiator# iscsiadm add static-config iqn.1986-03.com.sun:02:73d12edc-9bb9-cb44-efc4-c3b36c039405,
10.80.227.189
```

4 **查看静态配置信息。**

```
initiator# iscsiadm list static-config
Static Configuration Target: iqn.1986-03.com.sun:02:73d12edc-9bb9-cb44-efc4-c3b36c039405,
10.80.227.189:3260
```

启用搜索方法之前，不启动 iSCSI 连接。请参见下一步。

5 **配置以下目标搜索方法之一：**

- 如果已配置动态搜索 (SendTargets) 目标，请配置 SendTargets 搜索方法。

  ```
  initiator# iscsiadm add discovery-address 10.80.227.189
  ```

- 如果已配置动态搜索 (iSNS) 目标，请配置 iSNS 搜索方法。

  ```
  initiator# iscsiadm add isns-server 10.80.227.189
  ```

6 **启用以下目标搜索方法之一：**

- 如果已配置动态搜索 (SendTargets) 目标，请启用 SendTargets 搜索方法。

  ```
  initiator# iscsiadm modify discovery --sendtargets enable
  ```

- 如果已配置动态搜索 (iSNS) 目标，请启用 iSNS 搜索方法。

  ```
  initiator# iscsiadm modify discovery --iSNS enable
  ```

- 如果已配置静态目标，请启用静态目标搜索方法。

  ```
  initiator# iscsiadm modify discovery --static enable
  ```

7 **如有必要，请重新配置 /dev 名称空间来识别 iSCSI 磁盘。**

```
initiator# devfsadm -i iscsi
```

▼ 如何删除搜索到的 iSCSI 目标

执行以下任一操作之后，将注销关联目标：

- 删除搜索地址
- 删除 iSNS 服务器

- 删除静态配置
- 禁用搜索方法

如果这些关联目标仍在使用（例如，已挂载文件系统），则注销这些设备将失败，设备仍然位于活动目标列表中。

此可选过程假定您已登录到本地系统，在此系统上已配置了对 iSCSI 目标设备的访问权限。

1 成为管理员。

2 可选使用以下操作之一禁用 iSCSI 目标搜索方法：

- 如果要禁用 SendTargets 搜索方法，请使用以下命令：

```
initiator# iscsiadm modify discovery --sendtargets disable
```

- 如果要禁用 iSNS 搜索方法，请使用以下命令：

```
initiator# iscsiadm modify discovery --iSNS disable
```

- 如果要禁用静态目标搜索方法，请使用以下命令：

```
initiator# iscsiadm modify discovery --static disable
```

3 使用以下操作之一删除 iSCSI 设备搜索项：

- 删除 iSCSI SendTargets 搜索项。

  例如：

```
initiator# iscsiadm remove discovery-address 10.0.0.1:3260
```

- 删除 iSCSI iSNS 搜索项。

  例如：

```
# iscsiadm remove isns-server 10.0.0.1:3205
```

- 删除静态 iSCSI 搜索项。

  例如：

```
initiator# iscsiadm remove static-config eui.5000ABCD78945E2B,10.0.0.1
```

注 – 如果尝试禁用或删除正在使用关联逻辑单元 (logical unit, LUN) 的搜索项，则禁用或删除操作将失败，并会显示以下消息：

```
logical unit in use
```

如果出现这种错误，请停止 LUN 中的所有关联 I/O，取消挂载文件系统等。然后，重复执行禁用或删除操作。

---

4 **删除 iSCSI 目标设备。**

例如：

```
target# itadm delete-target target-IQN
```

如果目标仍然处于联机或忙碌状态，此命令可能错误：

```
The target is online or busy. Use the -f (force) option, or 'stmfadm offline-target
iqn.1986-03.com.sun:02:99619b8a-a4dc-4cfb-93f0-ee3debe7b0c8'
itadm delete-target failed with error 16
```

创建 iSCSI 目标门户组

可以创建目标门户组 (target portal group, TPG) 来管理多个 iSCSI 和 iSER 目标的搜索。TPG 是用来确定特定 iSCSI 目标将侦听哪些接口的 IP 地址列表。

TPG 包含 IP 地址和 TCP 端口号。要使用此功能，需要进行以下操作：

- 使用 `itadm create-tpg` 命令创建 TPG 作为 *ip-address:port* 说明符列表。
- 使用 `itadm modify-target -t` 命令将特定 iSCSI 目标绑定到 TPG。
- 当 iSCSI 目标处于活动状态时，为属于 TPG 且与该目标关联的每个 IP 地址和端口创建 iSCSI 侦听器。

TPG 是通过特定端口控制搜索哪些目标的一种有效方式。例如，可以限制 iSCSI 目标，以便它只能通过一个特定的 IP 地址，或只能通过一组具备 iSER 功能的 IP 地址可用。

注 – 请勿将目标门户组与目标组相混淆。目标组是 SCSI 目标端口列表，创建视图时均一视同仁。创建视图可帮助您方便进行 LUN 映射。每个视图项指定目标组、主机组和 LUN。有关目标组和 LUN 映射的更多信息，请参见第 241 页中的 “使 SCSI 逻辑单元可用”和 `stmfadm(1M)`。

要了解有关静态和 iSNS 目标搜索的信息，请参见第 231 页中的 “配置动态或静态目标搜索”。iSCSI 启动器使用 `iscsiadm` 命令来搜索 TPG。有关更多信息，请参见 `iscsiadm(1M)` 和 `itadm(1M)`。

iSER 与 TPG 结合使用

同时使用 SendTargets 搜索和 iSER 时，通用的约定就是使用 TPG 来关联一个特定的 iSCSI 目标端口，该端口只有具备 iSER 功能的 IP 地址。例如，如果目标系统有四个 IP 地址，即 A、B、C 和 D，而且只有地址 B 和 C 具备 iSER 功能，则 B 和 C 可添加到 TPG，并分配到目标 T。

具备以太网和 InfiniBand (IB) 接口的 iSCSI 启动器可以使用 SendTargets 搜索方法来搜索可能的存储目标。未使用 TPG 的情况下，启动器可能总是倾向于使用以太网接口，而不是 IB 接口。通过将目标 T 只与 IB 接口相关联，启动器连接到目标 T 时倾向于正确使用其具备 IB 功能的接口。

▼ 如何为 iSCSI 目标创建目标门户组

通过提供一个独特的名称，可以创建目标门户组 (target portal group, TPG)，并自动生成 TPG 标签（范围为 2-65535）。TPG 标签 1 为缺省 TPG 所保留，当您未在目标上明确设置 TPG 时使用。缺省 TPG 的门户与来自端口 `3260` 上所有网络接口的请求相匹配。

以下步骤说明如何创建两个 TPG，即 TPGA 和 TPGB，其中使用端口 `8000` 作为 TPGB 处 IP 地址。

1 成为管理员。

2 创建两个 TPG。

```
target# itadm create-tpg TPGA 192.168.0.1 192.168.0.2
target# itadm create-tpg TPGB 192.168.0.2:8000 192.168.0.2:8000
```

注 – IPv4 门户是以点分地址表示法指定的（例如`192.168.0.1`）。IPv6 门户地址必须用方括号括起来。

---

3 配置现有 iSCSI 目标以使用 TPG（TPGA 和 TPGB）。

```
# itadm modify-target -t TPGA,TPGB eui.20387ab8943ef7548
```

4 验证所创建的 TPG。

```
# itadm list-tpg -v
```

可以使用 `itadm delete-tpg` 命令删除 TPG。

▼ 如何访问 iSCSI 磁盘

Oracle Solaris iSCSI 启动器搜索到设备后，即会自动进行登录协商。Oracle Solaris iSCSI 驱动程序确定可用的 LUN 数量并创建设备节点。然后，即可将 iSCSI 设备作为其他任何 SCSI 设备对待。

可以在 LUN 上创建 ZFS 存储池，然后创建 ZFS 文件系统。

可以使用 `format` 实用程序查看本地系统上的 iSCSI 磁盘。

1 在 `format` 输出中查看 iSCSI LUN 信息。

```
initiator# format
       0. c0t600144F0B5418B0000004DDAC7C10001d0 <SUN-COMSTAR-1.0 cyl 1022 alt 2 hd 128 sec 32>
          /scsi_vhci/disk@g600144f0b5418b0000004ddac7c10001
```

```
       1. c8t0d0 <Sun-STK RAID INT-V1.0 cyl 17830 alt 2 hd 255 sec 63>
          /pci@0,0/pci10de,375@f/pci108e,286@0/disk@0,0
       2. c8t1d0 <Sun-STK RAID INT-V1.0-136.61GB>
          /pci@0,0/pci10de,375@f/pci108e,286@0/disk@1,0
       3. c8t2d0 <Sun-STK RAID INT-V1.0-136.61GB>
          /pci@0,0/pci10de,375@f/pci108e,286@0/disk@2,0
       4. c8t3d0 <Sun-STK RAID INT-V1.0 cyl 17830 alt 2 hd 255 sec 63>
          /pci@0,0/pci10de,375@f/pci108e,286@0/disk@3,0
Specify disk (enter its number): 0
selecting c0t600144F0B5418B0000004DDAC7C10001d0
[disk formatted]
```

在以上输出中，磁盘 0 是受 MPxIO 控制的 iSCSI LUN。磁盘 1-4 为本地磁盘。

2 可以在 iSCSI LUN 上创建 ZFS 存储池和 ZFS 文件系统。

```
initiator# zpool create pool-name c0t600144F0B5418B0000004DDAC7C10001d0
initiator# zfs create pool-name/fs-name
```

ZFS 文件系统在创建时会自动挂载，并在引导时重新挂载。

使 SCSI 逻辑单元可用

- 第 242 页中的 “如何使逻辑单元可用于所有系统”
- 第 242 页中的 “如何限制 LUN 访问选中的系统”

仅仅使用 STMF 框架注册逻辑单元 (logical unit, LUN) 并不能使其可用于网络上的主机（启动器）。本节介绍如何通过以下配置使 LUN 对启动器主机可见。

对于 iSCSI、光纤通道、和 FCoE 配置，LUN 必须映射后才可访问。可以选择以下方法之一，均使用 `stmfadm` 命令：

- **简单映射**－使用一个命令通过所有端口向所有启动器显示 LUN。使用此方法使 LUN 可用于所有主机。
- **选择性映射**－使您能够指定可以访问 LUN 的主机。使用此方法使 LUN 可用于选中的主机。此过程包括以下步骤：
  1. 定义主机组－主机组是被允许访问相同 LUN 的一组主机（启动器）的名称。如果同一组 LUN 对所有主机可见（如在简单映射中），则不需要此步骤。
  2. 定义目标组－目标组是将同一组 LUN 导出到同一组主机组的一组目标端口的名称。如果同一组 LUN 对所有主机可见（如在简单映射中），则不需要此步骤。
  3. 为每个逻辑单元添加一个或多个视图－添加视图可在 LUN 和主机启动器之间创建网桥。当来自主机组的启动器登录到包含在目标组中的目标端口时，LUN 可见。

注 – 不要将目标组和目标门户组 (target portal group, TPG) 相混淆。TPG 是 iSCSI 目标侦听的 IP 地址列表。TPG 可帮助您限制 iSCSI 目标，以便只能通过一个特定的 IP 地址可用。有关目标组的更多信息，请参见 `stmfadm(1M)`。

视图项由四部分组成：主机组、目标组、逻辑单元 (logical unit, LUN) 和 LUN 标识符。这四个部分只有 LUN 标识符是必需的。如果省略其他部分，则假设以下缺省值：

- 如果省略主机组，则假设所有启动器值。
- 如果省略目标组，则假设所有目标值。
- 如果省略 LUN，则系统为该项选择一个适当的 LUN。

▼ 如何使逻辑单元可用于所有系统

此过程使 LUN 可用于存储网络上的所有启动器主机。

1 获取 LUN 的全球唯一标识 (Global Unique Identification, GUID) 编号。

```
# stmfadm list-lu -v
```

2 为逻辑单元添加视图。

```
3 # stmfadm add-view GUID-number
```

▼ 如何限制 LUN 访问选中的系统

使用此过程限制 LUN 访问存储网络上选中的主机。如果正在使用光纤通道端口，首先要确定全局名称 (World Wide Names, WWN)。然后，选择性地将一个逻辑单元编号 (logical unit number, LUN) 映射到端口，例如 `host-a` 上的端口。目标组 (`targets-0`) 也被定义为一组给定的目标端口，将相同组的 LUN 导出到相同的主机组。

有关配置目标组的信息，请参见 `stmfadm(1M)`。

1 成为管理员。

2 在启动器上确定光纤通道 (Fibre Channel, FC) 端口。

```
initiator# fcinfo hba-port
HBA Port WWN: 210000e08b195dae
        Port Mode: Initiator
        Port ID: 0
        OS Device Name: /dev/cfg/c8
        Manufacturer: QLogic Corp.
        Model: 375-3108-xx
        Firmware Version: 03.03.28
        FCode/BIOS Version:  fcode: 1.13;
        Serial Number: not available
```

```
        Driver Name: qlc
        Driver Version: 20100408-3.01
        Type: unknown
        State: offline
        Supported Speeds: 1Gb 2Gb
        Current Speed: not established
        Node WWN: 200000e08b195dae
        NPIV Not Supported
.
.
.
```

3 **创建主机组。**

```
target# stmfadm create-hg host-a 210000e08b195dae 210100e08b395dae
```

4 **将前面的输出中确定的 WWN 作为主机组的成员添加。**

```
target# stmfadm add-hg-member -g host-a
```

5 **创建目标组。**

```
target# stmfadm create-tg targets-0
```

6 **通过添加目标名称指定目标组成员。**

每个 SCSI 目标只可成为一个目标组的成员。

```
target# stmfadm add-tg-member -g targets-0 wwn.guid-number
```

7 **为 LUN 确定 GUID 编号。**

```
target# stmfadm list-lu -v
```

8 **通过添加一个视图项，指定主机组名称和 LUN GUID 编号使 LUN 可用。**

```
target# stmfadm add-view -h host-a -t targets-0 -n 1 guid-number
```

使用 COMSTAR 配置光纤通道设备

可以在 FC 网络环境中的 SPARC 系统或 x86 系统上设置并配置一个 COMSTAR 目标，并使其可访问存储网络。确保已完成以下先决步骤：

- 第 233 页中的 “如何创建 iSCSI LUN”

为 COMSTAR 配置光纤通道端口

光纤通道 (Fibre Channel, FC) 端口提供程序可以使用用于 FC 启动器的相同 HBA。一个给定的 FC 端口可以作为启动器或目标使用，但不能同时作为两者使用。还可以配置一个双端口或四端口 FC HBA，以便 HBA 端口的子集处于目标模式，其余的端口处于启动器模式。

本节中的过程特定于 QLogic HBA。在启动器模式中结合 HBA 使用的驱动程序为 qlc，即 Qlogic 驱动程序。qlc 驱动程序只在启动器模式下运行，不能用于目标模式。用于目标 QLogic 4G HBA 的 COMSTAR 驱动程序为 qlt。

由于启动器模式和目标模式使用不同的驱动程序，附加到 HBA 端口的驱动程序将其功能定义为目标或启动器。通过指定 HBA 的 PCI 设备 ID 可以指定所有端口的驱动程序。或者，可以在端口对端口的基础上配置驱动程序。两种方法都使用 update_drv 命令并在本节中进行了介绍。有关更多信息，请参见 update_drv(1M)。

▼ 如何显示现有 FC 端口绑定

对 HBA 端口做出更改之前，首先要检查现有端口绑定。

1. **成为管理员。**

2. **显示当前绑定到端口驱动程序的情况。**

   在此示例中，当前的绑定是 pciex1077,2432。

```
# mdb -k
Loading modules: [ unix krtld genunix specfs ...
> ::devbindings -q qlc
30001617a08 pciex1077,2432, instance #0 (driver name: qlc)
300016177e0 pciex1077,2432, instance #1 (driver name: qlc)
> $q
```

▼ 如何将所有 FC 端口设置到特定模式

此过程使用特定的 PCI 设备 ID 将所有 HBA 上的所有端口更改到目标模式。PCI 设备 ID 号将驱动程序绑定到端口，从而使用该 PCI 设备 ID 设置所有 HBA 端口，例如，所有 QLogic 4G PCI 将 HBA 表达到目标模式。

1. **成为管理员。**

2. **删除当前绑定。**

   在此示例中，qlc 驱动程序主动绑定到 pciex1077,2432。必须删除 qlc 现有的绑定，然后才可以将该绑定添加到新的驱动程序。此语法中需要单引号。

```
# update_drv -d -i 'pciex1077,2432' qlc
Cannot unload module: qlc
Will be unloaded upon reboot.
```

   此消息**并未**指示错误。配置文件已更新，但 qlc 驱动程序仍然绑定到端口，直到系统重新引导。

3. **建立新的绑定。**

   在此示例中，qlt 已更新。此语法中需要单引号。

```
# update_drv -a -i 'pciex1077,2432' qlt
devfsadm: driver failed to attach: qlt
Warning: Driver (qlt) successfully added to system but failed to attach
```

此消息并未指示错误。qlc 驱动程序仍然绑定到端口，直到系统重新引导。qlt 驱动程序在系统重新引导时连接。

4 重新引导系统以附加新的驱动程序。然后重新检查绑定。

```
# init 6
.
.
.
# mdb -k

Loading modules: [ unix krtld genunix specfs dtrace ...
> ::devbindings -q qlt
30001615a08   pciex1077,2432, instance #0 (driver name: qlt)
30001615e0    pciex1077,2432, instance #1 (driver name: qlt)
> $q
```

5 验证目标模式框架是否可以访问 HBA 端口。

```
# stmfadm list-target -v
Target: wwn.210100E08BA54E60
Operational Status : Offline
Provider Name : qlt(1)
Alias : -
Sessions : 0
Target: wwn.210100E08BA54E60
Operational Status : Offline
Provider Name : qlt(0)
Alias : -
Sessions : 0
```

▼ 如何将选中的 FC 端口设置到启动器或目标模式

此过程使用**基于路径的绑定**。这说明如何使用特定的设备路径将端口绑定到驱动程序，该驱动程序与当前绑定的驱动程序不同。

1 成为管理员。

2 显示 HBA 端口及其各自的设备路径列表。

该示例显示了具有两个端口的单个 HBA 的设备路径。

```
# luxadm -e port
/devices/pci@780/QLGC,qlc@0,1/fp@0,0:devctl CONNECTED
/devices/pci@780/QLGC,qlc@0/fp@0,0:devctl CONNECTED
```

3 将顶部端口设置为目标模式，并使底部端口保持启动器模式。

删除路径的初始 /devices 部分，且包括一切直至 /fp@0...。/devices 部分删除的路径是系统绑定 qlt 驱动程序的路径。

此语法中需要单引号。

```
# update_drv -a -i '/pci@780/QLGC,qlc@0,1' qlt
devfsadm: driver failed to attach: qlt
Warning: Driver (qlt) successfully added to system but failed to attach.
```

此消息并未指示错误。**qlc** 驱动程序仍然绑定到端口，直到重新引导。**qlt** 驱动程序在重新引导时连接。

4 **重新引导系统以附加新的驱动程序。然后重新检查绑定。**

应该看到端口从启动器模式 (qlc) 更改到目标模式 (qlt)。

```
# init 6
.
.
.
# mdb -k
.
.
.
> $q
```

5 **验证目标模式框架是否可以访问 HBA 端口。**

```
# stmfadm list-target -v
```

使逻辑单元可用于 FC 和 FCoE

仅仅使用 STMF 框架注册逻辑单元 (logical unit, LUN) 并不能使其可用于网络上的主机（启动器）。必须通过映射逻辑单元使其对光纤通道和 FCoE 配置的启动器主机可见。要确定使用哪种方法以及如何映射逻辑单元，请参见第 242 页中的 “如何使逻辑单元可用于所有系统”。两种方法均使用 stmfadm 命令。以下附加步骤是关于 FC 和 FCoE 配置的步骤。

▼ 如何使逻辑单元可用于 FC 和 FCoE

此过程使 LUN 可用于存储网络上 FC 或 FCoE 配置的所有主机和选定的主机。这些步骤在主机上运行。

1 **成为管理员。**

2 **使 LUN 可用于主机。**

获取 LUN 的全球唯一标识 (Global Unique Identification, GUID) 编号。

```
# sbdadm list-lu
# stmfdm list-lu -v
```

确定主机 FC 或 FCoE 端口的 WWN。

```
# fcinfo hba-port
HBA Port WWN: *210000e08b83378d*
OS Device Name: /dev/cfg/c4
Manufacturer: Qlogic Corp.
Model: QLA2462
Firmware Version: 4.0.27
Fcode/BIOS Version: N/A
```

```
Type: N-port
State: online
Supported Speeds: 1Gb 2Gb 4Gb
Current Speed: 4Gb
Node WWN: 210000e08b83378d
HBA Port WWN: *210100e08ba3378d*
OS Device Name: /dev/cfg/c5
Manufacturer: Qlogic Corp.
Model: QLA2462
Firmware Version: 4.0.27
Fcode/BIOS Version: N/A
Type: N-port
State: online
Supported Speeds: 1Gb 2Gb 4Gb
Current Speed: 4Gb
Node WWN: 210100e08ba3378d
```

3 **添加视图并执行映射。**

按照第 242 页中的 “如何使逻辑单元可用于所有系统”中的说明。

4 **通过运行以下脚本，验证 LUN 在 Oracle Solaris 启动器主机上是否可见。**

```
#!/bin/ksh
fcinfo hba-port |grep "^HBA" | awk '{print $4}' | while read 1n
do
        fcinfo remote-port -p $1n -s >/dev/null 2>&amp;1
done
```

所有后续 LUN 出现在 `format` 输出中，因为脚本强制启动器通过所有端口触摸所有 LUN。如果未看到 LUN，请再次运行 `format` 命令。如果仍然未看到 LUN，使用 `svcs stmf` 命令确保服务已在目标上启用。还要确保已为 LUN 添加视图项，如第 242 页中的 “如何使逻辑单元可用于所有系统”中所述。

5 **验证 LUN 在其他系统上是否可见。**

- 对于 Linux 启动器主机，通过运行 HBA 供应商提供的实用程序验证 LUN 是否可见。实用程序扫描配置更改。
- 对于 Windows 启动器主机，通过选择 "Control Panel"（控制面板）→ "Administrative Tools"（管理工具）→ "Computer Management"（计算机管理）→ "Disk Management"（磁盘管理）验证逻辑单元是否可见。然后，从 "Action"（操作）菜单中选择 "Rescan Disks"（重新扫描磁盘）。

使用 COMSTAR 配置 FCoE 设备

可以在以太网光纤通道 (Fibre Channel over Ethernet, FCoE) 网络环境中设置并配置一个 COMSTAR 目标，然后使其可访问存储网络。在开始前确保系统满足必要的先决条件：

- 第 233 页中的 “如何创建 iSCSI LUN”
- 第 235 页中的 “如何创建 iSCSI 目标”

- 第 236 页中的 “如何配置 iSCSI 启动器”

配置 FCoE 端口

- 第 248 页中的 “启用以太网接口上的 802.3x PAUSE 和巨型帧 (Jumbo Frame)”
- 第 248 页中的 “如何创建 FCoE 目标端口”
- 第 249 页中的 “如何验证 FCoE 目标端口正在运行 ”
- 第 250 页中的 “如何删除 FCoE 目标端口”

FCoE 功能是通过以太网接口提供的。以太网光纤通道 (Fibre Channel over Ethernet, FCoE) 端口是与以太网接口相关联的逻辑项。在 Oracle Solaris 系统内，存在 FCoE 端口和 Ethernet 接口的一对一映射。只能将一个 FCoE 端口与给定的以太网接口相关联。FCoE 和 IP 不能共享同一个以太网接口。所以在以太网接口上创建 FCoE 端口之前，请确保接口是非活动的。

FCoE 端口配置在重新引导过程中具有持久性。系统重新引导之后，所有已配置的 FCoE 端口将自动在线创建和放置。

对于 FCoE 目标端口，在创建 FCoE 目标端口获得持久性行为之前，必须启用以下服务。

```
# svcadm enable svc:/system/fcoe_target:default
```

启用以太网接口上的 802.3x PAUSE 和巨型帧 (Jumbo Frame)

以太网硬件和驱动程序不同，这些设置也会有所不同。大多数情况下，您必须修改以太网接口的 `driver.conf` 文件，然后重新引导。有关如何启用这些功能的详细信息，请参见以太网接口的 `driver.conf` 文件。

在以太网接口上创建 FCoE 端口之前，请执行以下先决步骤之一：

- 在以太网接口上启用 802.3x（也称为 PAUSE）。
  这样做可确保一致的以太网传输。
- 启用以太网接口上的巨型帧（大于 2.5 KB）。
  光纤通道数据帧可达到 2136 字节。

▼ 如何创建 FCoE 目标端口

1 在指定的网络接口上创建 FCoE 目标端口。

```
# fcadm create-fcoe-port -t nxge0
```

如果所选的以太网接口不支持多个单播地址（例如 VMware 网络接口），系统会提示您明确启用该接口上的混杂模式。

2 启用混杂模式（若有提示）。

```
# fcadm create-fcoe-port -t -f e1000g0
```

如果没有出现错误消息，则创建 FCoE 目标端口，而且 FCoE 目标在线。

▼ 如何验证 FCoE 目标端口正在运行

1 显示所创建的 FCoE 端口。

```
# fcadm list-fcoe-ports
HBA Port WWN: 200000144fda7f66
    Port Type: Target
    MAC Name: nxge0
    MTU Size: 9194
    MAC Factory Address: 00144fda7f66
    MAC Current Address: 0efc009a002a
    Promiscuous Mode: On
```

2 显示主机上的所有目标模式光纤通道 HBA 端口。

```
# fcinfo hba-port -t
HBA Port WWN: 200000144fda7f66
    Port Mode: Target
    Port ID: 9a002a
    OS Device Name: Not Applicable
    Manufacturer: Sun Microsystems, Inc.
    Model: FCoE Virtual FC HBA
    Firmware Version: N/A
    FCode/BIOS Version: N/A
    Serial Number: N/A
    Driver Name: COMSTAR FCOET
    Driver Version: 1.0
    Type: F-port
    State: online
    Supported Speeds: 1Gb 10 Gb
    Current Speed: 10Gb
    Node WWN: 100000144fda7f66
```

3 查看 FCoE 目标端口列表。

-v 选项显示有关目标的更多信息，以及已登录启动器的 SCSI 会话信息。

```
# stmfadm list-target -v
Target wwn.200000144FDA7F66
     Operational Status: Online
     Provider Name     : fcoet
     Alias             : fcoet1
     Sessions          : 1
          Initiator: wwn.210000E08B818343
              Alias: #QLA2342 FW:v3.03.25 DVR:v8.02.14.01
              Logged in since: ...
```

▼ 如何删除 FCoE 目标端口

需要时可以禁用 FCoE 功能。

1 **使 FCoE 目标端口脱机。**

```
# stmfadm offline-target wwn.200000144fda7f66
```

2 **删除 FCoE 目标端口。**

```
# fcadm delete-fcoe-port nxge0
```

使用 COMSTAR 配置 SRP 设备

SCSI RDMA 协议 (SCSI RDMA Protocol, SRP) 通过将 SCSI 数据传输阶段映射到 Infiniband (IB) 远程直接内存访问 (Remote Direct Memory Access, RDMA) 操作加速了 SCSI 协议。因此，SRP 启动器可以在 CPU 利用率相对较低时以高数据率读写来自 COMSTAR SRP 目标的数据。

可以设置并配置 COMSTAR SRP 目标，并使其可用于 Infiniband (IB) 光纤网络。不论支持的 IB 主机通道适配器 (Host Channel Adapter, HCA) 安装在目标系统的哪个位置，SRP 目标都可用。

- **每个 IB HCA 一个 SCSI 目标**－COMSTAR SRP 目标使用简易模型，其中每个支持的 IB HCA 注册为 SCSI 目标。SCSI 目标是包含任务路由器的虚拟对象，它在 SCSI 传输（在该情况中为 SRP）和 SCSI 后端（STMF 和 SBD）之间起着连接作用。

  一个 HCA 可包含多个物理端口。一个 HCA 的所有端口之间共享相同的 SCSI 目标。代表 HCA 的 SCSI 目标自动可用于通过该 HCA 所有活动端口传入的连接。
- **SRP 目标 eui 标识符**－在 IB 结构中，每个 HCA 和每个端口由生产商分配一个 64 位的 GUID。为每个 HCA 创建的 COMSTAR SCSI 目标将被赋予与该 HCA 的 GUID 相对应的名称，其格式为：`eui.`*HCA-GUID*。例如，如果目标系统包含支持的 IB HCA 并具有 HCA GUID `0003BA0001002E48`，则将使用名称 `eui.0003BA0001002E48` 创建 SCSI 目标。字符串 `eui` 代表**扩展唯一标识**，并命名 SCSI 和 IB 标准中均使用的一类 GUID。
- **SRP 启动器 eui 标识符**－同样，SRP 使用一个 64 位的启动器 GUID 来识别启动器系统。选择使用哪个 GUID 由 SRP 启动器执行确定。许多启动器使用用于传出连接的 HCA 的 GUID。例如，使用 GUID `0003BA0001002EA5` 的启动器被 OMSTA 认为是 `eui.0003BA0001002EA5`。

结合 SRP 使用 COMSTAR 视图

COMSTAR 查看工具可用于创建目标组和主机组，其限制和配置哪个存储逻辑单元 (logical unit, LUN) 可通过每个 SCSI 目标或启动器受到访问，如第 241 页中的 “使 SCSI 逻

辑单元可用”中所述。SRP 启动器的 eui 标识符被添加到主机组。SRP SCSI 目标的 eui 标识符被添加到目标组。然后每个 LUN 的视图项确定每个启动器可访问的特定 LUN 组。

▼ 如何启用 SRP 目标服务

COMSTAR SRP 目标的 COMSTAR 端口提供程序由服务管理工具 (Service Management Facility, SMF) 管理。主要 SRP 目标设备为 `svc:/system/ibsrp/target:default`，可缩写为 `ibsrp/target`。

SRP 软件包为 `storage/scsi-rdma/scsi-rdma-target`。

1. **成为管理员。**
2. **递归启用 SRP 目标服务。**

   ```
   # svcadm enable -r ibsrp/target
   ```

3. **显示 SRP 目标服务信息。**

   ```
   # svcs -l ibsrp/target
   ```

▼ 如何验证 SRP 目标状态

1. **成为管理员。**
2. **验证系统上预期 SRP SCSI 目标的存在。**

   ```
   # srptadm list-target
   Target HCA 21280001A0D0F0:
       Enabled             : true
       SRP Target Name     : eui.0021280001A0D0F0
       Operational Status  : online
   ```

在基于 iSCSI 的存储网络中配置验证

为 iSCSI 设备设置验证是可选的。

在安全环境下，由于仅有受信任的启动器才能访问目标，因此不需要进行验证。

在安全性较低的环境中，目标不能确定连接请求是否真正来自给定主机。在这种情况下，目标可以使用质询握手身份验证协议 (Challenge-Handshake Authentication Protocol, CHAP) 来验证启动器。

CHAP 验证使用质询和响应的概念，这意味着目标将质询启动器以查验其身份。要使质询/响应方法可行，目标必须知道启动器的密钥，并且必须将启动器设置为响应质询。有关在数组中设置密钥的说明，请参阅数组供应商的文档。

iSCSI 支持单向和双向验证，如下：

- **单向验证**使目标可以验证启动器的身份。单向验证代表目标验证启动器。
- **双向验证**通过使启动器可以验证目标身份，提供了更高级别的安全性。双向验证由启动器驱动，控制是否执行了双向验证。该目标唯一所需的设置是必须正确定义 chap 用户和 chap 机密。

▼ 如何配置对 iSCSI 启动器的 CHAP 验证

此过程假定您已登录到本地系统，并且要在此系统中安全地访问已配置的 iSCSI 目标设备。

- COMSTAR iSCSI 目标的 CHAP 密钥长度必须最少为 12 个字符，最多为 255 个字符。有些启动器只支持少于最大长度的密钥。
- 使用 CHAP 识别自己的每个节点必须具有用户名和口令。在 Oracle Solaris 11 环境中，缺省情况下，CHAP 用户名会被设置为启动器或目标节点名称（即 `iqn` 名称）。可以将 CHAP 用户名设置为小于 512 个字节的任意长度的文本。512 个字节长度限制是 Oracle Solaris 11 限制。但是，如果未设置 CHAP 用户名，则系统会在初始化时将其设置为节点名称。

可以使用第三方 RADIUS 服务器来简化 CHAP 密钥管理，RADIUS 服务器充当集中验证服务。使用 RADIUS 时，RADIUS 服务器会存储节点名称组并匹配 CHAP 密钥。执行验证的系统将请求程序的节点名称和提供的机密转发给 RADIUS 服务器。RADIUS 服务器确认密钥是否是验证给定节点名称的合适密钥。iSCSI 和 iSER 均支持使用 RADIUS 服务器。

有关使用第三方 RADIUS 服务器的更多信息，请参见第 254 页中的 “使用第三方 RADIUS 服务器简化 iSCSI 配置中的 CHAP 管理”。

1. **成为管理员。**

   有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。

2. **确定要配置单向 CHAP 还是双向 CHAP。**

   - 单向验证可使目标验证启动器。此方法为缺省方法。只需完成步骤 3–5 即可。
   - 双向验证通过使启动器验证目标提供了更高级别的安全性。请完成步骤 3–9。

3. **单向 CHAP：设置启动器的密钥。**

   以下命令将启动一个用于定义 CHAP 密钥的对话框：

   ```
   initiator# iscsiadm modify initiator-node --CHAP-secret
   Enter CHAP secret: ************
   Re-enter secret: ************
   ```

4. **（可选的）单向 CHAP：设置启动器的 CHAP 用户名。**

   缺省情况下，启动器的 CHAP 用户名会设置为启动器节点名称。

使用以下命令来使用启动器的 CHAP 用户名：

```
initiator# iscsiadm modify initiator-node --CHAP-name new-CHAP-name
```

5 **单向 CHAP－在启动器上启用 CHAP 验证。**

```
initiator# iscsiadm modify initiator-node --authentication CHAP
```

CHAP 要求启动器节点具有用户名和口令。用户名通常由目标用于查找给定用户名的密钥。

6 **选择以下步骤之一启用或禁用双向 CHAP。**

- 启用双向 CHAP 以与目标连接。

  ```
  initiator# iscsiadm modify target-param -B enable target-iqn
  ```

- 禁用双向 CHAP。

  ```
  initiator# iscsiadm modify target-param -B disable target-iqn
  ```

7 **双向 CHAP：针对目标将验证方法设置为 CHAP。**

```
initiator# iscsiadm modify target-param --authentication CHAP target-iqn
```

8 **双向 CHAP：设置确定目标的目标设备密钥。**

以下命令将启动一个用于定义 CHAP 密钥的对话框：

```
initiator# iscsiadm modify target-param --CHAP-secret target-iqn
```

9 **双向 CHAP：如果目标使用备用 CHAP 用户名，请设置确定目标的 CHAP 名称。**

缺省情况下，目标的 CHAP 名称会设置为目标名称。

可以使用以下命令来更改目标的 CHAP 名称：

```
initiator# iscsiadm modify target-param --CHAP-name target-CHAP-name
```

▼ 如何配置对 iSCSI 目标的 CHAP 验证

此过程假定您已登录到包含 iSCSI 目标的本地系统。

1 **成为管理员。**

2 **确定要配置单向 CHAP 还是双向 CHAP。**

- 单向验证为缺省方法。只需完成步骤 3–5 即可。
- 对于双向验证。请完成步骤 3–7。

3 **单向/双向 CHAP：配置目标来要求启动器确定正在使用 CHAP。**

```
target# itadm modify-target -a chap target-iqn
```

4. **单向/双向 CHAP：创建描述启动器的启动器上下文。**

   使用启动器的完整节点名称和启动器的 CHAP 密钥创建启动器上下文。

   ```
   target# itadm create-initiator -s initiator-iqn
   Enter CHAP secret: ************
   Re-enter secret: ************
   ```

5. **单向/双向 CHAP：如果启动器使用备用 CHAP 名称，则使用备用名称配置启动器上下文。**

   ```
   target# itadm modify-initiator -u initiator-CHAP-name initiator-iqn
   ```

6. **双向 CHAP：设置确定目标的目标设备密钥。**

   ```
   target# itadm modify-target -s target-iqn
   Enter CHAP secret: ************
   Re-enter secret: ************
   ```

7. **（可选的）双向 CHAP：如果目标使用备用 CHAP 用户名而不是目标节点名称 (iqn)，请修改目标。**

   ```
   target# itadm modify-target -u target-CHAP-name target-iqn
   ```

使用第三方 RADIUS 服务器简化 iSCSI 配置中的 CHAP 管理

可以使用第三方 RADIUS 服务器来简化 CHAP 密钥管理。RADIUS 服务器充当集中验证服务。使用此方法时，推荐的做法是使用每个启动器节点的缺省 CHAP 名称。通常情况下，所有启动器使用缺省 CHAP 名称时，不必在目标上创建启动器上下文。

▼ 如何为 iSCSI 目标配置 RADIUS 服务器

此过程假定您已登录到本地系统，并且要在此系统中安全地访问已配置的 iSCSI 目标设备。

1. **成为管理员。**

2. **使用 RADIUS 服务器的 IP 地址和端口配置启动器节点。**

   缺省端口为 1812。一次在目标系统上针对所有 iSCSI 目标完成该配置。

   ```
   initiator# itadm modify-defaults -r RADIUS-server-IP-address
   Enter RADIUS secret: ************
   Re-enter secret: ************
   ```

3. **配置用于目标系统和 RADIUS 服务器之间通信的共享密钥。**

   ```
   initiator# itadm modify-defaults -d
   Enter RADIUS secret: ************
   Re-enter secret: ************
   ```

4 **配置目标系统来要求 RADIUS 验证。**

可以为单个目标执行该配置，或作为所有目标的缺省配置。

```
initiator# itadm modify-target -a radius target-iqn
```

5 **使用以下组件配置 RADIUS 服务器：**

- 目标节点的标识（例如其 IP 地址）
- 目标节点用来与 RADIUS 服务器通信的共享密钥
- 每个启动器需要验证的启动器 CHAP 名称（例如其 `iqn` 名称）和密钥

▼ 如何为 iSCSI 启动器配置 RADIUS 服务器

可以使用第三方 RADIUS 服务器来简化 CHAP 密钥管理。RADIUS 服务器充当集中验证服务。此步骤只在启动器要求双向 CHAP 验证时才有用。尽管仍然必须指定启动器的 CHAP 密钥，但是，在将双向验证与 RADIUS 服务器结合使用时，不再需要在启动器上指定每个目标的 CHAP 密钥。RADIUS 可在启动器或目标上独立配置。启动器和目标不必使用 RADIUS。

1 **成为管理员。**

2 **使用 RADIUS 服务器的 IP 地址和端口配置启动器节点。**

缺省端口为 1812。

```
# iscsiadm modify initiator-node --radius-server ip-address:1812
```

3 **使用 RADIUS 服务器的共享密钥配置启动器节点。**

iSCSI 与 RADIUS 服务器进行交互之前必须使用共享密钥配置 RADIUS 服务器。

```
# iscsiadm modify initiator-node --radius-shared-secret
Enter secret:
Re-enter secret
```

4 **启用 RADIUS 服务器。**

```
# iscsiadm modify initiator-node --radius-access enable
```

5 **设置 CHAP 双向验证的其他方面。**

```
# iscsiadm modify initiator-node --authentication CHAP
# iscsiadm modify target-param --bi-directional-authentication enable target-iqn
# iscsiadm modify target-param --authentication CHAP target-iqn
```

6 **使用以下组件配置 RADIUS 服务器：**

- 此节点的标识（例如其 IP 地址）
- 此节点用来与 RADIUS 服务器通信的共享密钥
- 每个需要验证的目标的 CHAP 名称（例如其 `iqn` 名称）和密钥

Oracle Solaris iSCSI 和 RADIUS 服务器错误消息

本节介绍与 Oracle Solaris iSCSI 和 RADIUS 服务器配置相关的错误消息。还介绍了可能的恢复解决方法。

空 RADIUS 共享密钥

原因: 已在启动器上启用了 RADIUS 服务器，但未设置 RADIUS 共享密钥。

解决方法: 使用 RADIUS 共享密钥配置启动器。有关更多信息，请参见第 254 页中的 “如何为 iSCSI 目标配置 RADIUS 服务器”。

警告：RADIUS 数据包验证失败

原因: 启动器无法验证 RADIUS 数据包。如果在启动器节点上配置的共享密钥不同于 RADIUS 服务器上的共享密钥，则会出现此错误。

解决方法: 使用正确的 RADIUS 共享密钥重新配置启动器。有关更多信息，请参见第 254 页中的 “如何为 iSCSI 目标配置 RADIUS 服务器”。

设置 Oracle Solaris iSCSI 多路径设备

多路径 I/O (MPxIO) 使得 I/O 设备可通过 I/O 设备单个实例中的多个主机控制器接口接受访问。

使用 Oracle Solaris iSCSI 多路径 (MPxIO) 设备时考虑以下原则：

- **Oracle Solaris iSCSI 和 MPxIO** – MPxIO 支持 Oracle Solaris iSCSI 配置中目标端口聚合和可用性，该配置在 iSCSI 启动器上为每个目标 (MS/T) 配置多个会话。
  - 为两个或更多 NIC 聚合和故障转移使用 IP 网络多路径 (IPMP)。
  - iSCSI 主机的基本配置是一个带有两个专用于 iSCSI 流量的 NIC 的服务器。使用 IPMP 配置 NIC。为非 iSCSI 流量提供附加的 NIC 以优化性能。
  - 只能通过使用 Oracle Solaris iSCSI MS/T 功能实现活动的多路径，以及 IPMP 配置的故障转移和冗余。
    - 如果一个 NIC 在 IPMP 配置中出现故障，则 IPMP 处理故障转移。MPxIO 驱动程序不检测故障。在非 IPMP 配置中，MPxIO 驱动程序出现故障并脱机。
    - 如果一个目标端口在 IPMP 配置中出现故障，则 MPxIO 驱动程序检测故障并提供故障转移。在非 IPMP 配置中，MPxIO 驱动程序检测故障并提供故障转移。

    有关为每个目标配置多个会话的更多信息，请参见第 257 页中的 “如何为目标启用多个 iSCSI 会话”。有关配置 IPMP 的信息，请参见《Oracle Solaris 管理：网络接口和网络虚拟化》中的第 15 章 “管理 IPMP”。
- **Oracle Solaris iSCSI、光纤通道 (Fibre Channel, FC) 和 MPxIO** – MPxIO 驱动程序在更复杂的 iSCSI/FC 配置中提供以下行为：

- 如果在 FC SAN 中具有到 FC 网桥的双 iSCSI，iSCSI 显示到 MPxIO 的目标路径。MPxIO 与每个 LUN 标识符的独特 SCSI 相匹配，如果它们相同则显示到 iSCSI 驱动程序的一个路径。
- 如果具有使用 iSCSI 和 FC 连接目标的配置，MPxIO 驱动程序可提供到相同设备的不同传输。在此配置中，MPxIO 使用这两种路径。
- 如果与 MPxIO 一同使用 iSCSI 和 FC，确保在 `/etc/driver/drv/fp.conf` 和 `/driver/drv/iscsi.conf` 文件中的 MPxIO 参数与要支持的 MPxIO 配置相匹配。例如，在 `fp.conf` 中，可以确定 MPxIO 是否在 HBA 或每个端口的基础上全局启用。

- **第三方硬件考虑因素**－找出第三方 HBA 是否有资格与 Oracle Solaris iSCSI 和 MPxIO 协同工作。

  如果正在使用第三方 HBA，可能需要询问供应商有关 `/driver/drv/scsi_vhci.conf` 文件的对称选项信息。

▼ 如何为目标启用多个 iSCSI 会话

可以使用此过程创建与单个目标连接的多个 iSCSI 会话。此方案对于支持登录重定向或在同一个目标门户组中具有多个目标门户的 iSCSI 目标设备非常有用。应该将每个目标的多个 iSCSI 会话与 Oracle Solaris 的 SCSI 多路径 (MPxIO) 功能结合使用。如果在主机端使用多个 NIC 连接到同一目标的多个门户上，可以实现更高的带宽。

MS/T 功能通过改变启动器的会话 ID (ISID) 在目标上创建两个或更多会话。启用此功能在网络上创建两个 SCSI 层路径，以便使多个目标通过 iSCSI 层暴露于 Oracle Solaris I/O 层。MPxIO 驱动程序处理这些路径上的预留空间。

有关 iSCSI 如何与 MPxIO 路径交互的更多信息，请参见第 256 页中的 “设置 Oracle Solaris iSCSI 多路径设备”。

为 iSCSI 目标配置多个会话之前，请查看以下各项：

- 典型的 MS/T 配置具有两个或更多已配置的会话。

  不过，如果存储支持多个 TPGT 而且正在主机系统上使用 SendTarget 搜索方法，则已配置的会话数可设置为 1。SendTarget 搜索自动检测是否存在多个路径，并创建多个目标会话。
- 确认 `mxpio` 配置参数已在 `/etc/driver/drv/iscsi.conf` 文件中启用。

  ```
  # cd /etc/driver/drv
  # grep mpxio iscsi.conf iscsi.conf
  iscsi.conf:mpxio-disable="no";
  ```
- 确认已使用 IPMP 配置多个网络连接。
- 确认多个网络连接可用。

  ```
  # ipadm show-addr
  ```

1. **成为管理员。**

2 列出 iSCSI 启动器和目标的当前参数值。

a. 列出 iSCSI 启动器的当前参数值。

```
initiator# iscsiadm list initiator-node
Initiator node name: iqn.1986-03.com.sun:01:0003ba4d233b.425c293c
Initiator node alias: zzr1200
        .
        .
        .
        Configured Sessions: 1
```

b. 列出 iSCSI 目标设备的当前参数值。

```
initiator# iscsiadm list target-param -v iqn.1992-08.com.abcstorage:sn.84186266
Target: iqn.1992-08.com.abcstorage:sn.84186266
        Alias: -
        .
        .
        .
        Configured Sessions: 1
```

已配置的会话值是将为目标门户组中的每个目标名称创建的已配置 iSCSI 会话的数目。

3 选择以下命令之一，修改启动器节点处已配置会话的数目，以便应用于所有目标；或修改某个目标级别的已配置会话的数目，以便应用于特定目标：

目标的会话数目必须介于 1 到 4 之间。

- 将所需的参数值应用于 iSCSI 启动器节点。

```
initiator# iscsiadm modify initiator-node -c 2
```

- 将所需的参数值应用于 iSCSI 目标。

```
initiator# iscsiadm modify target-param -c 2  iqn.1992-08.com.abcstorage:sn.84186266
```

- 将已配置的会话绑定到一个或多个本地 IP 地址。

  还可以将已配置的会话绑定到一个本地 IP 地址。使用此方法时，在以逗号分隔的列表中将提供一个或多个本地 IP 地址。每个 IP 地址都表示一个 iSCSI 会话。此方法也可以在 `initiator-node` 或 `target-param` 级别使用。例如：

```
initiator# iscsiadm modify initiator-node -c 10.0.0.1,10.0.0.2
```

注 – 如果指定的 IP 地址无法路由，则将忽略该地址，并对此会话使用缺省的 Oracle Solaris 路由和 IP 地址。

---

4 验证是否已修改参数。

a. 显示更新后的启动器节点信息。

```
initiator# iscsiadm list initiator-node
Initiator node name: iqn.1986-03.com.sun:01:0003ba4d233b.425c293c
Initiator node alias: zzr1200
```

```
.
.
.
Configured Sessions: 2
```

b. 显示更新后的目标节点信息。

```
initiator# iscsiadm list target-param -v iqn.1992-08.com.abcstorage:sn.84186266
Target: iqn.1992-08.com.abcstorage:sn.84186266
        Alias: -
        .
        .
        .
        Configured Sessions: 2
```

5 列出多个路径以确认 OS 设备名称与 `iscsiadm list` 输出相匹配，并且路径计数为 2 或更多。

```
initiator# mpathadm list lu
```

监视 iSCSI 配置

通过使用 `iscsiadm list` 命令，可以显示有关 iSCSI 启动器和目标设备的信息。

▼ 如何显示 iSCSI 配置信息

1 成为管理员。

2 显示有关 iSCSI 启动器的信息。

例如：

```
# iscsiadm list initiator-node
Initiator node name: iqn.1986-03.com.sun:01:0003ba4d233b.425c293c
Initiator node alias: zzr1200
        Login Parameters (Default/Configured):
                Header Digest: NONE/-
                Data Digest: NONE/-
        Authentication Type: NONE
        RADIUS Server: NONE
        RADIUS access: unknown
        Configured Sessions: 1
```

3 显示有关当前使用的搜索方法的信息。

例如：

```
# iscsiadm list discovery
Discovery:
    Static: enabled
    Send Targets: enabled
    iSNS: enabled
```

示例 14–1 显示 iSCSI 目标信息

以下示例说明如何显示特定 iSCSI 目标的参数值：

```
# iscsiadm list target-param iqn.1992-08.com.abcstorage:sn.33592219
        Target: iqn.1992-08.com.abcstorage:sn.33592219
```

`iscsiadm list target-param -v` 命令显示以下信息：

- 目标的验证值
- 目标登录参数的缺省值
- 每个登录参数的已配置值

`iscsiadm list target-param -v` 命令在 / 指示符前显示**缺省**参数值，并在 / 指示符后显示**已配置**参数值。如果未配置任何参数，其值将显示为连字符 (-)。有关更多信息，请参见以下示例。

```
# iscsiadm list target-param -v eui.50060e8004275511 Target: eui.50060e8004275511
        Alias: -
        Bi-directional Authentication: disabled
        Authentication Type: NONE
        Login Parameters (Default/Configured):
                Data Sequence In Order: yes/-
                Data PDU In Order: yes/-
                Default Time To Retain: 20/-
                Default Time To Wait: 2/-
                Error Recovery Level: 0/-
                First Burst Length: 65536/-
                Immediate Data: yes/-
                Initial Ready To Transfer (R2T): yes/-
                Max Burst Length: 262144/-
                Max Outstanding R2T: 1/-
                Max Receive Data Segment Length: 65536/-
                Max Connections: 1/-
                Header Digest: NONE/-
                Data Digest: NONE/-
        Configured Sessions: 1
```

以下示例输出显示了目标与启动器之间协商的参数：

```
# iscsiadm list target -v eui.50060e8004275511
Target: eui.50060e8004275511
        TPGT: 1
        ISID: 4000002a0000
        Connections: 1
                CID: 0
                  IP address (Local): 172.90.101.71:32813
                  IP address (Peer): 172.90.101.40:3260
                  Discovery Method: Static
                  Login Parameters (Negotiated):
                        Data Sequence In Order: yes
                        Data PDU In Order: yes
                        Default Time To Retain: 0
                        Default Time To Wait: 3
                        Error Recovery Level: 0
```

```
First Burst Length: 65536
Immediate Data: yes
Initial Ready To Transfer (R2T): yes
Max Burst Length: 262144
Max Outstanding R2T: 1
Max Receive Data Segment Length: 65536
Max Connections: 1
Header Digest: NONE
Data Digest: NONE
```

修改 iSCSI 启动器和目标参数

可以同时修改 iSCSI 启动器和 iSCSI 目标设备上的参数。但是，只能在 iSCSI 启动器上修改的参数包括以下各项：

- iSCSI 启动器节点名称－可以将启动器节点名称更改为不同的名称。如果更改了启动器节点名称，则系统可能会从启动器的目标列表中删除 iSNS 搜索到的目标，具体取决于更改名称时 iSNS 服务器上的搜索域配置。有关更多信息，请参见第 263 页中的 “如何修改 iSCSI 启动器和目标参数”。
- 头摘要－NONE（缺省值），或者 CRC32。
- 数据摘要－NONE（缺省值），或者 CRC32。
- 验证和 CHAP 密钥－有关设置验证的更多信息，请参见第 252 页中的 “如何配置对 iSCSI 启动器的 CHAP 验证”。

iSCSI 驱动程序为 iSCSI 启动器和 iSCSI 目标设备参数提供缺省值。如果修改了 iSCSI 启动器的参数，则修改后的参数将由 iSCSI 目标设备继承，除非 iSCSI 目标设备已有不同的值。

注意 – 请确保目标软件支持要修改的参数。否则，可能无法登录到 iSCSI 目标设备。有关支持的参数的列表，请参见数组文档。

应在启动器与目标之间的 I/O 完成后再修改 iSCSI 参数。使用 `iscsiadm modify` 命令进行更改后，iSCSI 驱动程序将重新连接会话。

调优 iSCSI 参数

可以调优 iSCSI 参数以适应 iSCSI 启动器的各种响应或连接时间值。可以调优 iSCSI 参数，取决于是否要在启动器上为所有目标调整参数（其中启动器系统已连接），或者为一个特定目标调整参数。

使用以下命令更改特定 iSCSI 目标的参数值。

```
iscsiadm modify target-param -T tunable-prop=value target-name
```

使用以下命令调整所有目标的参数值：

```
iscsiadm modify initiator-node -T tunable-prop=value
```

以下可调参数应用于活动连接，并更改 iSCSI 启动器和连接到启动器的目标的行为。配置 iSCSI 启动器时，动态调优参数的能力提供了灵活性。

表 14–1 iSCSI 可调参数

参数名	说明	有效值（秒）	缺省值（秒）
recv-login-rsp-timeout	会话登录响应时间 – 指定 iSCSI 启动器等待来自给定 iSCSI 目标的 iSCSI 会话登录请求的响应时间。	0－3600	60
conn-login-max	最大连接重试时间 – iSCSI 启动器连接到目标 I/O 超时或连接失败之后，确定 iSCSI 启动器尝试连接到目标的最大次数。	0－3600	180
polling-login-delay	登录重试时间间隔－iSCSI 启动器连接到目标 I/O 超时或连接失败之后，确定每个 iSCSI 会话登录重试之间的时间间隔。	0－3600	60

▼ 如何调优 iSCSI 参数

1 显示所有可调 iSCSI 参数。

显示所有目标的 iSCSI 参数信息。

```
# iscsiadm list initiator-node
Initiator node name: iqn.1986-03.com.sun:01:e00000000000.4e36d278
Initiator node alias: unknown
        Login Parameters (Default/Configured):
                Header Digest: NONE/-
                Data Digest: NONE/-
                Max Connections: 65535/-
        Authentication Type: NONE
        RADIUS Server: NONE
        RADIUS Access: disabled
        Tunable Parameters (Default/Configured):
                Session Login Response Time: 60/-
                Maximum Connection Retry Time: 180/-
                Login Retry Time Interval: 60/-
        Configured Sessions: 1
```

显示特定目标的 iSCSI 参数信息。

```
# iscsiadm list target-param [target-name]
```

例如：

```
# iscsiadm list target-param -v iqn.1986-03.com.sun:02:47ac0
506-cd48-67f5-fc0d-ab7544d37538
Target: iqn.1986-03.com.sun:02:47ac0506-cd48-67f5-fc0d-ab7544d37538
        Alias: -
        Bi-directional Authentication: disabled
        Authentication Type: NONE
        Login Parameters (Default/Configured):
                Data Sequence In Order: yes/-
                Data PDU In Order: yes/-
                Default Time To Retain: 20/-
                Default Time To Wait: 2/-
                Error Recovery Level: 0/-
                First Burst Length: 65536/-
                Immediate Data: yes/-
                Initial Ready To Transfer (R2T): yes/-
                Max Burst Length: 262144/-
                Max Outstanding R2T: 1/-
                Max Receive Data Segment Length: 8192/-
                Max Connections: 65535/-
                Header Digest: NONE/-
                Data Digest: NONE/-
        Tunable Parameters (Default/Configured):
                Session Login Response Time: 60/-
                Maximum Connection Retry Time: 180/-
                Login Retry Time Interval: 60/-
        Configured Sessions: 1
```

2 **调优 iSCSI 参数。**

例如，将一个目标的最大连接重试时间设为 90 秒：

```
# iscsiadm modify target-param -T conn-login-max=90 iqn.1986-03.com.sun:
02:47ac0506-cd48-67f5-fc0d-ab7544d37538
```

例如，将所有目标的最大连接重试时间设为 90 秒：

```
# iscsiadm modify initiator-node -T conn-login-max=90
```

▼ 如何修改 iSCSI 启动器和目标参数

此过程的第一部分说明了 iSCSI 目标设备如何继承 iSCSI 启动器的修改参数。此过程的第二部分说明如何在 iSCSI 目标设备上实际修改参数。

此可选过程假定您已登录到本地系统，在此系统上已配置了对 iSCSI 目标设备的访问权限。

1 **成为管理员。**

2 列出 iSCSI 启动器和目标设备的当前参数值。

a. 列出 iSCSI 启动器的当前参数值。

```
initiator# iscsiadm list initiator-node
Initiator node name: iqn.1986-03.com.sun:01:0003ba4d233b.425c293c
Initiator node alias: zzr1200
        Login Parameters (Default/Configured):
                Header Digest: NONE/-
                Data Digest: NONE/-
        Authentication Type: NONE
        RADIUS Server: NONE
        RADIUS access: unknown
        Configured Sessions: 1
```

b. 列出 iSCSI 目标设备的当前参数值。

```
initiator# iscsiadm list target-param -v iqn.1992-08.com.abcstorage:sn.84186266
Target: iqn.1992-08.com.abcstorage:sn.84186266
        Alias: -
        Bi-directional Authentication: disabled
        Authentication Type: NONE
        Login Parameters (Default/Configured):
                Data Sequence In Order: yes/-
                Data PDU In Order: yes/-
                Default Time To Retain: 20/-
                Default Time To Wait: 2/-
                Error Recovery Level: 0/-
                First Burst Length: 65536/-
                Immediate Data: yes/-
                Initial Ready To Transfer (R2T): yes/-
                Max Burst Length: 262144/-
                Max Outstanding R2T: 1/-
                Max Receive Data Segment Length: 65536/-
                Max Connections: 1/-
                Header Digest: NONE/-
                Data Digest: NONE/-
         Configured Sessions: 1
```

请注意，对于 iSCSI 启动器和 iSCSI 目标设备，当前已将头摘要和数据摘要参数设置为 NONE。

要查看 iSCSI 目标设备的缺省参数值，请参见示例 14–1中的 `iscsiadm list target-param` 输出。

3 修改 iSCSI 启动器的参数。

例如，将头摘要设置为 CRC32。

```
initiator# iscsiadm modify initiator-node -h CRC32
```

如果更改了启动器节点名称，则系统可能会注销 iSNS 搜索到的目标，并将其从启动器的目标列表中删除，前提是新名称与目标名称不属于同一搜索域。但是，如果当前正在使用目标，则不会将其删除。例如，如果某个文件已打开或已在这些目标上挂载了文件系统，将不会删除目标。

如果这些目标和新启动器节点名称属于同一搜索域，则在名称更改后还可能看到新目标。

4 验证是否已修改参数。

a. 显示 iSCSI 启动器的更新后参数信息。

```
initiator# iscsiadm list initiator-node
Initiator node name: iqn.1986-03.com.sun:01:0003ba4d233b.425c293c
Initiator node alias: zzr1200
        Login Parameters (Default/Configured):
                Header Digest: NONE/CRC32
                Data Digest: NONE/-
        Authentication Type: NONE
        RADIUS Server: NONE
        RADIUS access: unknown
        Configured Sessions: 1
```

请注意，现在已将头摘要设置为 CRC32。

b. 显示 iSCSI 目标设备的更新后参数信息。例如：

```
initiator# iscsiadm list target-param -v iqn.1992-08.com.abcstorage:sn.84186266
Target: iqn.1992-08.com.abcstorage:sn.84186266
        Alias: -
        Bi-directional Authentication: disabled
        Authentication Type: NONE
        Login Parameters (Default/Configured):
                Data Sequence In Order: yes/-
                Data PDU In Order: yes/-
                Default Time To Retain: 20/-
                Default Time To Wait: 2/-
                Error Recovery Level: 0/-
                First Burst Length: 65536/-
                Immediate Data: yes/-
                Initial Ready To Transfer (R2T): yes/-
                Max Burst Length: 262144/-
                Max Outstanding R2T: 1/-
                Max Receive Data Segment Length: 65536/-
                Max Connections: 1/-
                Header Digest: CRC32/-
                Data Digest: NONE/-
        Configured Sessions: 1
```

请注意，现在已将头摘要设置为 CRC32。

5 验证 iSCSI 启动器已重新连接至 iSCSI 目标。

```
initiator# iscsiadm list target -v iqn.1992-08.com.abcstorage:sn.84186266
Target: iqn.1992-08.com.abcstorage:sn.84186266
        TPGT: 2
        ISID: 4000002a0000
        Connections: 1
                CID: 0
                  IP address (Local): nnn.nn.nn.nnn:64369
                  IP address (Peer): nnn.nn.nn.nnn:3260
                  Discovery Method: SendTargets
                  Login Parameters (Negotiated):
                        .
                        .
                        .
                        Header Digest: CRC32
                        Data Digest: NONE
```

6 **（可选的）重置 iSCSI 启动器参数或 iSCSI 目标设备参数。**

通过使用 `iscsiadm modify` 命令可以将参数重置为其缺省值。或者，可以使用 `iscsiadm remove` 命令将所有参数重置为其缺省值。

`iscsiadm modify target-param` 命令仅更改在命令行中指定的参数。

以下示例说明如何将头摘要重置为 `NONE`：

```
initiator# iscsiadm modify target-param -h none iqn.1992-08.com.abcstorage:sn...
```

有关 `iscsiadm remove target-param` 命令的信息，请参见iscsiadm(1M)。

iSCSI 配置问题的故障排除

以下工具可用于解决一般的 iSCSI 配置问题：

- `snoop`－此工具已更新为支持 iSCSI 软件包。
- `wireshark`－可以从 `http://www.wireshark.org/` 获取此产品。

这两个工具都可以筛选端口 `3260` 上的 iSCSI 数据包。

以下各节介绍了各种 iSCSI 故障排除和错误消息解析方案。

从本地系统到 iSCSI 目标无连接

▼ 如何解决 iSCSI 连接问题

1 **成为管理员。**

2 **列出 iSCSI 目标信息。**

例如：

```
initiator# iscsiadm list target
Target: iqn.2001-05.com.abcstorage:6-8a0900-37ad70401-bcfff02df8a421df-zzr1200-01
        TPGT: default
        ISID: 4000002a0000
        Connections: 0
```

3 **如果 `iscsiadm list target` 输出中未列出任何连接，请检查 `/var/adm/messages` 文件，以找出连接失败的可能原因。**

通过使用 `ping` 命令或使用 `telnet` 命令与存储设备的 iSCSI 端口连接，还可验证连接是否可访问，从而可确保 iSCSI 服务可用。缺省端口为 `3260`。

此外，请检查存储设备的日志文件中是否有错误。

4 **如果目标未列在 `iscsiadm list target` 输出中，请检查 `/var/adm/messages` 文件，以找出可能的原因。**

如果当前使用 SendTargets 作为搜索方法，请尝试使用 `-v` 选项列出 *discovery-address*，以确保预期目标对于主机是可见的。例如：

```
initiator# iscsiadm list discovery-address -v 10.0.0.1
Discovery Address: 10.0.0.1:3260
    Target name: eui.210000203787dfc0
        Target address:        10.0.0.1:11824
    Target name: eui.210000203787e07b
        Target address:        10.0.0.1:11824
```

如果当前使用 iSNS 作为搜索方法，请尝试启用 iSNS 搜索模式，并使用 `-v` 选项列出 *isns-server*，以确保预期目标对主机是可见的。例如：

```
initiator# iscsiadm list isns-server -v
iSNS Server IP Address: 10.20.56.56:3205
        Target name: iqn.1992-08.com.xyz:sn.1234566
                Target address:   10.20.57.161:3260, 1
        Target name: iqn.2003-10.com.abc:group-0:154:abc-65-01
                Target address:   10.20.56.206:3260, 1
        Target name: iqn.2003-10.com.abc:group-0:154:abc-65-02
                Target address:   10.20.56.206:3260, 1
.
.
.
```

iSCSI 设备或磁盘在本地系统上不可用

▼ 如何解决 iSCSI 设备或磁盘的不可用性问题

1 **成为管理员。**

2 **确定在枚举过程中在该目标上搜索到的 LUN。**

`-S` 选项可显示在枚举过程中在该目标上搜索到的 LUN。

例如：

```
# iscsiadm list target -S
Target: iqn.2001-05.com.abcstorage:6-8a0900-37ad70401-bcfff02df8a421df-zzr1200-01
        TPGT: default
        ISID: 4000002a0000
        Connections: 1
        LUN: 0
            Vendor:  ABCSTOR
            Product: 0010
            OS Device Name: /dev/rdsk/c3t34d0s2

,
```

3 **请查看 `/var/adm/messages` 文件，以查看是否报告了错误。**

如果您认为某个 LUN 应该已列出，但却没有列出，请检查此日志文件。

4 **请检查存储设备的日志文件中是否有错误。**

5 **请确保任何存储设备 LUN 屏蔽均已正确配置。**

在使用 iSNS 搜索方法时使用 LUN 屏蔽

请避免将 iSNS 搜索域用作控制特定启动器的存储验证的方式。如果要确保只有获得授权的启动器才能访问 LUN，请改用 *LUN* **屏蔽**。

如果在目标正在使用时从搜索域中删除该目标，则 iSCSI 启动器将不会从该目标注销。如果不需要该启动器访问该目标（以及关联的 LUN），则必须使用 LUN 屏蔽。仅从搜索域中删除目标是不够的。

一般 iSCSI 错误消息

本节介绍可能在 `/var/adm/messages` 文件中找到的 iSCSI 消息以及可能的恢复解决方法。

消息格式如下：

`iscsi` *TYPE* (*OID*) *STRING* (*STATUS-CLASS#/STATUS-DETAIL#*)

TYPE	连接或会话。
OID	连接或会话的对象 ID。此 ID 对于 OS 实例是唯一的。
STRING	条件的说明。
STATUS-CLASS#/STATUS-DETAIL#	这些值将在 iSCSI 登录响应中返回，如 RFC 3720 所定义的那样。

`iscsi connection(`*OID*`) login failed - Miscellaneous iSCSI initiator errors.`

原因: 设备登录因某种形式的启动器错误而失败。

`iscsi connection(`*OID*`) login failed - Initiator could not be successfully authenticated.`

原因: 设备不能成功验证启动器。

解决方法: 如果适用，请验证 CHAP 名称、CHAP 口令或 RADIUS 服务器的设置是否正确。

`iscsi connection(OID) login failed - Initiator is not allowed access to the given target.`

原因: 该设备不允许启动器访问 iSCSI 目标设备。

解决方法: 验证启动器名称，并确认该名称已由存储设备正确屏蔽或提供。

`iscsi connection(OID) login failed - Requested ITN does not exist at this address.`

原因: 该设备不提供对正在请求的 iSCSI 目标名称 (iSCSI target name, ITN) 的访问权限。

解决方法: 验证正确指定了启动器搜索信息，并且正确配置了存储设备。

`iscsi connection(OID) login failed - Requested ITN has been removed and no forwarding address is provided.`

原因: 该设备无法再提供对正在请求的 iSCSI 目标名称 (ITN) 的访问权限。

解决方法: 验证正确指定了启动器搜索信息，并且正确配置了存储设备。

`iscsi connection(OID) login failed - Requested iSCSI version range is not supported by the target.`

原因: 存储设备不支持启动器的 iSCSI 版本。

`iscsi connection(OID) login failed - No more connections can be accepted on this Session ID (SSID).`

原因: 存储设备无法接受此启动器节点到 iSCSI 目标设备的更多连接。

`iscsi connection(OID) login failed - Missing parameters (e.g., iSCSI initiator and/or target name).`

原因: 存储设备正在报告未正确指定启动器或目标名称。

解决方法: 请正确指定 iSCSI 启动器或目标名称。

`iscsi connection(OID) login failed - Target hardware or software error.`

原因: 存储设备遇到硬件或软件错误。

解决方法: 请查阅存储文档或与存储设备供应商联系以获得进一步的帮助。

`iscsi connection(OID) login failed - iSCSI service or target is not currently operational.`

原因: 存储设备当前无法运行。

解决方法: 请查阅存储文档或与存储设备供应商联系以获得进一步的帮助。

`iscsi connection(`*OID*`) login failed - Target has insufficient session, connection or other resources.`
原因: 存储设备具有的资源不足。

解决方法: 请查阅存储文档或与存储设备供应商联系以获得进一步的帮助。

`iscsi connection(`*OID*`) login failed - unable to initialize authentication`

`iscsi connection( `*OID*`) login failed - unable to set authentication`

`iscsi connection(`*OID*`) login failed - unable to set username`

`iscsi connection(`*OID*`) login failed - unable to set password`

`iscsi connection(`*OID*`) login failed - unable to set ipsec`

`iscsi connection(`*OID*`) login failed - unable to set remote authentication`
原因: 该启动器无法正确初始化或设置验证。

解决方法: 验证是否已正确配置启动器的验证设置。

`iscsi connection(`*OID*`) login failed - unable to make login pdu`
原因: 启动器无法根据启动器或存储设备的设置生成登录有效负荷数据单元 (payload data unit, PDU)。

解决方法: 请尝试重置所有目标登录参数或其他非缺省设置。

`iscsi connection(`*OID*`) login failed - failed to transfer login`

`iscsi connection(`*OID*`) login failed - failed to receive login response`
原因: 启动器无法通过网络连接传输或接收登录有效负荷数据单元 (payload data unit, PDU)。

解决方法: 验证网络连接是否可以访问。

`iscsi connection(`*OID*`) login failed - received invalid login response (`*OP CODE*`)`
原因: 存储设备采用非预期响应来响应登录。

`iscsi connection(`*OID*`) login failed - login failed to authenticate with target`
原因: 启动器无法验证存储设备。

解决方法: 验证是否已正确配置启动器的验证设置。

`iscsi connection(`*OID*`) login failed - initiator name is required`
原因: 必须配置启动器名称才能执行所有操作。

解决方法: 验证是否已配置启动器名称。

`iscsi connection(`*OID*`) login failed - authentication receive failed`

`iscsi connection(`*OID*`) login failed - authentication transmit failed`

原因: 该启动器无法传输或接收验证信息。

解决方法: 根据情况验证存储设备或 RADIUS 服务器的网络连接。

`iscsi connection(`*OID*`) login failed - login redirection invalid`

原因: 存储设备尝试将启动器重定向到无效目标。

解决方法: 请查阅存储文档或与存储设备供应商联系以获得进一步的帮助。

`iscsi connection(`*OID*`) login failed - target protocol group tag mismatch, expected <TPGT>, received <TPGT>`

原因: 启动器和目标的 TPGT（target portal group tag，目标门户组标记）不匹配。

解决方法: 请验证启动器或存储设备上的 TPGT 搜索设置。

`iscsi connection(`*OID*`) login failed - can't accept` *PARAMETER* `in security stage`

原因: 设备已在登录安全阶段使用不支持的登录参数进行了响应。

解决方法: 请记录参数名称以供参考。请查阅存储文档或与存储设备供应商联系以获得进一步的帮助。

`iscsi connection(`*OID*`) login failed - HeaderDigest=CRC32 is required, can't accept` *VALUE*

`iscsi connection(`*OID*`) login failed - DataDigest=CRC32 is required, can't accept` *VALUE*

原因: 对于此目标，该启动器已配置为仅接受将 `HeaderDigest` 或 `DataDigest` 设置为 `CRC32` 的情况。设备返回值 *VALUE*。

解决方法: 验证启动器和设备摘要设置兼容。

`iscsi connection(`*OID*`) login failed - HeaderDigest=None is required, can't accept` *VALUE*

`iscsi connection(`*OID*`) login failed - DataDigest=None is required, can't accept` *VALUE*

原因: 对于此目标，该启动器已配置为只接受将 `HeaderDigest` 或 `DataDigest` 设置为 NONE 的情况。设备返回值 *VALUE*。

解决方法: 验证启动器和设备摘要设置兼容。

`iscsi connection(`*OID*`) login failed - can't accept` *PARAMETER*

原因: 启动器不支持此参数。

`iscsi connection(`*OID*`) login failed - can't accept MaxOutstandingR2T` *VALUE*
原因: 该启动器不接受 `MaxOutstandingR2T` 值为所记录 *VALUE* 的情况。

`iscsi connection(`*OID*`) login failed - can't accept MaxConnections` *VALUE*
原因: 该启动器不接受最大连接数为所记录 *VALUE* 的情况。

`iscsi connection(`*OID*`) login failed - can't accept ErrorRecoveryLevel` *VALUE*
原因: 该启动器不接受错误恢复级别值为所记录 *VALUE* 的情况。

`iscsi session(`*OID*`)` *NAME* `offline`
原因: 此目标 *NAME* 的所有连接都已删除或失败。

`iscsi connection(`*OID*`) failure - unable to schedule enumeration`
原因: 启动器无法枚举此目标上的 LUN。

解决方法: 可以通过运行 `devfsadm -i iscsi` 命令来强制执行 LUN 枚举。有关更多信息，请参见`devfsadm(1M)`。

`iscsi connection(`*OID*`) unable to connect to target` *NAME* `(errno:`*ERRNO* `)`
原因: 启动器无法建立网络连接。

解决方法: 有关连接故障的特定 *ERRNO* 的信息，请参见 `/usr/include/sys/errno.h` 文件。

◆ ◆ ◆ 第 15 章

15 配置和管理 Oracle Solaris Internet 存储名称服务 (Internet Storage Name Service, iSNS)

本章概述了 Internet 存储名称服务 (Internet Storage Name Service, iSNS)，并说明如何配置 Oracle Solaris iSNS 服务器、管理 iSNS 服务器和管理 iSNS 客户端。本章包含以下主题：

- 第 273 页中的 “iSNS 技术（概述）”
- 第 274 页中的 “配置 iSNS 服务器”
- 第 280 页中的 “管理 iSNS 服务器和客户端”

iSNS 技术（概述）

Internet 存储名称服务 (Internet Storage Name Service, iSNS) 是一个协议，允许 iSCSI 启动器和目标在 IP 存储区域网络 *SAN* 内进行动态搜索。iSNS 协议通过提供以下服务可以用于标识、连接至和管理 iSCSI 设备：

- **名称注册和搜索**：待存储的数据源（称为**启动器**）和存储对象（称为**目标**）注册其属性和地址，然后可以以动态方式获得关于可访问存储设备的信息。
- **搜索域和登录控制**：将典型的存储网络中的资源分成称为**搜索域**的组，可以通过网络管理应用程序对这些搜索域进行管理。搜索域通过向使用自身的访问控制无法启用的目标提供访问控制，同时将每个启动器的登录过程限制为网络中可用目标的相关子集，从而增强了安全性。
- **状态更改通知**：iSNS 服务器向相关 iSNS 客户端通知网络事件，例如，新创建的磁盘逻辑单元号 (Logical Unit Number, LUN)、脱机的存储资源、搜索域成员关系更改和网络中的链接故障。这些通知使得网络迅速适应拓扑中的更改，这对于可伸缩性和可用性都很关键。这是可选的服务。
- **实体状态查询**：iSNS 服务验证 iSNS 客户端是否可用。因此，可能发出状态更改通知。这是可选的服务。

在简单的配置中，待存储的数据源（启动器）与存储对象（目标）交换数据。启动器可以定位目标，并且目标始终会识别启动器。例如，Oracle StorageTek 5320 网络连接存储 (Network Attached Storage, NAS) 设备是 iSCSI 目标，因为该 NAS 设备存储数据。数据

来自各种 iSCSI 客户端，例如数据管理应用程序或作为启动器的网络接口卡。但是，在大且复杂的配置中，为每个目标配置每个启动器以及让每个目标识别每个启动器很困难，而且耗费时间。iSNS 服务器通过使用搜索和安全机制以动态方式自动识别启动器和目标，并管理与授权资源的连接，从而解决了该问题。

在 Oracle Solaris 系统已配置为 iSNS 服务器之后，所有的目标和启动器可以使用该服务器注册。目标和启动器成为 iSCSI **客户端**或 iSNS 服务器的**节点**。这些客户端是**缺省搜索域**的成员，即**缺省搜索域集**中唯一的域。当启用缺省搜索域集时，iSNS 服务器可以以一种简单方式为客户端提供 iSCSI 名称服务 (iSCSI Name Service, iSNS)。

要使用 iSCSI 名称服务的功能，请创建几个搜索域集和搜索域。然后将客户端指定到不同的域，重叠其成员关系。作为一个或多个搜索域的成员，iSNS 服务器保持跟踪客户端的状态。例如，当将新的存储设备添加到存储网络中，并且使用 iSNS 服务器注册时，其位于缺省搜索域集中的缺省搜索域中。然后将该目标指定到启动器会将其用作资源的搜索域。然后，iSNS 服务器将该目标作为缺省搜索域集中的缺省搜索域的成员删除。

所有的启动器和目标均指定到至少一个搜索域。将启动器指定到一个搜索域会限制其访问相同搜索域集中的那些目标。将启动器指定到几个搜索域使其可以找到并使用包括该启动器的搜索域在内的所有搜索域集中的目标。可以通过禁用或启用客户端的搜索域集管理对客户端的访问，而不影响其他的搜索域集中的客户端。

例如，除了缺省搜索域集，一个站点还具有两个搜索域集：生产和研究。在两个搜索域集内，除了缺省搜索域，还有三个域：开发、操作和财务。开发搜索域在研究搜索域集中，操作在生产域集中，而财务则是这两个搜索域集的成员。已将每个客户端指定到使用该客户端最频繁的搜索域集。操作搜索域中的数据应用程序可以定位并获取对生产搜索域集中的存储设备的访问权限，因为其为该搜索域集的成员，但是无法获取对研究搜索域集中的存储设备的访问权限。财务搜索域中的数据应用程序可以定位生产和研究搜索域集中的存储设备，因为其为这两个集的成员。如果禁用了研究搜索域集，财务搜索域中的启动器则不拥有对研究存储设备的访问权限，但是可以继续拥有对生产搜索域集中的存储设备的访问权限。

配置 iSNS 服务器

可以使用以下的任务列表和各节中的说明配置 iSNS 服务器。

任务		参考
1. 接受 iSNS 服务器的缺省属性，或更改这些属性。		
	A.服务器的状态更改通知	第 276 页中的 “如何设置服务器状态更改的通知”
	B.确定客户端的可用性的尝试次数	第 276 页中的 “如何设置客户端查询的重试次数”
	C.存储客户端数据的文件位置。	第 277 页中的 “如何指定数据存储位置”

任务	参考
2. 启用 iSNS 服务器并显示设置。	第 276 页中的 “如何安装 iSNS 服务器软件包” 第 277 页中的 “如何显示当前服务器配置”
3. 使用 iSNS 服务器注册所有客户端。	使用客户端的管理接口的 iSCSI 配置功能指定 iSNS 服务器的 IP 地址并允许搜索。
4. 启用缺省搜索域集。	第 278 页中的 “如何启用缺省搜索域集”

在这些任务之后，iSNS 服务器以最小的方式运行。所有的客户端都位于缺省搜索域中，并且未指定。每个客户端可以识别并获取对所有其他客户端的访问权限。

5. 为站点创建搜索域集。	第 278 页中的 “如何创建搜索域集”
6. 为站点创建搜索域。	第 278 页中的 “如何创建搜索域”
7. 将每个搜索域添加到一个或更多搜索域集。	第 279 页中的 “如何将搜索域添加到搜索域集”
8. 将客户端指定到一个或多个搜索域。	第 279 页中的 “如何将客户端指定到搜索域”
9. 验证搜索域中的客户端的成员关系以及搜索域集中的搜索域的成员关系。	第 281 页中的 “如何显示搜索域集的状态” 第 281 页中的 “如何显示搜索域的状态” 第 281 页中的 “如何显示客户端的状态”

下一节提供设置 iSNS 环境的说明。本章包含以下主题：

- 第 275 页中的 “设置 iSNS 管理设置”
- 第 277 页中的 “使用命令行界面配置 iSNS”

设置 iSNS 管理设置

本节介绍了更改 iSNS 服务的缺省管理设置和启动 iSNS 守护进程的步骤。如果在启动 iSNS 服务器之后更改设置，则需要刷新 iSNS 服务器。如果更改数据存储位置，则需要重新启动 iSNS 服务器。

- 第 276 页中的 “如何安装 iSNS 服务器软件包”
- 第 276 页中的 “如何设置服务器状态更改的通知”
- 第 276 页中的 “如何设置客户端查询的重试次数”
- 第 277 页中的 “如何指定数据存储位置”

有关这些操作的 `isns`(1M) 命令的详细信息，请参见手册页。

▼ 如何安装 iSNS 服务器软件包

安装 iSNS 服务器软件包和启动 iSNS 服务。

1. **成为管理员。**
2. **安装 iSNS 服务器软件包。**

```
# pkg install service/storage/isns
```

3. **启用 iSNS 服务。**

```
# svcadm enable isns_server
```

4. **验证该服务是否正在运行。**

```
# svcs svc:/network/isns_server:default
STATE          STIME    FMRI
online         16:10:49 svc:/network/isns_server:default
```

▼ 如何设置服务器状态更改的通知

缺省情况下，当 iSNS 服务器不可用时，会通知所有的客户端。要禁用这些通知，请更改 *Management_SCNs_Enabled* 属性。

1. **使用“iSNS 服务器管理”RBAC 配置文件获取管理 iSNS 服务所需授权。**

   角色包含授权和具有特权的命令。有关角色的更多信息，请参见《Oracle Solaris 管理：安全服务》中的“初次配置 RBAC（任务列表）”。

2. **使用 svccfg 命令禁用属性：**

```
# svccfg -s svc:/network/isns_server setprop config/Management_SCN_Enabled=false
```

3. **重新加载服务器配置：**

```
# svcadm refresh svc:/network/isns_server
```

▼ 如何设置客户端查询的重试次数

缺省重试次数为 3。如果服务器三次查询未获得响应，该服务器则将该客户端注册为不可用。要更改重试次数，请更改 *ESI Retry Threshold* 属性值。

1. **使用“iSNS 服务器管理”RBAC 配置文件获取管理 iSNS 服务所需授权。**

   角色包含授权和具有特权的命令。有关角色的更多信息，请参见《Oracle Solaris 管理：安全服务》中的“初次配置 RBAC（任务列表）”。

2. **例如使用 svccfg 命令将属性更改为 6 次重试：**

```
# svccfg -s svc:/network/isns_server setprop config/ESI_retry_threshold_count=6
```

3 **重新加载服务器配置：**

```
# svcadm refresh svc:/network/isns_server
```

▼ 如何指定数据存储位置

包含客户端数据的文件的缺省位置和名称为 `/etc/isns/isnsdata.xml`。如果具有包括一个或多个备份 iSNS 服务器的复杂网络环境，数据存储则必须位于通用位置，以便所有的服务器可以使用。使用 *data_store_location* 属性指定新的位置。还可以更改文件的名称。

1 **使用"iSNS 服务器管理"RBAC 配置文件获取管理 iSNS 服务所需授权。**

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《Oracle Solaris 管理：安全服务》中的"初次配置 RBAC（任务列表）"。

2 **例如使用 svccfg 命令将位置更改为 `/etc/isns2/isns_data.xml`：**

```
# svccfg -s svc:/network/isns_server setprop config/data_store_location="/etc/isns2/isns_data.xml"
```

3 **如果在服务器启用之后更改数据存储位置，则必须重新启动该服务器：**

```
# svcadm restart svc:/network/isns_server
```

使用命令行界面配置 iSNS

本节提供使用命令行界面配置 iSNS 服务器的步骤。本章包含以下主题：

- 第 277 页中的 "如何显示当前服务器配置"
- 第 278 页中的 "如何启用缺省搜索域集"
- 第 278 页中的 "如何创建搜索域集"
- 第 278 页中的 "如何创建搜索域"
- 第 279 页中的 "如何将搜索域添加到搜索域集"
- 第 279 页中的 "如何将客户端指定到搜索域"

这些步骤使用 `isnsadm`(1M) 命令。有关所有的命令选项的完整说明，请参见手册页。

▼ 如何显示当前服务器配置

- **以下命令显示了 iSNS 服务器的属性：**

```
# isnsadm show-config
        Data Store Location: /etc/isns/isnsdata.xml
        Entity Status Inquiry Non-Response Threshold: 3
        Management SCN Enabled: yes
        Authorized Control Node Names: -
```

▼ 如何启用缺省搜索域集

1 **使用“iSNS 服务器管理”RBAC 配置文件获取管理 iSNS 服务所需授权。**

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《Oracle Solaris 管理：安全服务》中的“初次配置 RBAC（任务列表）”。

2 **激活缺省搜索域集：**

```
# isnsadm enable-dd-set Default
```

▼ 如何创建搜索域集

1 **使用“iSNS 服务器管理”RBAC 配置文件获取管理 iSNS 服务所需授权。**

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《Oracle Solaris 管理：安全服务》中的“初次配置 RBAC（任务列表）”。

2 **创建搜索域集：**

```
# isnsadm create-dd-set set_name
```

3 **启用搜索域集：**

```
# isnsadm enable-dd-set set_name
```

4 **查看所有的搜索域集，包括新的搜索域集：**

```
# isnsadm list-dd-set -v
        DD Set name: Default
            State: Enabled
        DD Set name:set_name
            State: Enabled
```

搜索域集列表包括缺省搜索域集以及新的搜索域集。

▼ 如何创建搜索域

新的搜索域是缺省搜索域集的成员。创建新的搜索域之后，将其添加到新的搜索域集。

1 **使用“iSNS 服务器管理”RBAC 配置文件获取管理 iSNS 服务所需授权。**

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《Oracle Solaris 管理：安全服务》中的“初次配置 RBAC（任务列表）”。

2 **创建搜索域：**

```
# isnsadm create-dd domain_name
```

3 **在缺省搜索域集中查看新的搜索域：**

```
# isnsadm list-dd-set
        DD name: name
            DD set(s): Default
```

4 **创建其他搜索域。**

▼ 如何将搜索域添加到搜索域集

该任务将搜索域从缺省搜索域集中删除，并将其添加到指定的搜索域集。因为新的搜索域集已启用，所以可以通过 iSNS 服务器搜索其搜索域中的所有客户端。

无需具有列出搜索域和搜索域集成员的特权。

1 **使用“iSNS 服务器管理”RBAC 配置文件获取管理 iSNS 服务所需授权。**

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《Oracle Solaris 管理：安全服务》中的“初次配置 RBAC（任务列表）”。

2 **列出各个搜索域来确定需要添加的搜索域。**

```
# isnsadm list-dd -v
```

3 **列出各个搜索域集来确定需要包含新的搜索域的搜索域集。**

```
# isnsadm list-dd-set
```

4 **将搜索域移动到需要的搜索域集：**

```
# isnsadm add-dd domain_name -s set_name
```

5 **查看搜索域集中添加的新的搜索域：**

```
# isnsadm list-dd-set -v domain_name
```

▼ 如何将客户端指定到搜索域

开始之前 使用客户端的管理接口注册该客户端。使用 iSCSI 配置功能可以指定 iSNS 服务器的 IP 地址，并允许 iSNS 服务器搜索客户端。

1 **使用“iSNS 服务器管理”RBAC 配置文件获取管理 iSNS 服务所需授权。**

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《Oracle Solaris 管理：安全服务》中的“初次配置 RBAC（任务列表）”。

2 **验证是否已使用 iSNS 服务器注册客户端：**

```
# isnsadm list-node
       iSCSI Name: iqn.1986-03.com.sun:01:000e0c9f10da.45173FEA.engr
          Alias: STK5320_NAS
          Type: Target
          .
       iSCSI Name: iqn.1986-03.com.sun:01:000e0c9f10da.454F00A2.acct
```

```
        Alias:
        Type: Initiator
```

输出显示客户端的 iSNS 名称。

3 **验证搜索域的名称：**

```
# isnsadm list-dd
```

4 **将客户端添加到搜索域：**

```
# isnsadm add-node  -d domain_name  iSCSI_Name
```

例如，将名为“STK5320_NAS”的目标添加到 Eng-dd 搜索域：

```
# isnsadm add-node -d Eng-dd  iqn.1986-03.com.sun:01:000e0c9f10da.454F00A2.engr
```

5 **列出搜索域中的所有客户端来验证客户端是否已添加：**

```
# isnsadm list-dd -v domain_name
```

例如，检查 Eng-dd 搜索域：

```
# isnsadm list-dd -v Eng-dd
         DD name:    Eng-dd
         DD set: Development-dds
        iSCSI Name:    iqn.1986-03.com.sun:01:000e0c9f10da.45173FEA.engr
          iSCSI Name:    iqn.1986-03.com.sun:01:000e0c9f10da.454F00A2.acct
          iSCSI name: iqn.1986-03.com.sun:01:e00000000000.46fd8e2b
```

管理 iSNS 服务器和客户端

本节说明如何维护 iSNS 搜索域集及其成员、启动器和目标。随着站点的扩充，按照以下各节的说明继续添加客户端、搜索域和搜索域集：

- 第 278 页中的“如何创建搜索域集”
- 第 278 页中的“如何创建搜索域”
- 第 279 页中的“如何将搜索域添加到搜索域集”
- 第 279 页中的“如何将客户端指定到搜索域”

本节介绍管理 iSNS 服务器、使用命令行接口的其他步骤。本章包含以下主题：

- 第 281 页中的“如何显示搜索域集的状态”
- 第 281 页中的“如何显示搜索域的状态”
- 第 281 页中的“如何显示客户端的状态”
- 第 281 页中的“如何从搜索域中删除客户端”
- 第 282 页中的“如何从搜索域集中删除搜索域”
- 第 282 页中的“如何禁用搜索域集”
- 第 282 页中的“如何删除搜索域集”

▼ 如何显示搜索域集的状态

- 显示搜索域集的状态，并列出属于其成员的搜索域：

```
# isnsadm list-dd-set -v set_name
```

▼ 如何显示搜索域的状态

- 显示搜索域的状态，并列出属于其成员的客户端：

```
# isnsadm list-dd -v domain_name
```

▼ 如何显示客户端的状态

- 选择以下各项之一来显示客户端状态：
  - 显示所有客户端的状态：

```
# isnsadm list-node -v
```

  - 仅显示属于目标（即存储对象）的客户端的状态：

```
# isnsadm list-node -t
```

▼ 如何从搜索域中删除客户端

1. **使用“iSNS 服务器管理”RBAC 配置文件获取管理 iSNS 服务所需授权。**

   角色包含授权和具有特权的命令。有关角色的更多信息，请参见《Oracle Solaris 管理：安全服务》中的“初次配置 RBAC（任务列表）”。

2. **列出各个客户端来确定需要删除的客户端。**

```
# isnsadm list-node -v
       iSCSI Name: iqn.1986-03.com.sun:01:000e0c9f10da.45173FEA.engr
          Alias: STK5320_NAS
          Type: Target
          Network Entity: SE5310
          Portal: 172.20.57.95:3260
              Portal Group: 1
          Portal: 172.20.56.95:3260
              Portal Group: 1
          DD Name: Research,Finance
```

   输出显示客户端的 iSCSI 名称和属于其成员的搜索域的名称。

3. **从搜索域中删除客户端。**

```
# isnsadm remove-node -d domain_name iSCSI_name
```

▼ 如何从搜索域集中删除搜索域

1. **使用“iSNS 服务器管理”RBAC 配置文件获取管理 iSNS 服务所需授权。**
   角色包含授权和具有特权的命令。有关角色的更多信息，请参见《Oracle Solaris 管理：安全服务》中的“初次配置 RBAC（任务列表）”。
2. **列出各个搜索域来确定需要删除的搜索域。**
   ```
   # isnsadm list-dd -v
   ```
3. **从搜索域集中删除搜索域。**
   ```
   # isnsadm remove-dd set_name domain_name
   ```

▼ 如何禁用搜索域集

1. **使用“iSNS 服务器管理”RBAC 配置文件获取管理 iSNS 服务所需授权。**
   角色包含授权和具有特权的命令。有关角色的更多信息，请参见《Oracle Solaris 管理：安全服务》中的“初次配置 RBAC（任务列表）”。
2. **取消激活搜索域集：**
   ```
   # isnsadm disable-dd-set set_name
   ```
3. **验证搜索域集的状态是否已更改为 "Disabled"（已禁用）：**
   ```
   # isnsadm list-dd-set set_name
   ```

▼ 如何删除搜索域集

删除搜索域集之后，会保留其搜索域。搜索域必须至少是一个搜索域集的成员。

1. **使用“iSNS 服务器管理”RBAC 配置文件获取管理 iSNS 服务所需授权。**
   角色包含授权和具有特权的命令。有关角色的更多信息，请参见《Oracle Solaris 管理：安全服务》中的“初次配置 RBAC（任务列表）”。
2. **列出各个搜索域集来确定需要删除的搜索域集。**
   ```
   # isnsadm list-dd-set -v
   ```
3. **删除搜索域集。**
   ```
   # isnsadm remove-dd-set set_name
   ```

◆ ◆ ◆ 第 16 章

format 实用程序（参考）

本章介绍 format 实用程序的菜单和命令。

以下是本章中参考信息的列表。

- 第 283 页中的 “有关使用 format 实用程序的建议和要求”
- 第 284 页中的 “format 菜单和命令说明”
- 第 289 页中的 “format 命令的输入规则”
- 第 290 页中的 “获取有关 format 实用程序的帮助”

有关何时使用 format 实用程序的概述，请参见第 182 页中的 “format 实用程序”。

有关使用 format 实用程序的建议和要求

您必须是管理员或承担等效角色才能使用 format 实用程序。否则，尝试使用 format 实用程序时，将显示以下错误消息：

```
$ format
Searching for disks...done
No permission (or no disks found)!
```

使用 format 实用程序并需要保留现有数据时，请牢记以下原则：

- 备份磁盘驱动器上的所有文件。
- 使用 format 实用程序的 dump 命令将所有的缺陷列表都保存到文件中。文件名应包括驱动器类型、型号和序列号。
- 保存驱动器附带的制造商缺陷列表的书面副本。

format 菜单和命令说明

format 主菜单的显示与以下内容类似：

```
FORMAT MENU:
        disk       - select a disk
        type       - select (define) a disk type
        partition  - select (define) a partition table
        current    - describe the current disk
        format     - format and analyze the disk
        fdisk      - run the fdisk program (x86 only)
        repair     - repair a defective sector
        label      - write label to the disk
        analyze    - surface analysis
        defect     - defect list management
        backup     - search for backup labels
        verify     - read and display labels
        save       - save new disk/partition definitions
        inquiry    - show vendor, product and revision
        volname    - set 8-character volume name
        !<cmd>     - execute <cmd>, then return
        quit
format>
```

下表描述了 format 实用程序的主菜单项。

表 16–1 format 实用程序的主菜单项说明

菜单项	命令或菜单？	说明
disk	命令	列出系统的所有驱动器。此外，还可用于选择要在后续操作中使用的磁盘。该磁盘称为当前磁盘。
type	命令	标识当前磁盘的制造商和型号。另外，还可显示已知驱动器类型列表。可用于为所有 SCSI-2 磁盘驱动器选择 Auto configure 选项。
partition	菜单	创建和修改分片。有关更多信息，请参见 第 285 页中的 “partition 菜单”。
current	命令	显示以下有关当前磁盘的信息： ■ 设备名称和设备类型 ■ 柱面、备用柱面、磁头和扇区的数目 ■ 物理设备名称
format	命令	按照一定顺序使用以下信息源之一格式化当前磁盘： 1. 在 format.dat 文件中找到的信息 2. 来自自动配置过程的信息 3. 不存在 format.dat 项时在提示符下键入的信息 此命令不适用于 IDE 磁盘。IDE 磁盘由制造商进行预先格式化。

表 16–1 format 实用程序的主菜单项说明 （续）

菜单项	命令或菜单？	说明
fdisk	菜单	仅限于 x86 平台：运行 fdisk 程序可创建 Solaris fdisk 分区。 在容量超过 1 TB 且带有 EFI 标签的磁盘上，不能使用 fdisk 命令。
repair	命令	修复当前磁盘上的特定块。
label	命令	将新标签写入当前磁盘。
analyze	菜单	运行读取、写入和比较测试。有关更多信息，请参见 第 287 页中的 “analyze 菜单”。
defect	菜单	检索并显示缺陷列表。有关更多信息，请参见 第 288 页中的 “defect 菜单”。此功能不适用于 IDE 磁盘。IDE 磁盘会自动管理缺陷。
backup	命令	**VTOC**－搜索备份标签。 **EFI**－不支持。
verify	命令	显示以下有关当前磁盘的信息： ■ 设备名称和设备类型 ■ 柱面、备用柱面、磁头和扇区的数目 ■ 分区表
save	命令	**VTOC**－保存新的磁盘和分区信息。 **EFI**－不适用。
inquiry	命令	**仅限于 SCSI**－显示当前磁盘的供应商、产品名称和修订版级别。
volname	命令	使用指定的由八个字符组成的新卷名为磁盘设置标签。
quit	命令	退出 format 菜单。

partition 菜单

partition 菜单的显示与以下内容类似：

```
format> partition
PARTITION MENU:
        0      - change ‘0’ partition
        1      - change ‘1’ partition
        2      - change ‘2’ partition
        3      - change ‘3’ partition
        4      - change ‘4’ partition
        5      - change ‘5’ partition
        6      - change ‘6’ partition
        7      - change ‘7’ partition
```

```
        select - select a predefined table
        modify - modify a predefined partition table
        name   - name the current table
        print  - display the current table
        label  - write partition map and label to the disk
        quit
partition>
```

下表描述了 partition 菜单项。

表 16–2 partition 菜单项的说明

子命令	说明
change '*n*' partition	用于指定新分区的以下信息： ■ 标识标记 ■ 权限标志 ■ 起始柱面 ■ 大小
select	用于选择预定义的分区表。
modify	用于更改分区表中的所有分片。此命令优先于各个 change '*x*' partition 命令。
name	用于为当前分区表指定名称。
print	显示当前分区表。
label	将分区映射和标签写入当前磁盘。
quit	退出 partition 菜单。

x86: fdisk 菜单

fdisk 菜单仅出现在基于 x86 的系统上，其显示与以下内容类似。

```
format> fdisk
            Total disk size is 14169 cylinders
            Cylinder size is 2510 (512 byte) blocks

                                             Cylinders
      Partition   Status    Type          Start   End   Length    %
      =========   ======    ============  =====   ===   ======   ===
          1       Active    x86 Boot          1     9       9      0
          2                 Solaris2         10  14168    14159    100

SELECT ONE OF THE FOLLOWING:

  1. Create a partition
  2. Specify the active partition
  3. Delete a partition
  4. Change between Solaris and Solaris2 Partition IDs
```

```
  5. Exit (update disk configuration and exit)
  6. Cancel (exit without updating disk configuration)
  Enter Selection:
```

下表描述了 fdisk 菜单项。

表 16–3 x86: fdisk 菜单项的说明

菜单项	说明
Create a partition	创建 fdisk 分区。必须为每个 OS（如 Oracle Solaris 或 DOS）创建单独的分区。每个磁盘最多包含四个分区。系统会提示以磁盘百分比形式输入 fdisk 分区的大小。
Specify the active partition	用于指定要用于引导的分区。此菜单项可标识第一阶段引导程序查找第二阶段引导程序的位置。
Delete a partition	删除以前创建的分区。此命令将破坏分区中的所有数据。
Change between Solaris and Solaris2 Partition IDs	将分区 ID 从 130 (0x82) 更改为 191 (0xbf) 并执行相反操作。
Exit (update disk configuration and exit)	写入新版本的分区表并退出 fdisk 菜单。
Cancel (exit without updating disk configuration)	退出 fdisk 菜单，而不修改分区表。

analyze 菜单

analyze 菜单的显示与以下内容类似。

```
format> analyze

ANALYZE MENU:
    read     - read only test   (doesn't harm SunOS)
    refresh  - read then write  (doesn't harm data)
    test     - pattern testing  (doesn't harm data)
    write    - write then read      (corrupts data)
    compare  - write, read, compare (corrupts data)
    purge    - write, read, write   (corrupts data)
    verify   - write entire disk, then verify (corrupts data)
    print    - display data buffer
    setup    - set analysis parameters
    config   - show analysis parameters
    quit
analyze>
```

下表描述了 analyze 菜单项。

表 16–4 analyze 菜单项的说明

子命令	说明
read	读取当前磁盘上的每个扇区。缺省情况下会修复有缺陷的块。
refresh	读取当前磁盘上的数据然后再将数据写入磁盘，而不损害数据。缺省情况下会修复有缺陷的块。
test	将一组模式写入磁盘，而不损害数据。缺省情况下会修复有缺陷的块。
write	将一组模式写入磁盘，然后读回磁盘上的数据。破坏磁盘上的现有数据。缺省情况下会修复有缺陷的块。
compare	将一组模式写入磁盘，读回数据，然后将其与写缓冲区中的数据进行比较。破坏磁盘上的现有数据。缺省情况下会修复有缺陷的块。
purge	从磁盘中删除所有数据，以便不能通过任何方式检索数据。通过在整个磁盘（或磁盘的一部分）范围写入三种不同的模式来删除数据。如果验证通过，则将在整个磁盘（或磁盘的一部分）范围内写入十六进制位模式。 缺省情况下会修复有缺陷的块。
verify	第一遍将特殊数据写入整个磁盘上的每个块。第二遍读取并验证这些数据。破坏磁盘上的现有数据。缺省情况下会修复有缺陷的块。
print	显示读/写缓冲区中的数据。

setup	用于指定以下分析参数：
	`Analyze entire disk? yes`
	`Starting block number:` *depends on drive*
	`Ending block number:` *depends on drive*
	`Loop continuously? no`
	`Number of passes: 2`
	`Repair defective blocks? yes`
	`Stop after first error? no`
	`Use random bit patterns? no`
	`Number of blocks per transfer: 126 (0/`*n*`/`*nn*`)`
	`Verify media after formatting? yes`
	`Enable extended messages? no`
	`Restore defect list? yes`
	`Restore disk label? yes`

config	显示当前分析参数。
quit	退出 analyze 菜单。

defect 菜单

defect 菜单的显示与以下内容类似：

```
format> defect

DEFECT MENU:
        primary  - extract manufacturer's defect list
        grown    - extract manufacturer's and repaired defects lists
        both     - extract both primary and grown defects lists
        print    - display working list
        dump     - dump working list to file
        quit
defect>
```

下表描述了 defect 菜单项。

表 16–5 defect 菜单项说明

子命令	说明
primary（主）	从磁盘驱动器读取制造商的缺陷列表并更新内存中的缺陷列表。
grown	读取产生的缺陷列表，然后更新内存中的缺陷列表。产生的缺陷是指在分析过程中检测到的缺陷。
both	读取制造商的缺陷列表和产生的缺陷列表。然后更新内存中的缺陷列表。
print	显示内存中的缺陷列表。
dump	将内存中的缺陷列表保存到文件中。
quit	退出 defect 菜单。

format 命令的输入规则

使用 format 实用程序时，需要提供各种信息。本节介绍这类信息的规则。有关在指定数据时使用 format 的帮助功能的信息，请参见第 290 页中的 “获取有关 format 实用程序的帮助”。

为 format 命令指定数字

format 实用程序中的多个位置都要求使用数字作为输入。必须指定相应数据，或从选项列表中选择一个数字。在任一情况下，帮助功能都会导致 format 显示预期的数字上限和下限。只需输入合适的数字即可。如果未将基数显式指定为数字的一部分（例如，0x 表示十六进制），则假定该数字采用十进制格式。

以下是整数输入的示例：

```
Enter number of passes [2]: 34
Enter number of passes [34] 0xf
```

指定 format 命令名称

只要 format 实用程序显示菜单提示，就需要将命令名称作为输入。可以缩写命令名称，只要键入的内容可充分唯一标识所需命令即可。

例如，使用 p 访问 format 菜单中的 partition 菜单。那么，键入 p 显示当前的分片表。

```
format> p
PARTITION MENU:
        0      - change ‘0’ partition
        1      - change ‘1’ partition
        2      - change ‘2’ partition
        3      - change ‘3’ partition
        4      - change ‘4’ partition
        5      - change ‘5’ partition
        6      - change ‘6’ partition
        7      - change ‘7’ partition
        select - select a predefined table
        modify - modify a predefined partition table
        name   - name the current table
        print  - display the current table
        label  - write partition map and label to the disk
        quit
partition> p
```

为 format 命令指定磁盘名称

在 format 实用程序的某些位置，必须指定一些名称。在这些情况下，可以自由指定名称所需的任何字符串。如果名称中包含空格，则必须使用双引号 (") 将整个名称引起来。否则，仅使用名称的第一个单词。

例如，如果要标识磁盘的特定分区表，则可以使用 partition 菜单中可用的 name 子命令：

```
partition> name
Enter table name (remember quotes): "new disk3"
```

获取有关 format 实用程序的帮助

format 实用程序提供了 format 实用程序每次要求输入时可以使用的帮助功能。通过键入问号 (?)，可以请求有关要求的输入的帮助。 format 实用程序可显示所需输入类型的简短说明。

如果在菜单提示下键入 ?，则将显示可用命令的列表。

与 format 实用程序关联的手册页包括以下信息：

- format(1M)－介绍基本的 format 实用程序功能并提供所有命令行变量的说明。

- format.dat(4)－介绍 format 实用程序的磁盘驱动器配置信息。

◆ ◆ ◆ 第 17 章

管理文件系统（概述）

管理文件系统是最重要的系统管理任务之一。

以下是本章中概述信息的列表。

- 第 293 页中的 “Oracle Solaris 文件系统中的新增功能”
- 第 294 页中的 “有关文件系统管理任务的参考信息”
- 第 295 页中的 “文件系统概述”
- 第 295 页中的 “Oracle Solaris 文件系统的类型”
- 第 300 页中的 “缺省的 Oracle Solaris 文件系统”
- 第 299 页中的 “交换空间”
- 第 301 页中的 “挂载和取消挂载文件系统概述”
- 第 304 页中的 “确定文件系统的类型”

Oracle Solaris 文件系统中的新增功能

本节介绍 Oracle Solaris 发行版中文件系统方面的新增功能。

- 第 293 页中的 “文件系统监视工具 (fsstat)”
- 第 294 页中的 “Oracle Solaris ZFS 文件系统”

文件系统监视工具 (fsstat)

Oracle Solaris 11：`fsstat` 是一种新的文件系统监视工具，可用于报告文件系统操作。您可以使用多种选项报告活动，例如按挂载点或按文件系统类型。

例如，以下 `fsstat` 命令会显示自 ZFS 模块加载后的所有 ZFS 文件系统操作：

```
$ fsstat zfs
 new  name   name  attr  attr lookup rddir  read read  write write
 file remov  chng   get   set    ops   ops   ops bytes   ops bytes
 268K  145K 93.6K 28.0M 71.1K   186M 2.74M 12.9M 56.2G 1.61M 9.46G zfs
```

例如，以下 fsstat 命令会显示自 /export/ws 文件系统挂载后的所有文件系统操作。

```
$ fsstat /export/ws
new  name   name  attr  attr lookup rddir  read read  write write
file remov  chng   get   set    ops   ops   ops bytes   ops bytes
   0     0     0 18.1K     0  12.6M    52     0     0     0     0 /export/ws
```

缺省形式是采用易理解的值（例如 GB、KB 和 MB）报告统计信息。

有关更多信息，请参见 fsstat(1M)。

Oracle Solaris ZFS 文件系统

Oracle Solaris 11：Oracle Solaris ZFS 是一种具有革命性的全新文件系统，可提供简单管理、事务性语义、端到端数据完整性以及极好的可伸缩性。此外，ZFS 还可提供以下管理功能：

- 设备管理支持
- 持久性快照和克隆功能
- 可以为文件系统设置的配额
- 基于 ACL 的访问控制
- 为文件系统预留存储池空间
- 支持安装了区域的 Oracle Solaris 系统

有关使用 ZFS 的更多信息，请参见《Oracle Solaris 管理：ZFS 文件系统》。

有关文件系统管理任务的参考信息

请参见以下参考内容，以查找管理文件系统的逐步说明。

文件系统管理任务	更多信息
连接和配置新的磁盘设备。	第 10 章，管理磁盘（概述）
创建并挂载新的文件系统。	第 18 章，创建和挂载文件系统（任务）
使远程文件可供用户使用。	《Oracle Solaris 管理：网络服务》中的第 5 章“网络文件系统管理（任务）”

文件系统概述

文件系统是用于组织和存储文件的目录结构。术语**文件系统**可用于描述以下各项：

- 特定类型的文件系统：基于磁盘、基于网络或虚拟
- 整个文件树，从根 (/) 目录开始
- 磁盘分片或其他介质存储设备的数据结构
- 文件树结构的一部分，已连接到主文件树上的挂载点，以便可以访问文件

通常，可根据上下文知道具体的含义。

Oracle Solaris OS 使用**虚拟文件系统** (virtual file system, VFS) 体系结构，该体系结构可为不同的文件系统类型提供标准接口。VFS 体系结构使内核可处理读取、写入和列出文件等基本操作。VFS 体系结构还使添加新文件系统变得更容易。

Oracle Solaris 文件系统的类型

Oracle Solaris OS 支持三种类型的文件系统：

- 基于磁盘
- 基于网络
- 虚拟

要确定文件系统的类型，请参见第 304 页中的 “确定文件系统的类型”。

Oracle Solaris 基于磁盘的文件系统

基于磁盘的文件系统存储在硬盘和 DVD 等物理介质上。可以采用不同的格式向基于磁盘的文件系统中写入数据。下表介绍了可用的格式。

基于磁盘的文件系统	格式说明
ZFS	ZFS 是 Oracle Solaris 11 发行版中缺省的基于磁盘的根文件系统。有关更多信息，请参见《Oracle Solaris 管理：ZFS 文件系统》。
UFS	传统 UNIX 文件系统（基于 4.3 Tahoe 发行版中提供的 BSD Fat Fast File 系统）。
HSFS	High Sierra、Rock Ridge 和 ISO 9660 文件系统。High Sierra 是第一个 CD-ROM 文件系统。ISO 9660 是 High Sierra 文件系统的官方标准版本。HSFS 文件系统在 CD-ROM 上使用，它是只读文件系统。Oracle Solaris HSFS 支持 ISO 9660 的 Rock Ridge 扩展。这些扩展存在于 CD-ROM 上时，可提供所有文件系统功能和文件类型，但可写性和硬链接除外。
PCFS	PC 文件系统，该系统可用于对 DOS 格式磁盘上（针对基于 DOS 的个人计算机上写入）的数据和程序进行读写访问。

基于磁盘的文件系统	格式说明
UDFS	通用磁盘格式 (Universal Disk Format, UDF) 文件系统，这是使用称为 DVD（Digital Versatile Disc 或 Digital Video Disc，数字通用光盘或数字视频光盘）的光介质技术来存储信息的行业标准格式。

每种类型的基于磁盘的文件系统通常都与特定的介质设备关联，如下所示：

- ZFS 或 UFS 与硬盘
- HSFS 与 CD-ROM
- PCFS 与 USB 磁盘
- UDF 与 DVD

但是，这些关联并不具有限制性。例如，可以在 DVD 上创建 ZFS 或 UFS 文件系统。

通用磁盘格式 (Universal Disk Format, UDF) 文件系统

有关在可移除介质上创建 UDFS 文件系统的信息，请参见第 31 页中的 “如何在可移除介质上创建文件系统”。

UDF 文件系统是用于在 *DVD*（Digital Versatile Disc 或 Digital Video Disc，数字通用光盘或数字视频光盘）光介质上存储信息的行业标准格式。

UDF 文件系统是作为可动态装入的 32 位和 64 位模块提供的，并且使用系统管理实用程序同时在 SPARC 和 x86 平台上创建、挂载和检查该文件系统。Oracle Solaris UDF 文件系统适用于支持的 ATAPI 和 SCSI DVD 驱动器、CD-ROM 设备以及磁盘驱动器。此外，Oracle Solaris UDF 文件系统完全符合 UDF 1.50 规范。

UDF 文件系统可提供以下功能：

- 可以访问行业标准的 CD-ROM 和 DVD-ROM 介质，前提是这些介质包含 UDF 文件系统
- 跨平台和操作系统灵活交换信息
- 使用基于 UDF 格式的 DVD 视频规范，实现具有丰富的广播质量视频、高质量声音和交互性的新应用程序的机制

UDF 文件系统中不包括以下功能：

- 对使用连续一次写入记录方式和增量记录方式的一次写入介质 (CD-RW) 的支持

UDF 文件系统需要以下内容：

- 支持的 SPARC 或 x86 平台
- 支持的 CD-ROM 或 DVD-ROM 设备

Oracle Solaris UDF 文件系统实现提供了以下内容：

- 对行业标准读/写 UDF 版本 1.50 的支持

- 完全国际化的文件系统实用程序

基于网络的文件系统

可以从网络对**基于网络的文件系统**进行访问。通常，基于网络的文件系统驻留在一个系统上（通常是服务器），并由其他系统通过网络进行访问。

通过 NFS 服务，可以提供分布式**资源**（文件或目录），方法是从服务器共享这些资源，然后在单独的客户机上进行挂载。有关更多信息，请参见第 302 页中的 “NFS 环境”。

通过 Oracle SMB 服务，可以向 Windows 和 Mac OS 系统提供分布式**资源**（文件或目录），方法是从服务器共享这些资源，然后在单独的客户机上进行挂载。有关更多信息，请参见第 303 页中的 “Oracle Solaris SMB 服务”。

虚拟文件系统

虚拟文件系统是基于内存的文件系统，该文件系统提供对特殊内核信息和工具的访问。大多数虚拟文件系统不占用文件系统磁盘空间。另外，一些虚拟文件系统（如临时文件系统 (temporary file system, TMPFS)）还会使用磁盘上的交换空间。

临时文件系统

临时文件系统 (temporary file system, TMPFS) 使用本地内存进行文件系统的读取和写入。使用 TMPFS 可以节省利用本地磁盘或网络读取和写入临时文件的成本，从而提高系统性能。例如，在编译程序时即会创建临时文件。OS 在处理这些文件的同时会产生大量的磁盘活动或网络活动。使用 TMPFS 保存这些临时文件可以显著加快对其进行创建、处理和删除的速度。

TMPFS 文件系统中的文件不是永久性的。在取消挂载文件系统以及关闭或重新引导系统时，将删除这些文件。

TMPFS 是 Oracle Solaris OS 中 `/tmp` 目录的缺省文件系统类型。可以将文件复制或移动到 `/tmp` 目录中或从中复制或移动文件，就像在 ZFS 或 UFS 文件系统中进行的操作一样。

TMPFS 文件系统使用交换空间作为临时后备存储。如果包含 TMPFS 文件系统的系统没有充足的交换空间，则可能会出现以下两种问题：

- TMPFS 文件系统的空间可能不足，就像常规文件系统一样。
- 由于 TMPFS 会分配交换空间以保存文件数据（如有必要），因此一些程序可能因交换空间不足而无法执行。

有关创建 TMPFS 文件系统的信息，请参见第 18 章，创建和挂载文件系统（任务）。有关增加交换空间的信息，请参见第 19 章，配置附加交换空间（任务）。

回送文件系统

使用回送文件系统 (loopback file system, LOFS)，可以创建新的虚拟文件系统，以便使用替代的路径名访问文件。例如，可以在 /tmp/newroot 上创建根 (/) 目录的回送挂载。此回送挂载使整个文件系统分层结构看起来就像在 /tmp/newroot 下对其进行复制一样，其中包括从 NFS 服务器挂载的任何文件系统。路径名以根目录 (/) 或 /tmp/newroot 开头的所有文件都将是可访问的。

有关如何创建 LOFS 文件系统的信息，请参见第 18 章，创建和挂载文件系统（任务）。

进程文件系统

进程文件系统 (process file system, PROCFS) 驻留在内存中，并且包含 /proc 目录中按进程号排序的活动进程的列表。/proc 目录中的信息由 ps 等命令使用。调试器和其他开发工具也可以使用文件系统调用来访问进程的地址空间。

注意 – 请勿删除 /proc 目录中的文件。从 /proc 目录中删除进程不会中止这些进程。/proc 文件不占用磁盘空间，因此不必从此目录中删除文件。

/proc 目录无需管理。

其他虚拟文件系统

以下列出的其他类型的虚拟文件系统用于参考。这些文件系统无需管理。

虚拟文件系统	说明
CTFS	CTFS（contract file system，合同文件系统）是用于创建、控制和查看合同的界面。合同可通过提供更丰富的错误报告，以及采用延迟资源删除时间的方法（可选），增强进程与其所依赖的系统资源之间的关系。 服务管理工具 (service management facility, SMF) 使用进程合同（一种合同类型）来跟踪构成服务的进程，以便可以将多进程服务某部分中出现的故障标识为该服务的故障。
FIFOFS（先入先出）	为进程提供对数据的公共访问的指定管道文件
FDFS（文件描述符）	使用文件描述符提供用于打开文件的显式名称
MNTFS	为本地系统提供对已挂载文件系统表的只读访问
NAMEFS	主要由 STREAMS 用来在文件顶部动态挂载文件描述符
OBJFS	OBJFS（对象）文件系统可描述当前由内核装入的所有模块的状态。此文件系统由调试器用来访问有关内核符号的信息，而不必直接访问内核。

虚拟文件系统	说明
SHAREFS	提供对本地系统共享文件系统表的只读访问
SPECFS（特殊）	提供对字符特殊设备和块设备的访问
SWAPFS	由内核使用以进行交换

扩展的文件属性

ZFS、UFS、NFS 和 TMPFS 文件系统均已增强，包括扩展的文件属性。通过扩展的文件属性，应用程序开发者可以将特定属性与文件关联。例如，开发用于管理窗口系统的应用程序的开发者可能会选择将显示图标与文件关联。扩展的文件属性在逻辑上表示为与目标文件关联的隐藏目录中的文件。

可以使用 `runat` 命令在扩展属性名称空间中添加属性和执行 shell 命令。此名称空间是与指定文件关联的隐藏属性目录。

要使用 `runat` 命令向文件添加属性，必须首先创建属性文件。

```
$ runat filea cp /tmp/attrdata attr.1
```

然后，使用 `runat` 命令列出文件的属性。

```
$ runat filea ls -l
```

有关更多信息，请参见 `runat(1)` 手册页。

许多 Oracle Solaris 文件系统命令已经修改，可以通过提供可识别属性的选项来支持文件系统属性。使用此选项可查询、复制或查找文件属性。有关更多信息，请参见每个文件系统命令的特定手册页。

交换空间

Oracle Solaris OS 将一些磁盘分片用于临时存储而不是用于文件系统。这些分片称为**交换**分片或**交换空间**。如果系统没有足够的物理内存来处理当前进程，则可将交换空间用于虚拟内存存储区域。

由于许多应用程序依赖于交换空间，因此您应该知道如何规划、监视和添加更多交换空间（如有需要）。有关交换空间的概述和添加交换空间的说明，请参见第 19 章，配置附加交换空间（任务）。

缺省的 Oracle Solaris 文件系统

ZFS 文件系统是一个分层文件系统，从根目录 (/) 开始，向下延伸出许多目录。在 Oracle Solaris 安装过程中，可以安装一组缺省目录，并使用一组约定将类似的文件类型组合在一起。

有关 Oracle Solaris 文件系统和目录的简要概述，请参见 `filesystem(5)`。

下表汇总了缺省的 Oracle Solaris 文件系统。

表 17–1 缺省的 Oracle Solaris 文件系统

文件系统或目录	文件系统类型	说明
根 (/)	ZFS	分层文件树的顶层。根 (/) 目录包含对系统操作至关重要的目录和文件，如内核、设备驱动程序和用于引导系统的程序。根 (/) 目录还包含挂载点目录，可以在这些目录中将本地文件系统和远程文件系统附加到文件树。
`/usr`	ZFS	可以与其他用户共享的系统文件和目录。仅在某些类型的系统上运行的文件（例如 SPARC 可执行文件）包含在 `/usr` 文件系统或目录中。可以在所有类型的系统上使用的文件（例如手册页）可能包含在 `/usr/share` 目录中。
`/export/home` 或 `/home`	NFS 或 ZFS	用户起始目录（用于存储用户的工作文件）的挂载点。缺省情况下，`/home` 目录是自动挂载的文件系统。
`/var`	ZFS	可能随本地系统的生命周期而更改或增长的系统文件和目录。这些系统文件和目录包括系统日志，例如 `vi` 和 `ex` 备份文件。
`/opt`	NFS 或 ZFS	第三方软件的可选挂载点。在一些系统上，`/opt` 目录可能是 UFS 文件系统或 ZFS 文件系统。
`/tmp`	TMPFS	临时文件，每次引导系统或取消挂载 `/tmp` 文件系统将会删除这些文件。
`/proc`	PROCFS	按进程号排序的活动进程的列表。
`/etc/mnttab`	MNTFS	一种虚拟文件系统，可为本地系统提供对已挂载文件系统表的只读访问。
`/system/volatile`	TMPFS	基于内存的文件系统，用于存储引导系统后不需要的临时文件。
`/system/contract`	CTFS	一种虚拟文件系统，用于维护合同信息。
`/system/object`	OBJFS	一种虚拟文件系统，由调试器用来访问有关内核符号的信息，而不必直接访问内核。

挂载和取消挂载文件系统概述

需要首先挂载文件系统，然后才能访问该文件系统上的文件。挂载文件系统时，需要将该文件系统附加到目录（**挂载点**）并使其对系统可用。始终会挂载根目录 (/) 文件系统。可以将其他任何文件系统与根目录 (/) 文件系统连接或断开。

大多数文件系统在系统引导时通过 SMF 服务自动进行挂载。通常无需对文件系统手动进行挂载或取消挂载。有关挂载不同文件系统类型的更多信息，请参见第 306 页中的 “挂载和取消挂载 Oracle Solaris 文件系统”。

挂载文件系统时，只要挂载了文件系统，底层挂载点目录中的任何文件或目录都将无法使用。挂载进程不会永久影响这些文件。取消挂载文件系统时，这些文件即重新变为可用。但是，由于通常不希望隐藏现有文件，因此挂载目录经常为空。

有关如何挂载文件系统的逐步说明，请参见第 306 页中的 “挂载和取消挂载 Oracle Solaris 文件系统”。

已挂载文件系统表

每次挂载或取消挂载文件系统时，都会使用当前挂载的文件系统的列表修改 `/etc/mnttab`（挂载表）文件。可以使用 `cat` 或 `more` 命令显示此文件的内容。但是，不能编辑此文件。以下是 `/etc/mnttab` 文件的示例：

```
$ more /etc/mnttab
rpool/ROOT/zfsBE       /       zfs     dev=40d0002     0
/devices        /devices        devfs   dev=9500000     1307562777
/dev    /dev    dev     dev=9580000     1307562777
ctfs    /system/contract        ctfs    dev=95c0001     1307562777
proc    /proc   proc    dev=9540000     1307562777
mnttab  /etc/mnttab     mntfs   dev=9600001     1307562777
swap    /system/volatile        tmpfs   xattr,dev=9640001       1307562777
objfs   /system/object  objfs   dev=9680001     1307562777
sharefs /etc/dfs/sharetab       sharefs dev=96c0001     1307562777
fd      /dev/fd fd      rw,dev=97c0001  1307562811
swap    /tmp    tmpfs   xattr,dev=9640002       1307562812
rpool/export    /export zfs     rw,devices,setuid,nonbmand,exec,rstchown,
xattr,atime,dev=40d0003         1307562818
rpool/export/home       /export/home    zfs     rw,devices,setuid,nonbmand,
exec,rstchown,xattr,atime,dev=40d0004         1307562818
```

虚拟文件系统表

大多数文件系统在系统引导时通过 SMF 服务自动进行挂载。

可能需要编辑 `/etc/vfstab` 文件来挂载传统文件系统或更改 ZFS 交换卷。有关更改 ZFS 交换卷的信息，请参见第 19 章，配置附加交换空间（任务）。

要添加项来挂载传统文件系统，需要指定如下信息：

- 文件系统所驻留的设备
- 文件系统的挂载点
- 文件系统类型
- 是否希望文件系统在系统引导时自动挂载（通过使用 `mountall` 命令）
- 任何挂载选项

以下为来自运行 ZFS 根文件系统的系统时的 `vfstab` 示例。

```
# cat /etc/vfstab
#device         device          mount           FS      fsck    mount   mount
#to mount       to fsck         point           type    pass    at boot options
#
fd              -               /dev/fd         fd      -       no      -
/proc           -               /proc           proc    -       no      -
/dev/zvol/dsk/rpool/swap -      -               swap    -       no      -
/devices        -               /devices        devfs   -       no      -
sharefs         -               /etc/dfs/sharetabsharefs -      no      -
ctfs            -               /system/contract ctfs   -       no      -
objfs           -               /system/object  objfs   -       no      -
swap            -               /tmp            tmpfs   -       yes     -
```

ZFS 文件系统在系统引导时通过 SMF 服务自动进行挂载。可以使用传统挂载功能从 `vfstab` 挂载 ZFS 文件系统。有关更多信息，请参见《Oracle Solaris 管理：ZFS 文件系统》。

有关每个 `/etc/vfstab` 字段的说明以及如何编辑和使用该文件的信息，请参见 `vfstab(4)`。

NFS 环境

NFS 是一种分布式文件系统服务，可以用于将一个系统（通常为服务器）中的**资源**（文件或目录）与网络中的其他系统共享。例如，您可能希望与其他系统上的用户共享第三方应用程序或源文件。

使用 NFS 时资源的实际物理位置与用户无关。使用 NFS 可以在一个系统的磁盘上放置一个副本并使所有其他系统通过网络对其进行访问，而不是在每个系统上都放置常用文件的副本。使用 NFS 时，远程文件和本地文件实际没有分别。

有关更多信息，请参见《Oracle Solaris 管理：网络服务》中的第 4 章 “管理网络文件系统（概述）”。

如果系统包含要在网络中共享的资源，则该系统将成为 NFS 服务器。服务器保存了当前共享的资源及其访问限制（如读/写访问或只读访问）的列表。

共享某个资源时，即可使该资源供远程系统进行挂载。

可以使用以下方法共享资源：

- 使用 ZFS `share` 属性创建 ZFS 共享。

- 使用 share 命令创建传统共享。

有关 NFS 的完整说明，请参见《Oracle Solaris 管理：网络服务》中的第 4 章 “管理网络文件系统（概述）”。

NFS 版本 4

此 Oracle Solaris 发行版包括 Oracle 实现的 NFS 版本 4 分布式文件访问协议。

NFS 版本 4 将文件访问、文件锁定和挂载协议集成到一个统一的协议中，从而使穿越防火墙进行遍历更为容易，并提高了安全性。Oracle Solaris 实现的 NFS 版本 4 与 Kerberos V5 完全集成（也称作 SEAM），因此可以提供验证、完整性和保密性。使用 NFS 版本 4，还可在客户机和服务器之间协商使用各自适用的安全风格。使用 NFS 版本 4，服务器可以为不同的文件系统提供不同的安全风格。

有关 NFS 版本 4 功能的更多信息，请参见《Oracle Solaris 管理：网络服务》中的“NFS 服务的新增功能”。

自动挂载 (autofs)

可以使用称为**自动挂载**（即 *autofs*）的客户端服务来挂载 NFS 文件系统资源。每次访问 NFS 资源时，autofs 服务都允许系统对其进行自动挂载和取消挂载。只要仍在目录中并且使用该目录中的文件，资源就会保持挂载状态。如果在某个时间段内未访问资源，则会自动取消挂载该资源。

autofs 服务提供了以下功能：

- 系统引导时无需挂载 NFS 资源，从而节省引导时间。
- 用户无需知道用于挂载和取消挂载 NFS 资源的超级用户口令。
- 可能会减少网络通信流量，因为仅在使用 NFS 资源时才会将其挂载。

autofs 服务通过 automount 实用程序进行初始化，该实用程序在引导系统时自动运行。automountd 守护进程会持续不断地运行，并且负责按需挂载和取消挂载 NFS 文件系统。缺省情况下，/home 文件系统由 automount 守护进程挂载。

使用 autofs，可以指定多台服务器提供同一文件系统。这样，如果其中一台服务器关闭，则 autofs 可以尝试从其他计算机挂载该文件系统。

有关如何设置和管理 autofs 的完整信息，请参见《Oracle Solaris 管理：网络服务》中的第 5 章 “网络文件系统管理（任务）”。

Oracle Solaris SMB 服务

Oracle Solaris OS 提供了 Server Message Block (SMB) 协议服务器和客户端实现，包括对多种 SMB 变种（含 NT LM 0.12 以及通用网络文件系统 (Common Internet File System, CIFS)）提供的支持。术语 CIFS 和 SMB 可以交替使用。

Solaris SMB 服务器允许本地 Oracle Solaris 系统提供文件作为 SMB **共享**到 SMB 启用的客户机（该客户机挂载文件系统共享）。Windows、Mac OS 或 Solaris 客户机可以集成 Solaris SMB 服务器，该服务器将适合 Windows 服务器。Solaris SMB 服务器可以在工作组模式或域模式下工作。在工作组模式下，Solaris SMB 服务器在请求访问共享资源时负责对用户进行本地验证。该验证进程机也称为本地登录。在域模式下，Solaris SMB 服务器使用传递验证，此时用户验证被指定给域控制器。

有关更多信息，请参见《Oracle Solaris Administration: SMB and Windows Interoperability》。

确定文件系统的类型

可以使用以下方法之一确定文件系统的类型：

- 虚拟文件系统标（`/etc/vfstab` 文件）中的 `FS type` 字段
- 本地文件系统的 `/etc/default/fs` 文件
- NFS 文件系统的 `/etc/dfs/fstypes` 文件

如何确定文件系统的类型

无论是否挂载了文件系统，这些命令都适用。

如果已知磁盘分片的原始设备名称，则可以使用 `fstyp` 命令确定文件系统的类型（如果磁盘分片包含文件系统）。有关更多信息，请参见 `fstyp(1M)`。

示例 17–1 确定文件系统的类型

以下示例使用 `fstyp` 命令确定文件系统类型。

```
# fstyp /dev/rdsk/c0t0d0s0
zfs
```

以下示例使用 `/etc/vfstab` 文件确定 `/legacy` 文件系统的文件系统类型。

```
$ grep /legacy /etc/vfstab
/dev/dsk/c0t3d0s6   /dev/rdsk/c0t3d0s6  /legacy ufs   2       yes     -
```

◆ ◆ ◆ 第 18 章

18 创建和挂载文件系统（任务）

本章介绍如何创建和挂载 ZFS 文件系统、临时 (TMPFS) 文件系统和回送 (LOFS) 文件系统。由于 TMPFS 和 LOFS 是虚拟文件系统，因此您实际上是通过挂载这两种文件系统来对其进行“访问”的。此外，本章也介绍了传统 UFS 文件系统的创建和挂载。

以下是本章中逐步说明的列表。

- 第 309 页中的 “如何创建 ZFS 文件系统”
- 第 310 页中的 “如何创建和挂载传统 UFS 文件系统”
- 第 311 页中的 “如何创建和挂载 TMPFS 文件系统”
- 第 312 页中的 “如何创建和挂载 LOFS 文件系统”

创建 Oracle Solaris 文件系统

本节提供有关创建 Oracle Solaris 文件系统的概述。

创建 ZFS 文件系统

ZFS 文件系统并不与特定磁盘分区相关联。ZFS 文件系统包含在 ZFS 存储池中，该存储池可包含许多设备。可将整个磁盘或磁盘分片添加到 ZFS 存储池中。您可能要在池内创建其他文件系统。文件系统提供管理点，允许您管理同一池中不同的数据集。

ZFS 文件系统是使用 `zfs create` 命令创建的。ZFS 文件系统在创建时自动挂载。有关更多信息，请参见第 309 页中的 “如何创建 ZFS 文件系统”。

有关创建 ZFS 存储池和文件系统的更多信息，请参见《Oracle Solaris 管理：ZFS 文件系统》。

创建临时文件系统

临时文件系统 (*temporary file system, TMPFS*) 使用本地内存进行文件系统的读取和写入。TMPFS 文件系统通过节省利用本地磁盘或网络读取和写入临时文件的成本，从而可以提高系统性能。重新引导或取消挂载期间，TMPFS 文件系统中的文件无效。

如果创建多个 TMPFS 文件系统，请注意这些文件系统都使用相同的系统资源。除非使用 `mount` 命令的 `-o size` 选项限制 TMPFS 的大小，否则在 TMPFS 文件系统下创建的文件将用尽可用于其他任何 TMPFS 文件系统的空间。

有关创建和挂载 TMPFS 文件系统的更多信息，请参见第 311 页中的 “如何创建和挂载 TMPFS 文件系统”。

有关常规信息，请参见 `tmpfs(7FS)`。

创建 LOFS 文件系统

LOFS **文件系统**是一种虚拟文件系统，该文件系统提供现有文件系统的替代路径。将其他文件系统挂载到 LOFS 文件系统上时，不会更改原始文件系统。

有关创建和挂载 LOFS 文件系统的更多信息，请参见第 312 页中的 “如何创建和挂载 LOFS 文件系统”。

有关常规信息，请参见 `lofs(7FS)`。

注 – 创建 LOFS 文件系统时请务必谨慎。由于 LOFS 文件系统是虚拟文件系统，因此极有可能会混淆用户和应用程序。

挂载和取消挂载 Oracle Solaris 文件系统

ZFS 文件系统可自动进行挂载和取消挂载。通过挂载传统 UFS 文件系统可使该文件系统可用，这会将文件系统附加到指定挂载点的系统目录树。始终会挂载根目录 (/) 文件系统。

下表提供了有关根据您对文件系统的使用方法挂载文件系统的指南。

所需挂载类型	建议挂载方法
无需频繁挂载的本地或远程文件系统。	从命令行手动键入的 `mount` 命令。
需要频繁挂载的本地传统 UFS 文件系统。本地 ZFS 文件系统通过 SMF 服务自动挂载。	`/etc/vfstab` 文件，在多用户状态下引导系统时自动挂载文件系统。

所需挂载类型	建议挂载方法
需要频繁挂载的远程传统 UFS 文件系统，如起始目录。	■ /etc/vfstab 文件，在多用户状态下引导系统时自动挂载文件系统。 ■ autofs，当您访问文件系统时自动挂载文件系统，或当您更改到另一目录时取消挂载文件系统。

有关挂载可移除介质的更多信息，请参见第 1 章，管理可移除介质（概述）。

可以使用 mount 命令来确定哪些是已经挂载的文件系统。

```
$ mount [ -v ]
```

-v 以详细模式显示已挂载文件系统的列表。

示例 18–1 确定哪些是已挂载文件系统

此示例说明如何使用 mount 命令显示有关当前已挂载文件系统的信息。

```
$ mount
/ on rpool/ROOT/zfsBE read/write/setuid/devices/rstchown/dev=40d0002 on Wed ...
/devices on /devices read/write/setuid/devices/rstchown/dev=9500000 on Wed ...
/dev on /dev read/write/setuid/devices/rstchown/dev=9580000 on Wed Jun ...
/system/contract on ctfs read/write/setuid/devices/rstchown/dev=95c0001 ...
/proc on proc read/write/setuid/devices/rstchown/dev=9540000 on Wed Jun  ...
/etc/mnttab on mnttab read/write/setuid/devices/rstchown/dev=9600001 on Wed ...
/system/volatile on swap read/write/setuid/devices/rstchown/xattr/dev=9640001 ...
/system/object on objfs read/write/setuid/devices/rstchown/dev=9680001 on Wed ...
/etc/dfs/sharetab on sharefs read/write/setuid/devices/rstchown/dev=96c0001 on ...
/dev/fd on fd read/write/setuid/devices/rstchown/dev=97c0001 on Wed Jun  8 ...
/tmp on swap read/write/setuid/devices/rstchown/xattr/dev=9640002 on Wed Jun  ...
/export on rpool/export read/write/setuid/devices/rstchown/nonbmand/exec/xattr/
/export/home on rpool/export/home read/write/setuid/devices/rstchown
/rpool on rpool read/write/setuid/devices/rstchown/nonbmand/exec/
/home/rimmer on pluto:/export/home/rimmer remote/read/write/setuid/xattr/...
```

此示例说明如何使用 zfs mount 命令显示有关当前已挂载的 ZFS 文件系统的信息。

```
$ zfs mount
rpool/ROOT/solaris              /
rpool/ROOT/solaris/var          /var
rpool                           /rpool
rpool/export                    /rpool/export
rpool/export/home               /rpool/export/home
rpool/export/home/admin         /rpool/export/home/admin
```

/etc/vfstab 文件的字段说明

/etc/vfstab 文件中的项包含七个字段，下表为这些字段的说明。

表 18–1 /etc/vfstab 文件的字段说明

字段名称	说明
device to mount	此字段确定了以下内容之一： ■ 本地传统 UFS 文件系统的块设备名称（例如 /dev/dsk/c8t1d0s7）。 ■ 远程文件系统的资源名称（例如 myserver:/export/home）。有关 NFS 的更多信息，请参见《Oracle Solaris 管理：IP 服务》。 ■ 其上要进行交换的分片的块设备名称（例如 /dev/dsk/c0t3d0s1）。 ■ 虚拟文件系统的目录。
device to fsck	与 device to mount 字段所确定的传统 UFS 文件系统对应的原始（字符）设备名称（例如 /dev/rdsk/c8t1d0s7）。此字段确定了 fsck 命令使用的原始接口。若诸如只读文件系统或远程文件系统之类的文件系统没有适用的设备，请使用破折号 (-)。
mount point	确定文件系统的挂载位置（例如 /usr）。
FS type	确定文件系统的类型。
fsck pass	fsck 命令用于确定是否检查传统 UFS 文件系统的传送号。如果此字段包含破折号 (-)，则不会检查文件系统。目前，在引导过程中，/etc/vfstab 文件中的 fsck pass 值会被忽略。 如果此字段包含零，则不会检查传统 UFS 文件系统。如果此字段包含大于零的值，则始终会检查 UFS 文件系统。 对于此字段中值为 1 的所有传统 UFS 文件系统，将按它们在 vfstab 文件中出现的顺序进行检查（每次检查一个）。如果在 fsck pass 值大于 1 的多个 UFS 文件系统上运行 fsck 命令并且使用了整理选项 (-o p)，则 fsck 命令自动并行检查不同磁盘上的文件系统，以最大程度地提高效率。否则，传送号的值不起任何作用。
mount at boot	设置为 yes 或 no，以指示在引导系统时 mountall 命令是否应该自动挂载文件系统。请注意，此字段与 autofs 无关。根目录 (/)、/usr 和 /var 文件系统最初不是从 vfstab 文件挂载的。对于这些文件系统和诸如 /proc 和 /dev/fd 之类的虚拟文件系统，应始终将此字段设置为 no。
mount options	用于挂载文件系统的一系列以逗号分隔的选项（中间无空格）。使用破折号 (-) 表示没有选项。有关更多信息，请参见 vfstab(4)。

注 – 您必须在 /etc/vfstab 文件的每个字段中都输入一个值。如果某个字段没有值，请务必指定一个破折号 (-)。否则，系统可能无法成功引导。同样，字段值也不能为空白空间。

取消挂载 Oracle Solaris 文件系统的先决条件

取消挂载文件系统的先决条件包括：

- 您必须是管理员或担任同等角色。
- 您不能取消挂载正忙的文件系统。如果出现以下任一情况，则认为文件系统正忙：用户正在访问文件系统中的目录；程序打开了文件系统中的文件；正在共享文件系统。通过执行以下操作，可以使文件系统处于可取消挂载状态：
  - 更改到另一不同文件系统的目录。
  - 注销该系统。
  - 使用 `fuser` 命令列出正在访问该文件系统的所有进程，然后停止这些进程（如有必要）。有关更多详细信息，请参见第 316 页中的 “如何停止所有进程访问文件系统”。

    如果您需要取消挂载用户正在使用的文件系统，请通知他们。
  - 取消共享该文件系统。有关取消共享文件系统的信息，请参见 `unshare(1M)`。

要验证是否已取消挂载一个或多个文件系统，请检查 `mount` 命令的输出。

```
$ mount | grep unmounted-file-system
```

创建和挂载 Oracle Solaris 文件系统

本节提供有关创建和挂载 Oracle Solaris 文件系统的示例。

▼ 如何创建 ZFS 文件系统

1. **成为管理员。**

   有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。

2. **创建 ZFS 存储池。**

   以下示例说明如何在一个命令中创建名为 `tank` 的简单镜像存储池和名为 `tank` 的 ZFS 文件系统。假定磁盘 `/dev/dsk/c1t0d0` 和 `/dev/dsk/c2t0d0` 全部都可使用。

   ```
   # zpool create tank mirror c1t0d0 c2t0d0
   ```

3. **创建 ZFS 文件系统。**

   ```
   # zfs create tank/fs
   ```

   新 ZFS 文件系统 `tank/fs` 可根据需要尽可能使用磁盘空间，并在 `/tank/fs` 处自动挂载。

4 **确认是否创建文件系统。**

```
# zfs list -r tank
NAME      USED  AVAIL  REFER  MOUNTPOINT
tank      117K   268G    21K  /tank
tank/fs    21K   268G    21K  /tank/fs
```

▼ 如何创建和挂载传统 UFS 文件系统

开始之前 请确保已满足以下先决条件：

- 磁盘必须已格式化并分片。
- 如果要重建现有的传统 UFS 文件系统，请取消挂载该文件系统。
- 需要知道将包含文件系统的分片的设备名称。

有关查找磁盘和磁盘分片编号的信息，请参见第 11 章，管理磁盘（任务）。

有关格式化磁盘和将磁盘分片的信息，请参见第 10 章，管理磁盘（概述）。

1 **成为管理员。**

有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。

2 **创建传统 UFS 文件系统。**

 # newfs [**-N**] [**-b** *size*] [**-i** *bytes*] **/dev/rdsk/***device-name*

系统会要求确认。

注意 – 执行此步骤之前，请确保为分片指定了正确的设备名称。如果指定错误的分片，则在创建新文件系统时将删除其内容。此错误可能会导致系统出现紧急情况。

3 **要验证是否已创建传统 UFS 文件系统，请检查新文件系统。**

 # fsck /dev/rdsk/*device-name*

其中 *device-name* 参数指定包含新文件系统的磁盘设备的名称。

`fsck` 命令检查新文件系统的一致性、报告所有问题并在修复问题之前进行提示。有关 `fsck` 命令的更多信息，请参见 `fsck(1M)`。

4 **挂载传统 UFS 文件系统。**

```
# mkdir /directory-name
# mount /dev/dsk/device-name  /directory-name
```

示例 18–2 创建和挂载传统 UFS 文件系统

以下示例说明如何在 `/legacy` 上创建和挂载 UFS 文件系统 `/dev/rdsk/c0t1d0s0`。

```
# newfs /dev/rdsk/c0t1d0s0
newfs: construct a new file system /dev/rdsk/c0t1d0s0: (y/n)? y
/dev/rdsk/c0t1d0s0:     286722656 sectors in 46668 cylinders of 48 tracks, 128 sectors
        140001.3MB in 2917 cyl groups (16 c/g, 48.00MB/g, 5824 i/g)
super-block backups (for fsck -F ufs -o b=#) at:
 32, 98464, 196896, 295328, 393760, 492192, 590624, 689056, 787488, 885920,
Initializing cylinder groups:
.....................................................
super-block backups for last 10 cylinder groups at:
 285773216, 285871648, 285970080, 286068512, 286166944, 286265376, 286363808,
 286462240, 286560672, 286659104
# fsck /dev/rdsk/c0t1d0s0
# mkdir /legacy
# mount /dev/dsk/c0t1d0s0 /legacy
```

更多信息 **创建传统 UFS 文件系统后……**

要在引导时自动挂载传统 UFS 文件系统，请转到第 313 页中的“如何向 /etc/vfstab 文件添加项”。

▼ 如何创建和挂载 TMPFS 文件系统

1. **成为管理员。**

   有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。

2. **如有必要，请创建要作为 TMPFS 文件系统挂载的目录。**

   ```
   # mkdir /mount-point
   ```

   其中 *mount-point* 是 TMPFS 文件系统的挂载目录。

3. **挂载 TMPFS 文件系统。**

   ```
   # mount -F tmpfs [-o size=number]  swap mount-point
   ```

   `-o size=`*number*　指定 TMPFS 文件系统的大小限制（以 MB 为单位）。

   mount-point　指定 TMPFS 文件系统的挂载目录。

   要将系统设置为在引导时自动挂载 TMPFS 文件系统，请参见示例 18–4。

4. **验证是否已创建 TMPFS 文件系统。**

   ```
   # mount -v
   ```

示例 18–3 **创建和挂载 TMPFS 文件系统**

以下示例说明如何创建、挂载 TMPFS 文件系统 `/export/reports` 并将其大小限制为 50 MB。

```
# mkdir /export/reports
# chmod 777 /export/reports
# mount -F tmpfs -o size=50m swap /export/reports
# mount -v
```

示例 18–4 在引导时挂载 TMPFS 文件系统

通过添加 /etc/vfstab 项，可以将系统设置为在引导时自动挂载 TMPFS 文件系统。以下示例说明了 /etc/vfstab 文件中的某一项，该项在引导时将 /export/test 作为 TMPFS 文件系统进行挂载。由于未指定 size=*number* 选项，因此 /export/test 上的 TMPFS 文件系统的大小仅受可用系统资源的限制。

```
swap - /export/test  tmpfs   -  yes  -
```

▼ 如何创建和挂载 LOFS 文件系统

1. **成为管理员。**

   有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。

2. **如有必要，请创建要作为 LOFS 文件系统挂载的目录。**

   # mkdir *loopback-directory*

3. **授予新创建的目录适当的权限和所有权。**

4. **如有必要，请创建要在其中挂载 LOFS 文件系统的挂载点。**

   # mkdir */mount-point*

5. **挂载 LOFS 文件系统。**

   # mount -F lofs *loopback-directory /mount-point*

   loopback-directory 指定要在回送挂载点上挂载的文件系统。

   /mount-point 指定 LOFS 文件系统的挂载目录。

6. **验证是否已挂载 LOFS 文件系统。**

   # mount -v

示例 18–5 创建和挂载 LOFS 文件系统

以下示例说明如何创建、挂载并作为回送文件系统测试 /new/dist 目录中的新软件，而无需实际安装该软件。

```
# mkdir /tmp/newroot
# mount -F lofs /new/dist /tmp/newroot
# chroot /tmp/newroot newcommand
```

示例 18–6 在引导时挂载 LOFS 文件系统

通过向 /etc/vfstab 文件的结尾添加一项，可以将系统设置为在引导时自动挂载 LOFS 文件系统。以下示例说明了 /etc/vfstab 文件中的某一项，该项为根目录 (/) 文件系统在 /tmp/newroot 上挂载 LOFS 文件系统。

```
/ - /tmp/newroot  lofs   -  yes  -
```

请确保回送项是 /etc/vfstab 文件中的最后几项。否则，如果回送文件系统的 /etc/vfstab 项先于其中要包括的文件系统，则无法挂载回送文件系统。

▼ 如何向 /etc/vfstab 文件添加项

除非某些 ZFS 文件系统需要先前挂载的行为，否则请使用此过程以在引导时挂载非 ZFS 文件系统。有关挂载 ZFS 文件系统的更多信息，请参见《Oracle Solaris 管理：ZFS 文件系统》。

1. **成为管理员。**

   有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。

2. **如有必要，请为要挂载的文件系统创建挂载点。**

   # mkdir */mount-point*

   本地系统上必须存在用于挂载文件系统的挂载点。**挂载点**是指附加已挂载文件系统的目录。

3. **编辑 /etc/vfstab 文件并添加项。请确保执行以下操作：**

   a. **使用空白空间（空格或制表符）分隔各个字段。**

   b. **如果字段没有内容，请指定破折号 (-)。**

   c. **保存更改。**

注 – 由于在引导过程中内核会将根目录 (/) 文件系统挂载为只读，因此只有 remount 选项（以及可以与 remount 一起使用的选项）会影响 /etc/vfstab 文件中的根目录 (/) 项。

示例 18–7 向 /etc/vfstab 文件添加项

以下示例说明如何将磁盘分片 /dev/dsk/c0t3d0s7 作为传统 UFS 文件系统挂载到挂载点 /files1。将原始字符设备 /dev/rdsk/c0t3d0s7 指定为 device to fsck。fsck pass 值为 2 表明将检查文件系统，但不是按顺序检查。

```
#device           device             mount    FS       fsck   mount    mount
#to mount         to fsck            point    type     pass   at boot  options
#
/dev/dsk/c0t3d0s7 /dev/rdsk/c0t3d0s7 /files1  ufs      2      yes       -
```

以下示例说明如何将 /export/man 目录作为 NFS 文件系统从系统 pluto 挂载到挂载点 /usr/man 上。由于该目录是 NFS 文件系统，因此既不会指定 device to fsck，也不会指定 fsck pass。在本示例中，mount options 为 ro（只读）和 soft。

```
#device           device             mount    FS       fsck   mount    mount
#to mount         to fsck            point    type     pass   at boot  options
pluto:/export/man   -                /usr/man nfs      -      yes       ro,soft
```

▼ 如何挂载文件系统（/etc/vfstab 文件）

1. **成为管理员。**

   有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。

2. **挂载 /etc/vfstab 文件中列出的文件系统。**

   # mount */mount-point*

   其中 */mount-point* 指定了 /etc/vfstab 文件的 mount point 或 device to mount 字段中的项。通常，指定挂载点较为容易。

示例 18–8 挂载文件系统（/etc/vfstab 文件）

以下示例说明如何挂载 /etc/vfstab 文件中列出的 /legacy 文件系统。

```
# mount /legacy
```

示例 18–9 挂载所有文件系统（/etc/vfstab 文件）

以下示例说明在已挂载文件系统的情况下使用 mountall 命令时显示的消息。

```
# mountall
mount: /tmp is already mounted or swap is busy
```

以下示例说明如何挂载 /etc/vfstab 文件中列出的所有本地系统。

```
# mountall -l
```

以下示例说明如何挂载所有可用的 ZFS 文件系统。

```
# zfs mount -a
```

以下示例说明如何挂载 /etc/vfstab 文件中列出的所有远程文件系统。

```
# mountall -r
```

▼ 如何挂载 NFS 文件系统（ mount 命令 ）

1 **成为管理员。**

有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。

2 **如有必要，请为要挂载的文件系统创建挂载点。**

```
# mkdir /mount-point
```

本地系统上必须存在用于挂载文件系统的挂载点。挂载点是指附加已挂载文件系统的目录。

3 **请确保服务器中的资源（ 文件或目录 ）是可用的。**

要挂载 NFS 文件系统，必须使用 share 命令使服务器上的资源可用。有关如何共享资源的信息，请参见《Oracle Solaris 管理：网络服务》中的“关于 NFS 服务”。

4 **挂载 NFS 文件系统。**

```
# mount -F nfs [-o mount-options] server:/directory /mount-point
```

示例 18–10 挂载 NFS 文件系统（ mount 命令 ）

以下示例说明如何将 /export/packages 目录从服务器 pluto 挂载到 /mnt 上。

```
# mount -F nfs pluto:/export/packages /mnt
```

▼ x86: 如何从硬盘挂载 PCFS (DOS) 文件系统（ mount 命令 ）

使用以下过程从硬盘挂载 PCFS (DOS) 文件系统。

1 **成为管理员。**

有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。

2 **如有必要，请为要挂载的文件系统创建挂载点。**

```
# mkdir /mount-point
```

本地系统上必须存在用于挂载文件系统的挂载点。**挂载点**是指附加已挂载文件系统的目录。

3 **挂载 PCFS 文件系统。**

```
# mount -F pcfs [-o rw | ro] /dev/dsk/device-name:logical-drive /mount-point
```

`-o rw | ro`	指定可以将 PCFS 文件系统挂载为读取/写入 (`rw`) 或只读 (`ro`)。如果不指定此选项，则缺省情况下挂载为 `rw`。
`/dev/dsk/`*device-name*	指定整个磁盘的设备名称（例如 `/dev/dsk/c0t0d0p0`）。
logical-drive	指定 DOS 逻辑驱动器盘符（c 到 z）或驱动器编号（1 到 24）。驱动器 c 相当于驱动器 1，表示驱动器上的主 DOS 分片。所有其他字母或数字均表示扩展的 DOS 分片内的 DOS 逻辑驱动器。
/mount-point	指定文件系统的挂载目录。

请注意，必须使用冒号分隔 *device-name* 和 *logical-drive*。

示例 18–11 x86: 从硬盘挂载 PCFS (DOS) 文件系统（ mount 命令 ）

以下示例说明如何将主 DOS 分片中的逻辑驱动器挂载到 `/pcfs/c` 目录上。

```
# mount -F pcfs /dev/dsk/c0t0d0p0:c /pcfs/c
```

以下示例说明如何将扩展的 DOS 分片中的第一个逻辑驱动器在 `/mnt` 目录上挂载为只读。

```
# mount -F pcfs -o ro /dev/dsk/c0t0d0p0:2 /mnt
```

▼ 如何停止所有进程访问文件系统

1 **成为管理员。**

有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。

2 **列出正在访问文件系统的所有进程，以便确定要停止的进程。**

```
# fuser -c [ -u ] /mount-point
```

`-c`	报告是文件系统挂载点的文件以及已挂载文件系统内的所有文件。
`-u`	显示每个进程 ID 的用户登录名称。
/mount-point	指定要停止进程的文件系统的名称。

3 **停止正在访问文件系统的所有进程。**

```
# fuser -c -k /mount-point
```

SIGKILL 会被发送到正在使用文件系统的每个进程。

注 – 不得在未事先警告用户的情况下停止用户进程。

---

4 **验证是否所有进程都未在访问文件系统。**

```
# fuser -c /mount-point
```

示例 18–12 停止所有进程访问文件系统

以下示例说明如何停止正在使用 /export/home 文件系统的进程 4006c。

```
# fuser -c /export/home
/export/home:     4006c
# fuser -k -c /export/home
/export/home:     4006c
# fuser -c /export/home
/export/home:
```

▼ 如何取消挂载文件系统

使用以下过程取消挂载文件系统。

1 **请确保已满足第 309 页中的“取消挂载 Oracle Solaris 文件系统的先决条件”中列出的先决条件。**

2 **取消挂载文件系统。**

```
# umount /mount-point
```

其中 */mount-point* 是指要取消挂载的文件系统的名称。该名称可以是以下各项之一：

- 文件系统的挂载目录名称
- 文件系统的设备名称路径
- NFS 文件系统的资源
- LOFS 文件系统的回送目录

示例 18–13 取消挂载文件系统

以下示例说明如何取消挂载传统 UFS 文件系统。

```
# umount /legacy
```

以下示例说明如何强制取消挂载 UFS /legacy 文件系统。

```
# umount -f /legacy
#
```

以下示例说明如何取消挂载所有 ZFS 文件系统。

```
# zfs umount -a
```

除正忙的文件系统以外，其他所有文件系统都被取消挂载。

◆ ◆ ◆ 第 19 章

19 配置附加交换空间（任务）

本章提供了安装 Oracle Solaris OS 后配置 ZFS 根文件系统的附加交换空间的指导和逐步说明。

以下是本章中主题的列表。

- 第 319 页中的 “关于交换空间”
- 第 322 页中的 “如何了解是否需要更多交换空间？ ”
- 第 323 页中的 “如何分配交换空间”
- 第 323 页中的 “规划交换空间”
- 第 324 页中的 “监视交换资源”
- 第 326 页中的 “在 Oracle Solaris ZFS 根环境中添加或更改交换空间”

关于交换空间

要确定以下内容，应了解 SunOS 交换机制的功能：

- 交换空间要求
- 交换空间与 TMPFS 文件系统之间的关系
- 如何从与交换空间有关的错误消息恢复

交换空间与虚拟内存

Oracle Solaris OS 软件和应用程序软件可以将一些磁盘空间用于临时存储而不是用于文件系统。磁盘的保留区称为**交换**空间。当系统没有足够的物理内存来处理当前进程时，将交换空间用作虚拟内存存储区域。在 ZFS 根文件系统中，为交换所保留的磁盘空间为 ZFS 卷。

虚拟内存系统将磁盘上文件的物理副本映射到内存中的虚拟地址。文件系统中的常规文件或交换空间可以备份包含这些映射的数据的物理内存页面。如果交换空间对内存进行了备份，则该内存称为**匿名内存**，因为没有为备份内存的磁盘空间指定标识。

Oracle Solaris OS 使用的**虚拟交换空间**概念是指，匿名内存页面和实际备份这些页面的物理存储（或磁盘备份交换空间）之间的层。系统的虚拟交换空间等于其所有物理（磁盘备份）交换空间与当前部分可用物理内存的和。

虚拟交换空间具有以下优点：

- 由于虚拟交换空间不必对应于物理（磁盘）存储，因此减少了对大量物理交换空间的需求。
- 称为 SWAPFS 的伪文件系统可为匿名内存页面提供地址。由于 SWAPFS 可以控制内存页面的分配，因此它在确定要对页面执行的操作方面具有更大的灵活性。例如，SWAPFS 可更改页面对磁盘备份交换存储的要求。

交换空间与 TMPFS 文件系统

在 Oracle Solaris 环境中，TMPFS 文件系统由 `/etc/vfstab` 文件中的项自动激活。TMPFS 文件系统将文件及其关联信息存储在内存中（存储在 `/tmp` 目录中）而不磁盘上，从而加快了访问这些文件的速度。对于大量使用 `/tmp` 的应用程序（例如编译器和 DBMS 产品），此功能可显著增强这些应用程序的性能。

TMPFS 文件系统从系统的交换资源分配 `/tmp` 目录中的空间。此功能意味着，用尽 `/tmp` 目录中的空间时，也就用尽了交换空间。因此，如果应用程序大量使用 `/tmp` 目录，而且您未监视交换空间使用情况，则系统可能会用尽交换空间。

如果要使用 TMPFS，但交换资源有限，请采用以下措施：

- 使用尺寸选项 (`-o` *size*) 挂载 TMPFS 文件系统，以便控制 TMPFS 可以使用的交换资源量。
- 使用编译器的 `TMPDIR` 环境变量指向另一个较大的目录。

  使用编译器的 `TMPDIR` 变量仅可以控制编译器是否使用 `/tmp` 目录。此变量对其他程序是否使用 `/tmp` 目录没有影响。

交换空间与转储设备配置

转储设备通常是为存储系统故障转储信息而保留的磁盘空间。安装系统时，会自动创建一个 ZFS 交换卷和转储卷。可以使用 `dumpadm` 命令更改系统的转储卷。有关更多信息，请参见《Oracle Solaris 管理：常见任务》中的第 17 章 “管理系统故障转储信息（任务）”。

在 ZFS 根环境中，交换和转储都可配置为独立的 ZFS 卷。此模型具有如下优点：

- 不必将磁盘分区以包括交换和转储区域。
- 交换和转储设备可受益于底层 ZFS I/O 流水线体系结构。
- 可在交换和转储设备上设置特征，例如压缩。

- 可重置交换和转储设备大小。例如：

```
# zfs set volsize=2G rpool/dump
# zfs get volsize rpool/dump
NAME        PROPERTY  VALUE       SOURCE
rpool/dump  volsize   2G          -
```

  请记住，重新分配一个大转储设备的进程非常耗时。
- 在活动系统上，可以考虑添加第二个交换卷，以增加整体交换大小。

有关使用 ZFS 交换和转储设备的更多信息，请参见《Oracle Solaris 管理：ZFS 文件系统》中的“管理 ZFS 交换和转储设备”。

交换空间和动态重新配置

建议分配足够的交换空间，以便在动态重新配置期间支持出现故障的 CPU 或系统板。否则，CPU 或系统板故障可能会导致主机或域重新引导时内存不足。

如果此交换空间不可用，则一个或多个应用程序可能会因内存不足而无法启动。此问题将需要进行手动干预，以便添加附加交换空间或重新配置这些应用程序的内存使用情况。

如果已分配附加交换空间来处理重新引导时内存的潜在不足，则所有需要大量内存的应用程序可能会照常启动。这意味着，用户可以使用系统，但系统速度可能会因一些其他交换而变慢。

有关更多信息，请参见硬件动态重新配置指南。

在 SAN 环境中配置交换空间

查看以下几点以确定是否可在与网络连接的磁盘上配置交换空间，例如在 SAN 环境中：

- 诊断本地连接磁盘上的交换空间问题要比诊断与网络连接的磁盘上的交换空间问题更加简便。
- SAN 上的交换空间的性能应与本地连接磁盘上配置的交换空间的性能相当。
- 分析性能数据之后，向具有性能问题的系统添加更多内存可解决 SAN 上交换空间的性能问题，较之于将交换空间移至本地连接磁盘，这样做更好。

如何了解是否需要更多交换空间？

使用 swap -l 命令可确定系统是否需要更多交换空间。

例如，以下 swap -l 输出显示，此系统的交换空间几乎完全占用或者已经 100% 分配。

```
% swap -l
swap -l
swapfile             dev    swaplo   blocks     free
/dev/zvol/dsk/rpool/swap 182,2        16 67108848 67102992
```

当系统的交换空间为 100% 分配时，会暂时锁定应用程序的内存页面。可能不会出现应用程序错误，但可能会降低系统性能。

与交换相关的错误消息

这些消息指示应用程序尝试获取更多的匿名内存。但是，没有剩余的交换空间提供支持。

application is out of memory

malloc error 0

```
messages.1:Sep 21 20:52:11 mars genunix: [ID 470503 kern.warning]
WARNING: Sorry, no swap space to grow stack for pid 100295 (myprog)
```

与 TMPFS 相关的错误消息

如果写入文件时无法分配页面，则会显示以下消息。如果 TMPFS 尝试写入的量超出其所允许的量，或者当前执行的程序正在使用大量内存，则可能会出现此问题。

directory: File system full, swap space limit exceeded

以下消息表明，TMPFS 尝试创建新的文件或目录时物理内存不足：

directory: File system full, memory allocation failed

有关从与 TMPFS 相关的错误消息恢复的信息，请参见 tmpfs(7FS)。

如何分配交换空间

最初，交换空间是在 Oracle Solaris 安装过程中分配的。在 ZFS 根环境中，缺省交换大小范围通常是 512 MB 至 2 GB。

有关分配交换空间的一般规则，请参见第 323 页中的 “规划交换空间”。

交换区域和 /etc/vfstab 文件

安装系统后，交换区域和交换文件将在 `/etc/vfstab` 文件中列出。在引导系统时，它们将由 `/sbin/swapadd` 脚本激活。

`/etc/vfstab` 文件中交换设备对应的项包含以下内容：

- 具有 ZFS 根文件系统的系统上交换卷路径名的全路径名
- 交换分片或交换文件的文件系统类型

激活交换文件之前，必须挂载包含交换文件的文件系统。因此，在 `/etc/vfstab` 文件中，应确保挂载文件系统的对应项位于激活交换文件的对应项之前。

规划交换空间

确定交换空间大小的最重要因素是系统的软件应用程序的要求。例如，大型应用程序（例如计算机辅助设计模拟器、数据库管理产品、事务监视器和地质分析系统）占用的交换空间可高达 200–1000 MB。

请咨询应用程序供应商，以了解其应用程序的交换空间要求。

如果无法通过应用程序供应商确定交换空间要求，请根据系统类型使用以下一般规则来分配交换空间。

注 – 将故障转储内容进行压缩，从而使转储设备不必与物理内存的大小相同。缺省情况下，转储内容值设置为内核页。但是，如果将转储内容值设置为转储所有内存页，则请考虑将转储大小增加至物理内存大小的一半或更多。

表 19–1 ZFS 文件系统的交换和转储卷大小

系统类型	交换卷大小	转储卷大小
物理内存约为 4 GB 的系统	1 GB	2 GB
物理内存约为 8 GB 的中端服务器	2 GB	4 GB
物理内存约为 16 到 128 GB 的高端服务器	4 GB	8-64 GB

表 19–1 ZFS 文件系统的交换和转储卷大小 （续）

系统类型	交换卷大小	转储卷大小
物理内存为 128 GB 以上的高端服务器	1/4 物理内存大小	1/2 物理内存大小

注 – 具有许多活动的 ZFS 文件系统的忙系统可使用 1/2 至 3/4 物理内存大小作为转储设备大小。

在一个具有 ZFS 根文件系统的系统上，如果尝试使用 `dumpadm -d` 命令指定一个太小而不能容纳崩溃系统的转储设备，则将看到类似于以下内容的消息：

```
dumpadm: dump device /dev/zvol/dsk/rpool/dump is too small to hold a
system dump dump size 43467329536 bytes, device size 42949672960 bytes
```

为基于 ZFS 的系统分配交换空间

初次安装 ZFS 根文件系统期间，将在 ZFS 根池中的 ZFS 卷上自动创建交换区域。

在 ZFS 根池中，不会将交换设备预分配至大小固定的分片，因此，随后可以很容易地修改交换大小。

在评估应用程序的交换要求后，可在初次安装期间或在安装后使用缺省交换大小或调整交换卷大小（如有必要）。

在初次安装期间，内核根据 `dumpadm` 信息和物理内存大小计算缺省转储卷大小。

在 ZFS 环境中，文件系统占用池的空间，因此，`/var/crash` 目录根据保存的故障转储量来占用它所需要的空间。

监视交换资源

`/usr/sbin/swap` 命令用于管理交换区域。`-l` 和 `-s` 这两个选项显示有关交换资源的信息。

使用 `swap -l` 命令可识别系统的交换区域。在 `swapfile` 列中列出了已激活的交换设备或文件。例如：

```
# swap -l
swapfile             dev  swaplo blocks   free
/dev/dsk/c0t0d0s1   136,1      16 1638608 1600528
```

在具有 ZFS 根文件系统的系统上，`swap -l` 命令可确定类似的输出，除非它可确定 ZFS 卷路径名。例如：

```
# swap -l
swapfile                   dev  swaplo blocks   free
/dev/zvol/dsk/rpool/swap 256,1      16 1058800 1058800
```

使用 swap -s 命令可监视交换资源。

```
# swap -s
total: 57416k bytes allocated + 10480k reserved = 67896k used,
833128k available
```

used 值与 available 值之和等于系统上的总交换空间，其中包括部分物理内存和交换设备（或文件）。

可以使用可用交换空间量和已用交换空间量（在 swap -s 输出中）监视在一段时间内交换空间的使用情况。如果系统性能良好，请使用 swap -s 确定可用交换空间量。如果系统性能下降，请检查可用交换空间量以便确定其是否已下降。然后，可以确定对系统所做的哪些更改可能导致交换空间使用量增加。

使用此命令时，请记住，可用于交换的物理内存量会因内核和用户进程的锁定和物理内存的释放而动态变化。

注 – swap -l 命令以 512 字节块为单位显示交换空间。swap -s 命令以 1024 字节块为单位显示交换空间。如果将 swap -l 中的块累加，然后将其转换为 KB，则结果将小于 used + available（在 swap -s 输出中）。原因是 swap -l 在计算交换空间时不包括物理内存。

下表概述了 swap -s 命令的输出。

表 19–2 swap -s 命令的输出

关键字	说明
bytes allocated	当前已分配作为后备存储（磁盘备份交换空间）的总交换空间量（以 1024 字节块为单位）。
reserved	当前未分配、但内存已请求供将来使用的总交换空间量（以 1024 字节块为单位）。
used	已分配或保留的总交换空间量（以 1024 字节块为单位）。
available	当前可用于将来进行预留和分配的总交换空间量（以 1024 字节块为单位）。

在 Oracle Solaris ZFS 根环境中添加或更改交换空间

下一节介绍了如何在 ZFS 根环境中添加或更改交换空间。要确定系统或应用程序是否需要附加交换空间，请参见上述各节。

▼ 如何在 Oracle Solaris ZFS 根环境中添加交换空间

1 **成为管理员。**

有关更多信息，请参见《Oracle Solaris 管理：安全服务》中的“如何获取管理权限”。

2 **确定当前交换卷。**

无法移除使用中的交换卷。可通过比较 blocks 列中列出的块数和 free 列中列出的块数来判断当前交换卷是否正在使用中。如果两列中的块数相等，则说明交换区域不忙。例如：

```
# swap -l
swapfile                 dev  swaplo   blocks   free
/dev/zvol/dsk/rpool/swap 256,1      16 1058800 1058800
```

3 **选择以下操作之一调整交换卷的大小。**

a. **如果当前交换区域不在使用中，则可以调整当前交换卷的大小。**

例如：

```
# zfs get volsize rpool/swap
NAME        PROPERTY  VALUE    SOURCE
rpool/swap  volsize   517M     -
# zfs set volsize=2g rpool/swap
# zfs get volsize rpool/swap
NAME        PROPERTY  VALUE    SOURCE
rpool/swap  volsize   2G       -
```

b. **如果当前交换区域正在使用中，则可以添加另一个交换卷。**

例如：

```
# zfs create -V 2G rpool/swap2
```

c. **激活第二个交换卷。**

例如：

```
# swap -a /dev/zvol/dsk/rpool/swap2
# swap -l
swapfile                  dev  swaplo   blocks   free
/dev/zvol/dsk/rpool/swap  256,1      16 1058800 1058800
/dev/zvol/dsk/rpool/swap2 256,3      16 4194288 4194288
```

◆ ◆ ◆ 第 20 章

20 复制文件和文件系统（任务）

本章介绍如何使用各种备份命令将文件和文件系统复制到磁盘、磁带和软盘。

以下是本章中逐步说明的列表。

- 第 329 页中的 “如何在文件系统之间复制目录 (cpio)”
- 第 330 页中的 “如何将文件复制到磁带 (tar)”
- 第 331 页中的 “如何列出磁带中的文件 (tar)”
- 第 332 页中的 “如何从磁带中检索文件 (tar)”
- 第 333 页中的 “使用 pax 命令将文件复制到磁带”
- 第 333 页中的 “如何将目录中的所有文件复制到磁带 (cpio)”
- 第 335 页中的 “如何列出磁带中的文件 (cpio)”
- 第 335 页中的 “如何从磁带中检索所有文件 (cpio)”
- 第 336 页中的 “如何从磁带中检索特定文件 (cpio)”
- 第 337 页中的 “如何将文件复制到远程磁带设备（tar 和 dd）”
- 第 338 页中的 “如何从远程磁带设备中提取文件”

用于复制文件系统的命令

当需要复制或移动各个文件、文件系统的某些部分或完整的文件系统时，可以使用本章中介绍的过程。

下表介绍 Oracle Solaris 发行版中可用的各种备份和恢复命令。对于企业环境，请考虑使用企业级的备份产品。Oracle 技术网络提供有关企业级备份产品的信息。

表 20–1 各种备份命令汇总

命令名称	是否可识别文件系统边界？	是否支持多卷备份？	物理复制还是逻辑复制？
volcopy	是	是	物理
tar	否	否	逻辑

表 20–1 各种备份命令汇总 （续）

命令名称	是否可识别文件系统边界？	是否支持多卷备份？	物理复制还是逻辑复制？
cpio	否	是	逻辑
pax	是	是	逻辑
dd	是	否	物理
zfs send and zfs receive	是	N/A	逻辑
zfs snapshot	是	N/A	逻辑

下表介绍了其中某些命令的优点和缺点。

表 20–2 tar、pax 和 cpio 命令的优点和缺点

命令	功能	优点	缺点
tar	用于将文件和目录子树复制到单个磁带。	■ 可以在大多数 UNIX 操作系统中使用 ■ 可以轻松访问公共域版本	■ 无法识别文件系统边界 ■ 全路径名的长度不能超过 255 个字符 ■ 不能用于创建多个磁带卷
pax	用于复制需要多个磁带卷的文件、特殊文件或文件系统。或者，当需要向符合 POSIX 的系统复制文件或从其复制文件时使用。	■ 对于符合 POSIX 的系统，其可移植性比 tar 或 cpio 命令高 ■ 多供应商支持	与 tar 命令的缺点相同，不同的是 pax 命令可以创建多个磁带卷。
cpio	用于复制需要多个磁带卷的文件、特殊文件或文件系统。或者，当需要将文件从运行当前 Oracle Solaris 发行版的系统复制到运行早期 Solaris 发行版的系统时使用。	■ 与使用 tar 命令相比，可以更有效地将数据打包到磁带 ■ 恢复时可跳过磁带中的任何坏点 ■ 提供以不同的头格式编写文件的选项（如 tar、ustar、crc、odc、bar），以实现不同系统类型之间的可移植性 ■ 创建多个磁带卷	该命令语法比 tar 或 pax 命令更难。

以下各节介绍有关如何使用这些命令的逐步说明和示例。

在文件系统之间复制目录（cpio 命令）

可以使用 cpio（复制入和复制出）命令复制各个文件、多组文件或完整的文件系统。本节介绍如何使用 cpio 命令复制完整的文件系统

cpio 命令是一个归档程序，可以将文件列表复制到单个大型输出文件中。此命令会在各个文件之间插入头，以便于进行恢复。可以使用 cpio 命令将完整的文件系统复制到其他分片、其他系统或介质设备（如磁带或 USB 磁盘）。

由于 cpio 命令可以识别介质结尾并会提示您插入另一个卷，因此它是最有效的命令，可用于创建需要多盒磁带或多张 USB 磁盘的归档文件。

使用 cpio 命令时，经常将 ls 和 find 命令结合使用以列出并选择要复制的文件，然后将输出传输给 cpio 命令。

▼ 如何在文件系统之间复制目录 (cpio)

1 成为管理员。

2 转到相应的目录。

```
# cd filesystem1
```

3 将 find 和 cpio 命令结合使用，将目录树从 *filesystem1* 复制到 *filesystem2*。

```
# find . -print -depth | cpio -pdm filesystem2
```

.　从当前工作目录开始。

-print　列出文件名。

-depth　向下派生目录分层结构并从下到上列出文件名。

-p　创建文件列表。

-d　根据需要创建目录。

-m　在目录中设置正确的修改时间。

有关更多信息，请参见 cpio(1)。

指定目录名中的文件已复制。符号链接会被保留。

可能还要指定 -u 选项。此选项将强制执行无条件复制。否则，旧文件不会替换新文件。在需要目录的精确副本且复制的某些文件可能已存在于目标目录中时，此选项可能非常有用。

4 通过显示目标目录的内容，验证复制是否成功。

```
# cd filesystem2
# ls
```

5 **删除源目录（如果可行）。**

```
# rm -rf filesystem1
```

示例 20–1 在文件系统之间复制目录 (cpio)

```
# cd /data1
# find . -print -depth | cpio -pdm /data2
19013 blocks
# cd /data2
# ls
# rm -rf /data1
```

将文件和文件系统复制到磁带

可以使用 tar、pax 和 cpio 命令将文件和文件系统复制到磁带。所选的命令取决于复制所需的灵活性和精确度。由于所有这三个命令都使用原始设备，因此在使用磁带之前不需要在其中格式化或创建文件系统。

使用的磁带机及设备名称取决于每个系统的硬件配置。有关磁带设备名称的更多信息，请参见第 341 页中的 “选择要使用的介质”。

将文件复制到磁带（tar 命令）

以下是在使用 tar 命令将文件复制到磁带之前应了解的信息：

- 如果使用带有 -c 选项的 tar 命令将文件复制到磁带，则会损坏磁带中位于当前磁带位置以及该位置后的任何已有文件。
- 可以使用文件名替换通配符（? 和 *）作为复制文件时指定的文件名的一部分。例如，要复制带有 .doc 后缀的所有文档，请键入 *.doc 作为文件名参数。
- 从 tar 归档文件中提取文件时，不能使用文件名称替换通配符。

▼ 如何将文件复制到磁带 (tar)

1 **转到包含要复制的文件的目录。**

2 **将可写磁带插入磁带机中。**

3 **将文件复制到磁带。**

```
$ tar cvf /dev/rmt/n filenames
```

c 指明要创建归档文件。

v	归档时显示每个文件的名称。
f /dev/rmt/*n*	指明应该将归档文件写入指定的设备或文件。
filenames	指明要复制的文件和目录。使用空格分隔多个文件。

将指定的文件名复制到磁带，会覆盖该磁带上的任何现有文件。

4. **从磁带机中取出磁带。在磁带标签中写入文件名称。**

5. **验证复制的文件是否位于该磁带中。**

```
$ tar tvf /dev/rmt/n
```

有关列出 tar 磁带中的文件的更多信息，请参见第 331 页中的 “如何列出磁带中的文件 (tar)”。

示例 20–2 将文件复制到磁带 (tar)

以下示例说明如何将三个文件复制到磁带机 0 中的磁带。

```
$ cd /export/home/kryten
$ ls reports
reportA reportB reportC
$ tar cvf /dev/rmt/0 reports
a reports/ 0 tape blocks
a reports/reportA 59 tape blocks
a reports/reportB 61 tape blocks
a reports/reportC 63 tape blocks
$ tar tvf /dev/rmt/0
```

▼ 如何列出磁带中的文件 (tar)

1. **将磁带插入磁带机。**

2. **显示磁带内容。**

```
$ tar tvf /dev/rmt/n
```

t	列出磁带中的文件目录。
v	与 t 选项一同使用，提供有关磁带中的文件的详细信息。
f /dev/rmt/*n*	指明磁带设备。

示例 20–3 列出磁带中的文件 (tar)

以下示例显示磁带机 0 中的磁带上的文件列表。

```
$ tar tvf /dev/rmt/0
drwxr-xr-x   0/0        0 Jul 14 13:50 2010 reports/
-r--r--r--   0/0   206663 Jul 14 13:50 2010 reports/reportC
-r--r--r--   0/0   206663 Jul 14 13:50 2010 reports/reportB
-r--r--r--   0/0   206663 Jul 14 13:50 2010 reports/reportA
```

▼ 如何从磁带中检索文件 (tar)

1. **转到要放置文件的目录。**
2. **将磁带插入磁带机。**
3. **从磁带中检索文件。**

   ```
   $ tar xvf /dev/rmt/n [filenames]
   ```

x	指明应从指定的归档文件中提取文件。指定磁带机中的磁带上的所有文件都会被复制到当前目录。
v	检索时显示每个文件的名称。
f /dev/rmt/ *n*	指明包含归档文件的磁带设备。
filenames	指定要检索的文件。使用空格分隔多个文件。

   有关更多信息，请参见 tar(1) 手册页。

4. **验证文件是否已被复制。**

   ```
   $ ls -l
   ```

示例 20–4 检索磁带中的文件 (tar)

以下示例说明如何从磁带机 0 中的磁带检索所有文件。

```
$ cd /var/tmp
$ tar xvf /dev/rmt/0
x reports/, 0 bytes, 0 tape blocks
x reports/reportA, 0 bytes, 0 tape blocks
x reports/reportB, 0 bytes, 0 tape blocks
x reports/reportC, 0 bytes, 0 tape blocks
x reports/reportD, 0 bytes, 0 tape blocks
$ ls -l
```

故障排除 从磁带中提取的文件名称必须与归档文件中存储的文件名称完全匹配。如果对文件的名称或路径有任何疑问，请首先列出磁带中的文件。有关列出磁带中的文件的说明，请参见第 331 页中的 “如何列出磁带中的文件 (tar)”。

使用 pax 命令将文件复制到磁带

▼ 如何将文件复制到磁带 (pax)

1. 转到包含要复制的文件的目录。
2. 将可写磁带插入磁带机中。
3. 将文件复制到磁带。

   ```
   $ pax -w -f /dev/rmt/n filenames
   ```

   `-w` 启用写模式。

   `-f /dev/rmt/n` 确定磁带机。

   filenames 指明要复制的文件和目录。使用空格分隔多个文件。

   有关更多信息，请参见 pax(1) 手册页。
4. 验证文件是否已被复制到磁带。

   ```
   $ pax -f /dev/rmt/n
   ```
5. 从磁带机中取出磁带。在磁带标签中写入文件名称。

示例 20–5 将文件复制到磁带 (pax)

以下示例说明如何使用 pax 命令来复制当前目录中的所有文件。

```
$ pax -w -f /dev/rmt/0 .
$ pax -f /dev/rmt/0
filea fileb filec
```

使用 cpio 命令将文件复制到磁带

▼ 如何将目录中的所有文件复制到磁带 (cpio)

1. 转到包含要复制的文件的目录。
2. 将可写磁带插入磁带机中。

3. **将文件复制到磁带。**

   ```
   $ ls | cpio -oc > /dev/rmt/n
   ```

`ls`	为 `cpio` 命令提供文件名列表。
`cpio -oc`	指定 `cpio` 命令应该在复制出模式 (`-o`) 下操作，并以 ASCII 字符格式 (`-c`) 写入头信息。这些选项确保可移植到其他供应商的系统。
`> /dev/rmt/n`	指定输出文件。

   将目录中的所有文件复制到指定磁带机中的磁带，会覆盖该磁带中的任何现有文件。屏幕将显示复制的总块数。

4. **验证文件是否已被复制到磁带。**

   ```
   $ cpio -civt < /dev/rmt/n
   ```

`-c`	指定 `cpio` 命令应以 ASCII 字符格式读取文件。
`-i`	指定 `cpio` 命令应在复制入模式下操作，即使该命令仅列出了此时的文件。
`-v`	显示与 `ls -l` 命令的输出格式类似的输出。
`-t`	列出指定磁带机中的磁带上的文件目录。
`< /dev/rmt/n`	指定现有 `cpio` 归档文件的输入文件。

5. **从磁带机中取出磁带。在磁带标签中写入文件名称。**

示例 20–6 将目录中的所有文件复制到磁带 (cpio)

以下示例说明如何将 `/export/home/kryten` 目录中的所有文件复制到磁带机 0 中的磁带。

```
$ cd /export/home/kryten
$ ls | cpio -oc > /dev/rmt/0
1280 blocks
$ cpio -civt < /dev/rmt/0
-r--r--r--    1 kryten   staff      206663 Jul 14 13:52 2010, filea
-r--r--r--    1 kryten   staff      206663 Jul 14 13:52 2010, fileb
-r--r--r--    1 kryten   staff      206663 Jul 14 13:52 2010, filec
drwxr-xr-x    2 kryten   staff           0 Jul 14 13:52 2010, letters
drwxr-xr-x    2 kryten   staff           0 Jul 14 13:52 2010, reports
1280 blocks
```

▼ 如何列出磁带中的文件 (cpio)

注 – 列出磁带中的目录需要很长时间，因为 cpio 命令必须处理整个归档文件。

1. **将归档文件磁带插入磁带机。**
2. **列出磁带中的文件。**

```
$ cpio -civt < /dev/rmt/n
```

示例 20–7 列出磁带中的文件 (cpio)

以下示例说明如何列出磁带机 0 中的磁带上的文件。

```
$ cpio -civt < /dev/rmt/0
-r--r--r--    1 kryten   staff      206663 Jul 14 13:52 2010, filea
-r--r--r--    1 kryten   staff      206663 Jul 14 13:52 2010, fileb
-r--r--r--    1 kryten   staff      206663 Jul 14 13:52 2010, filec
drwxr-xr-x    2 kryten   staff           0 Jul 14 13:52 2010, letters
drwxr-xr-x    2 kryten   staff           0 Jul 14 13:52 2010, reports
1280 blocks
```

▼ 如何从磁带中检索所有文件 (cpio)

如果已使用相对路径名创建了归档文件，则检索文件时会在当前目录中以目录形式生成输入文件。但是，如果已使用绝对路径名创建了归档文件，则将使用相同的绝对路径在系统中重新创建该文件。

注意 – 使用绝对路径名可能会有危险，因为可能会覆盖系统中的现有文件。

1. **转到要放置文件的目录。**
2. **将磁带插入磁带机。**
3. **从磁带中提取所有文件。**

```
$ cpio -icvd < /dev/rmt/n
```

`-i`	从标准输入中提取文件。
`-c`	指定 cpio 命令应以 ASCII 字符格式读取文件。
`-v`	在以与 ls 命令输出类似的格式检索文件时，显示这些文件。
`-d`	根据需要创建目录。

`< /dev/rmt/`*n* 指定输出文件。

4 验证文件是否已被复制。

```
$ ls -l
```

示例 20–8 从磁带中检索所有文件 (cpio)

以下示例说明如何从磁带机 0 中的磁带检索所有文件。

```
$ cd /var/tmp
cpio -icvd < /dev/rmt/0
answers
sc.directives
tests
8 blocks
$ ls -l
```

▼ 如何从磁带中检索特定文件 (cpio)

1 转到要放置文件的目录。

2 将磁带插入磁带机。

3 从磁带中检索文件子集。

```
$ cpio -icv "*file" < /dev/rmt/n
```

`-i` 从标准输入中提取文件。

`-c` 指定 `cpio` 命令应以 ASCII 字符格式读取头。

`-v` 在以与 `ls` 命令输出类似的格式检索文件时，显示这些文件。

"*file* " 指定与该模式匹配的所有文件都将复制到当前目录。可以指定多种模式，但是每种模式都必须用双引号引起来。

`< /dev/rmt/`*n* 指定输入文件。

有关更多信息，请参见 `cpio(1)` 手册页。

4 验证文件是否已被复制。

```
$ ls -l
```

示例 20–9 从磁带中检索特定文件 (cpio)

以下示例说明如何从磁带机 0 中的磁带检索带有 `chapter` 后缀的所有文件。

```
$ cd /home/smith/Book
$ cpio -icv "*chapter" < /dev/rmt/0
Boot.chapter
Directory.chapter
Install.chapter
Intro.chapter
31 blocks
$ ls -l
```

将文件复制到远程磁带设备

▼ 如何将文件复制到远程磁带设备（tar 和 dd）

1. **必须满足以下先决条件，才能使用远程磁带机：**

   a. **本地主机名和（可选）执行复制的用户的用户名必须出现在远程系统的 `/etc/hosts.equiv` 文件中。或者，执行复制的用户在远程计算机上必须具有可访问的对应起始目录，而且在 `$HOME/.rhosts` 中必须具有本地计算机名。**

      有关更多信息，请参见 `hosts.equiv(4)` 手册页。

   b. **远程系统的项必须位于本地系统的 `/etc/inet/hosts` 文件中，或位于名称服务 `hosts` 文件中。**

2. **要测试您是否具有执行远程命令的适当权限，请键入以下内容：**

   ```
   $ rsh remotehost echo test
   ```

   如果系统向您回显了 `test`，则您有权执行远程命令。如果系统向您回显了 `Permission denied`，请按照步骤 1 所述检查您的设置。

3. **转到要放置文件的目录。**

4. **将磁带插入磁带机。**

5. **将文件复制到远程磁带机。**

   ```
   $ tar cvf - filenames | rsh remote-host dd of=/dev/rmt/n obs=block-size
   ```

`tar cf`	创建磁带归档文件，归档时列出文件并指定磁带设备。
`v`	提供有关 tar 文件项的其他信息。
`-`（连字符）	表示磁带设备的占位符。
filenames	确定要复制的文件。使用空格分隔多个文件。
`rsh` | *remote-host*	将 `tar` 命令的输出传输到远程 shell。

`dd of=` /dev/rmt/*n*　表示输出设备。

`obs=`*block-size*　表示分块因子。

6 **从磁带机中取出磁带。在磁带标签中写入文件名称。**

示例 20–10 将文件复制到远程磁带机（tar 和 dd）

```
# tar cvf - * | rsh mercury dd of=/dev/rmt/0 obs=126b
a answers/ 0 tape blocks
a answers/test129 1 tape blocks
a sc.directives/ 0 tape blocks
a sc.directives/sc.190089 1 tape blocks
a tests/ 0 tape blocks
a tests/test131 1 tape blocks
6+9 records in
0+1 records out
```

▼ 如何从远程磁带设备中提取文件

1 **将磁带插入磁带机。**

2 **转到临时目录。**

```
$ cd /var/tmp
```

3 **从远程磁带设备中提取文件。**

```
$ rsh remote-host dd if=/dev/rmt/n | tar xvBpf -
```

`rsh` *remote-host*　指明使用 `dd` 命令从磁带设备中提取文件时启动的远程 shell。

`dd if=/dev/rmt/`*n*　指明输入设备。

`| tar xvBpf -`　将 `dd` 命令的输出传输到用于恢复文件的 `tar` 命令。

4 **验证文件是否已被提取。**

```
$ ls -l
```

示例 20–11 从远程磁带机中提取文件

```
$ cd /var/tmp
$ rsh mercury dd if=/dev/rmt/0 | tar xvBpf -
x answers/, 0 bytes, 0 tape blocks
x answers/test129, 48 bytes, 1 tape blocks
20+0 records in
20+0 records out
x sc.directives/, 0 bytes, 0 tape blocks
x sc.directives/sc.190089, 77 bytes, 1 tape blocks
x tests/, 0 bytes, 0 tape blocks
```

```
x tests/test131, 84 bytes, 1 tape blocks
$ ls -l
```

◆ ◆ ◆ 第 21 章

21 管理磁带机（任务）

本章介绍了如何在 Oracle Solaris OS 中管理磁带机。

以下是本章中逐步说明的列表。

- 第 344 页中的 “如何显示磁带机状态”
- 第 345 页中的 “重新张紧盒式磁带”
- 第 345 页中的 “反绕盒式磁带”

以下是本章中概述信息的列表。

- 第 341 页中的 “选择要使用的介质”
- 第 342 页中的 “备份设备名称”
- 第 344 页中的 “显示磁带机状态”

选择要使用的介质

通常使用以下磁带介质来备份 Oracle Solaris 系统 :

- 1/2 英寸盘式磁带
- 1/4 英寸流盒式磁带
- 8 毫米盒式磁带
- 4 毫米盒式磁带 (DAT)

选择哪一种介质取决于支持该介质的设备以及用于存储文件的介质（通常为磁带）是否可用。尽管必须从本地系统进行备份，但可以将文件写入远程设备。

下表显示了用于备份文件系统的典型磁带设备。每个设备的存储容量取决于驱动器的类型和写入磁带中的数据。

表 21–1 介质存储容量

备份介质	存储容量
1/2 英寸盘式磁带	140 MB (6250 bpi)
2.5 GB 1/4 英寸盒式 (QIC) 磁带	2.5 GB
DDS3 4 毫米盒式磁带 (DAT)	12–24 GB
14 GB 8 毫米盒式磁带	14 GB
DLT 7000 1/2 英寸盒式磁带	35–70 GB

备份设备名称

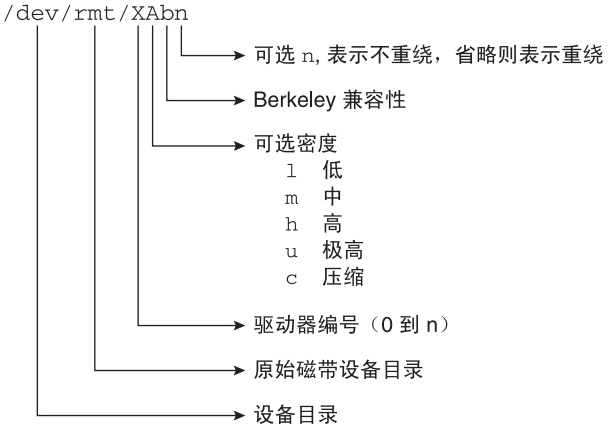
通过提供逻辑设备名称，可以指定用于备份的磁带或设备。此名称指向包含“原始”设备文件的子目录，并且名称中包括磁带机的逻辑单元号。磁带机命名约定使用逻辑设备名称而不是物理设备名称。下表显示了此命名约定。

表 21–2 备份设备的基本设备名称

设备类型	名称
磁带	`/dev/rmt/`*n*

通常，可以按照下图所示指定磁带设备。

图 21–1 磁带机设备名称



如果不指定密度，则磁带机通常会按照其“首选”密度进行写入。首选密度通常表示磁带机支持的最高密度。大多数 SCSI 磁带机可以自动检测磁带的密度或格式，并相应地进行读取。要确定磁带机支持的不同密度，请查看 /dev/rmt 子目录。该子目录中包含支持每种磁带的不同输出密度的磁带设备文件集。

此外，一个 SCSI 控制器最多可有七个 SCSI 磁带机。

指定磁带机的反绕选项

通常，可以按照逻辑单元号（范围从 0 至 *n*）指定磁带机。下表介绍了如何使用反绕选项或不反绕选项来指定磁带设备名称。

表 21–3 指定磁带机反绕或不反绕

磁带机和反绕值	使用此选项
第一个磁带机，反绕	/dev/rmt/0
第一个磁带机，不反绕	/dev/rmt/0n
第二个磁带机，反绕	/dev/rmt/1
第二个磁带机，不反绕	/dev/rmt/1n

指定磁带机的不同密度

缺省情况下，磁带机按照“首选”密度写入，该密度通常为磁带机支持的最高密度。如果不指定磁带设备，则命令会按照设备支持的缺省密度写入磁带机编号 0。

要将磁带传输到其磁带机仅支持特定密度的系统，请指定按照所需密度写入的设备名称。下表介绍了如何指定磁带机的不同密度。

表 21–4 指定磁带机的不同密度

磁带机、密度和反绕值	使用此选项
第一个磁带机，低密度，反绕	/dev/rmt/0l
第一个磁带机，低密度，不反绕	/dev/rmt/0ln
第二个磁带机，中密度，反绕	/dev/rmt/1m
第二个磁带机，中密度，不反绕	/dev/rmt/1mn

其他密度值显示在第 342 页中的“备份设备名称”中。

显示磁带机状态

可使用 mt 命令的 status 选项获取有关磁带机的状态信息。mt 命令可报告有关 /kernel/drv/st.conf 文件中介绍的所有磁带机的信息。

▼ 如何显示磁带机状态

1. **将磁带装入需要了解其相关信息的磁带机。**
2. **显示磁带机状态。**

   ```
   # mt -f /dev/rmt/n status
   ```

3. **重复步骤 1–2，替换磁带机编号 0、1、2、3 等，以显示有关所有可用磁带机的信息。**

示例 21–1 显示磁带机状态

以下示例显示了 QIC-150 磁带机 (/dev/rmt/0) 的状态：

```
$ mt -f /dev/rmt/0 status
Archive QIC-150 tape drive:
   sense key(0x0)= No Additional Sense   residual= 0   retries= 0
   file no= 0   block no= 0
```

以下示例显示了 Exabyte 磁带机 (/dev/rmt/1) 的状态：

```
$ mt -f /dev/rmt/1 status
Exabyte EXB-8200 8mm tape drive:
sense key(0x0)= NO Additional Sense residual= 0  retries= 0
file no= 0   block no= 0
```

以下示例显示了快速轮询系统并查找其所有磁带机的方法：

```
$ for drive in 0 1 2 3 4 5 6 7
> do
> mt -f /dev/rmt/$drive status
> done
Archive QIC-150 tape drive:
   sense key(0x0)= No Additional Sense   residual= 0   retries= 0
   file no= 0   block no= 0
/dev/rmt/1: No such file or directory
/dev/rmt/2: No such file or directory
/dev/rmt/3: No such file or directory
/dev/rmt/4: No such file or directory
/dev/rmt/5: No such file or directory
/dev/rmt/6: No such file or directory
/dev/rmt/7: No such file or directory
$
```

处理盒式磁带

如果读取磁带时出现错误，则可以重新张紧磁带、清洁磁带机，然后重试。

重新张紧盒式磁带

使用 mt 命令重新张紧盒式磁带。

例如：

```
$ mt -f /dev/rmt/1 retension
$
```

注 – 请勿重新张紧非 QIC 磁带机。

反绕盒式磁带

要反绕盒式磁带，请使用 mt 命令。

例如：

```
$ mt -f /dev/rmt/1 rewind
$
```

磁带机维护和介质处理指南

不能读取的备份磁带是无用的。因此，请定期清洁和检查磁带机，以确保其正确操作。有关清洁磁带机的过程说明，请参见硬件手册。可通过将某些文件复制到磁带，读回这些文件，然后将原始文件与复制的文件进行比较，来检查磁带硬件

请注意，硬件可能会出现系统未报告的故障。

请始终在备份后为磁盘设置标签。此标签决不能更改。每次执行备份时，请创建另一个包含以下信息的磁带标签：

- 备份日期
- 备份的计算机和文件系统的名称
- 备份级别
- 磁带编号（1 至 *n* 中的任意一个数字，前提是备份跨多个卷）
- 特定于站点的任何信息

请将磁带存储在远离磁性设备的无尘安全位置。有些地方会将归档的磁带存储在偏僻位置的防火柜中。

应该创建一个日志并对其进行维护，该日志用于跟踪存储每个作业（备份）的介质（磁带卷）和每个备份文件的位置。

索引

数字和符号

/export/home 目录，300
1394 (FireWire) 支持, 说明，108
4.3 Tahoe 文件系统，295
9660 DVD 格式，36

A

autofs，303

B

BSD Fat Fast File 系统，295

C

cdrw 命令
  说明，43
  限制访问（如何），46
  写入数据 CD 和 DVD 以及音频 CD（概述），45
cfgadm
  PCI 热插拔（概述），76
  SCSI 热插拔（概述），76
cpio 命令
  （概述），329
  从磁带中提取所有文件（如何），335
  复制文件系统之间的目录（如何），329
  列出磁带中的文件（如何），335
CTFS 文件系统，298

D

datadm 命令，163
/dev/dsk 目录，71
/dev/rdsk 目录，71
/dev 文件系统, 说明，57
devfsadm 命令，70
DOS, 文件系统，295
driver not attached 消息，62
dumpadm 命令，72
DVD, ISO 9660 格式，36
DVD-RAM, 创建文件系统（如何），31
DVD-ROM，296

E

EFI 标签
  （概述），178
  安装系统，180
  问题疑难解答，180
  限制，179
  与 VTOC 标签比较，179
eject 命令, 可移除介质（如何），39

F

FDFS 文件系统，298
FIFOFS 文件系统，298
format 实用程序
  analyze 菜单，287
  defect 菜单，288–289

format 实用程序（续）
fdisk 菜单，286
partition 菜单，285, 286
（概述），182
创建 Solaris fdisk 分区（如何），221
功能和优势，182
何时使用，183
恢复损坏的磁盘标签（如何），202
确定磁盘是否已格式化（如何），192
确定系统中的磁盘（如何），190
确定系统中的磁盘（示例），191
使用帮助功能，290
使用指南，183–184
输入，289, 290
输入命令名称（如何），290
为磁盘设置标签
示例，197
显示磁盘分片信息（示例），195
主菜单，284
fsstat 命令, 说明，293
fsstat 命令（示例），293
fstypes 文件，304
fuser 命令
查明可移除介质是否在使用（如何），38
中止正在访问可移除介质的进程（如何），38

G

GRUB
使用 GRUB 管理磁盘
x86，176

H

hald, 说明，22
High Sierra 文件系统，295
/home（自动挂载），303
HSFS, **请参见**High Sierra 文件系统

I

InfiniBand 设备
动态重新配置（概述），153
概述，149
更新 IB p_key 表（如何），160
配置 IB 端口、HCA_SVC 或 VPPA 设备（如何），156
配置 IB 伪设备（如何），157
取消配置 IB 端口、HCA_SVC 或 VPPA（如何），156
取消配置 IB 伪设备（如何），157
删除现有 IB 端口、HCA_SVC 或 VPPA 通信服务（如何），161
添加 VPPA 通信服务（如何），161
显示 HCA 的内核 IB 客户机（如何），158
显示 IB 通信服务（如何），160
显示（如何），154
installboot 命令为 ZFS 根文件系统，211
Internet 存储名称服务, **请参见**iSNS
iSCSI
（概述），227
动态和静态目标搜索，231
访问 iSCSI 磁盘（如何），240
监视 iSCSI 配置（如何），259
解决 iSCSI 配置问题（如何），266
配置单向或双向 CHAP 验证（如何），252
删除搜索的 iSCSI 目标（如何），238
修改 iSCSI 启动器和目标参数（如何），263
一般 iSCSI 错误消息，268
iscsiadm list, 显示 ISCSI 配置信息（示例），259
iscsiadm modify 命令
启用 CHAP（示例），253
启用或禁用静态或动态目标（示例），238
iscsiadm remove 命令, 删除静态或动态目标（示例），238
iSNS
nodes，274
管理客户端，280–282
管理目标，280–282
管理启动器，280–282
技术概述，273
客户端，274
缺省搜索域，274
显示客户端状态，281

iSNS（续）
显示目标状态，281
显示启动器状态，281
iSNS 服务器
配置，274
配置, CLI，277–280
启用、禁用服务器状态通知，276
启用缺省搜索域集，278
设置重试阈值，276–277
搜索域
创建, CLI，278–279
删除, CLI，282
删除客户端，281
添加, CLI，279
显示状态, CLI，281
指定客户端, CLI，279–280
搜索域集
创建, CLI，278
禁用, CLI，282
删除，282
显示状态, CLI，281
显示设置，277
指定数据存储位置，273, 277
ISO 9660 文件系统，295
ISO 标准, 9660 DVD 格式，36

K
/kernel/drv 目录，61

M
media was found 消息，29
mkisofs 命令, 创建数据 CD 或 DVD 文件系统（如何），48
MNTFS 文件系统，300
mnttab 文件，301
mt 命令，345

N
NAMEFS 文件系统，298
NFS
vfstab 项用于，314
服务器说明，302
说明，302
nfsd 守护进程
开始，40
验证是否正在运行，40
no media was found 消息，29

O
OBJFS 文件系统，298
/opt 目录，300

P
PCFS 文件系统，295
PCI Express(PCIe) 支持, 说明，58
PCI 设备
PCI 配置问题故障排除，96
删除 PCI 适配卡（如何），93
添加 PCI 适配卡（如何），94
显示 PCI 插槽配置信息（如何），92
/proc 目录，298, 300
prtvtoc 命令，72
（示例），201

R
RCM 脚本
概述，98
命令，99
rmvolmgr, 说明，22
Rock Ridge 扩展（HSFS 文件系统），295

S
SATA 设备
配置 SATA 设备（如何），97
取消配置 SATA 设备（如何），97
显示 SATA 设备信息（如何），96

SCSI 磁带机，343
SCSI 设备
  SCSI 配置问题故障排除，89
  解决失败的 SCSI 取消配置操作（如何），90
  连接 SCSI 控制器（如何），85
  配置 SCSI 控制器（如何），83
  配置 SCSI 设备（如何），83
  取消配置 SCSI 控制器（如何），82
  使用 cfgadm 命令断开（如何），84
  添加 SCSI 设备至 SCSI 总线（如何），86
  显示有关（如何），82
  移除 SCSI 设备（如何），88
  在 SCSI 控制器上更换相同 SCSI 设备（如何），87
share 命令，303
  使可移除介质可供其他系统使用（如何），40
shareall 命令，303
SHAREFS 文件系统，299
Solaris fdisk 分区，指南，221
SPECFS 文件系统，299
SunOS 缺省的文件系统，300
svcadm disable，（示例），24
swap 命令，72
swapadd 命令，323
SWAPFS 文件系统，299
sysdef 命令，63

T

tar 命令
  从磁带中检索文件（如何），332
  列出磁带中的文件（如何），331
  使用 dd 命令从远程磁带中检索文件（如何），338
  使用 dd 命令将文件复制到远程磁带（如何），337
tar命令，（概述），330
/tmp 目录，297，300
TMPFS 文件系统，概述，297

U

uDAPL
  （概述），161
  更新 DAT 静态注册表（如何），163
  启用（如何），162
  在 DAT 静态注册表中注册服务提供商（如何），163
  在 DAT 静态注册表中注销服务提供商（如何），163
UDF 文件系统，296
UFS 文件系统，295
  挂载，314
  使用 /etc/vfstab 挂载，314
UNIX 文件系统，295
USB 设备
  2.0 版本的设备概述，115
  EHCI、OHCI 和 UHCI 支持，114
  Oracle Solaris USB 体系结构 (USB Architecture, USBA)，114
  Prolific 和 Keyspan 串行适配器支持，113
  USB 2.0 支持，115
  （概述），110
  标识主 USB 音频设备（如何），141
  磁盘设备（概述），124
  电缆，119
  电源管理，119
  复合设备，112
  挂载海量存储（如何），137
  挂载或卸载海量存储设备（如何），136
  滚轮鼠标支持，118
  海量存储设备的疑难解答提示，138
  键盘和鼠标设备，117
  接口关联描述符 (Interface Association Descriptor, IAD) 支持，108
  可热插拔属性，108
  可移除海量存储（概述），123
  连接 USB 设备（如何），146
  名称，112
  配置 USB 设备（如何），146
  驱动程序，112
  取消挂载海量存储（如何），137
  取消配置设备（如何），145
  热插拔（概述），126
  设备节点，114

USB 设备（续）
设备类，112
使用非符合的海量存储设备（概述），124
首字母缩略词，110
添加 USB 海量存储设备（如何），126
添加 USB 相机（如何），127
添加音频设备（如何），141
物理设备分层结构，111
显示 USB 设备信息（如何），129
显示总线信息（如何），144
虚拟键盘和鼠标支持，109
移除 USB 海量存储设备（如何），128
以逻辑方式断开 USB 设备连接（如何），146
以逻辑方式断开 USB 设备子树连接（如何），147
音频
（概述），140
更改主设备（如何），142
设备所有权，142
音频设备问题疑难解答，142
在海量存储设备中创建 Solaris 分区和修改分片（如何），135
在海量存储设备中创建文件系统（如何），130
在海量存储设备中修改分区和创建 PCFS 文件系统（如何），132
重置 USB 设备（如何），147
主机控制器和根集线器，118
总线供电设备，117
总线说明，111
组合设备，112

V

/var 目录，300
vfstab 文件，304, 323
LOFS 的项，313
添加交换至，323
添加项至（如何），313
vold, 删除，22
volfs, 删除，22

Z

ZFS 根文件系统的磁盘设置
连接（如何）
SPARC，206
x86，214
ZFS 文件系统，295
设置磁盘（如何）
x86，220
ZFS 文件系统的磁盘
设置（如何）
SPARC，212

备

备份, 设备名称，342–343

不

不受支持的设备，61

查

查找, 文件系统的类型，304

创

创建
DVD-RAM 上的文件系统（如何），31
ZFS 根文件系统磁盘分片（如何）
x86，215
ZFS 根文件系统的磁盘分片
SPARC，207
回送文件系统（概述），306
可移除介质上的文件系统（如何），31
数据 CD 或 DVD 文件系统（如何），48
在 USB 海量存储设备中创建 Solaris 分区和修改分片（如何），135
在 USB 海量存储设备中创建文件系统（如何），130

磁

磁带
- 存储容量，341
- 大小，341
- 使用 tar 命令检索文件（如何），332

磁带机
- 反绕，343
- 维护，345
- 最大 SCSI，343

磁带设备（命名），73

磁盘
- 创建文件系统（如何），31
- 格式化（概述），184
- 何时格式化（概述），192
- 恢复损坏的磁盘标签（概述），202
- 恢复损坏的磁盘标签（如何恢复），202
- 确定是否已格式化（如何），192
- 使用 rmformat 格式化（如何），30
- 使用卷管理装入（如何），29
- 为 ZFS 文件系统设置（概述）
  - SPARC，206
  - x86，214–221

磁盘标签
- 创建（概述），196
- 说明，178

磁盘分片
- 定义，181
- 为 ZFS 根文件系统（如何）
  - x86，215
- 为 ZFS 根文件系统创建
  - SPARC，207
- 显示信息（概述），194

磁盘控制器，72

存

存储（虚拟内存），定义，319

存储容量（介质），341

错

错误消息，iSCSI，268

带

带有面向总线的控制器的磁盘，73

弹

弹出，可移除介质（如何），39

动

动态重新配置，InfiniBand 设备，153

动态重新配置（概述），76

断

断开，SCSI 控制器（如何），84

访

访问
- iSCSI 磁盘（如何），240
- 磁带设备，73
- 磁盘设备，71
- 可移除介质（如何），37

分

分片（定义），181

分区（交换），定义，319

服

服务 (SMF)，hal 服务，描述，21

浮

浮动分片 (free hog slice)，**请参见**提供分片

复

复制
- 单个文件，使用 cpio 命令（概述），329
- 多组文件，使用 cpio 命令（概述），329
- 系统之间的目录，使用 cpio 命令（概述），329

概

概述, iSNS 技术，273

格

格式化, 具有 rmformat 的磁盘（如何），30
格式化磁盘,（概述），184

根

根目录 (/) 文件系统或目录，300

更

更改, 主 USB 音频设备（如何），142
更换, SCSI 控制器上的相同 SCSI 设备（如何），87
更新
- DAT 静态注册表（如何），163
- IB p_key 表（如何），160

共

共享, 文件，302–303

故

故障排除
- PCI 配置问题，96
- SCSI 配置问题，89
- 失败的 SCSI 取消配置操作，90

挂

挂载
- NFS 文件系统，314
- UFS 文件系统，314
- USB 海量存储设备（如何），136, 137
- 可移除介质
  - 自动挂载相比于，25
- 使用 /etc/vfstab 挂载文件系统，314
- 手动挂载远程可移除介质（示例），42
- 文件系统自动，303

挂载表，301
挂载点, 定义，301

管

管理
- iSNS，281
  - 客户端，CLI，280–282
- 磁盘，使用 GRUB
  - x86，176

回

回送文件系统 (LOFS), 创建（概述），306

基

基于磁盘的文件系统，295
基于网络的文件系统，297

监

监视, iSCSI 配置（如何），259

检

检测介质结尾, cpio 命令，329
检查, 可移除介质上的文件系统（如何），32
检索, 磁带中的文件，使用 tar 命令（如何），332

交

交换分区, 定义，319
交换文件
  添加至 vfstab，323
  显示，324–325

节

节点
  iSNS，274
  iSNS, 管理, CLI，280–282
  iSNS, 显示状态, CLI，281

解

解决
  iSCSI 配置问题（如何），266
  失败的 SCSI 取消配置操作（如何），90

介

介质结尾检测, cpio 命令，329

禁

禁用
  可移除介质服务，24
  可移除介质服务（如何），37
  可移除介质上的写保护（如何），33

进

进程文件系统,（概述），298
进程文件系统 (PROCFS)，298

开

开始, nfsd 守护进程，40

可

可移除介质
  dbus 服务, 描述，21
  /media
    挂载点，21, 36
  rmvolmgr 服务, 描述，21
  查明介质是否在使用（如何），38
  创建文件系统（如何），31
  弹出（如何），39
  访问（如何），37
  访问其他系统上的介质（示例），42
  服务, 禁用，24
  服务, 描述，21
  格式化具有 rmformat 的磁盘（如何），30
  挂载
    手动和自动的比较，25
  挂载远程介质（示例），42
  管理, 优点，24
  管理改进，21
  检查文件系统（如何），32
  禁用或启用可移除介质服务（如何），37
  启用写保护（如何），33
  添加新的可移除介质驱动器（如何），36
  向后兼容性 (/media)，22
  修复可移除介质上的坏块（如何），32
  在 DVD-RAM 上创建文件系统（如何），31
  中止正在访问的进程（如何），38
  装入磁盘（如何），29

客

客户端
  iSNS，274
  iSNS, 管理, CLI，280–282
  iSNS, 显示状态, CLI，281

块

块磁盘设备接口
  定义，71
  何时使用，72

连

连接, SCSI 控制器（如何），85

临

临时文件系统 (TMPFS), 概述，297

逻

逻辑设备名称
- 磁带，73
- 磁盘，71
- 定义，70
- 可移除介质，74

目

目标
- iSNS, 管理, CLI，280–282
- iSNS，显示状态, CLI，281

目录
- `/proc`，298
- `/tmp`，297
- 使用 `cpio` 命令在文件系统之间进行复制（概述），329

内

内存存储（虚拟）, 定义，319

配

配置
- IB 端口、HCA_SVC 或 VPPA 设备（如何），156
- IB 伪设备（如何），157
- iSCSI 的单向或双向 CHAP 验证（如何），252
- iSNS 服务器，274, 277–280
- SATA 设备（如何），97
- SCSI 控制器（如何），83
- SCSI 设备（如何），83
- USB 设备（如何），146

启

启动器
- iSNS, 管理, CLI，280–282
- iSNS, 显示状态, CLI，281

启用
- uDAPL，162
- 可移除介质服务（如何），37
- 可移除介质上的写保护（如何），33

取

取消挂载, USB 海量存储设备（如何），137

取消配置
- IB 端口、HCA_SVC 或 VPPA 设备（如何），156
- IB 伪设备（如何），157
- SATA 设备（如何），97
- SCSI 控制器（如何），82
- USB 设备（如何），145

缺

缺省, `/tmp` (TMPFS) 的文件系统，297

缺省的, SunOS 文件系统，300

缺省搜索域, iSNS，274

缺省搜索域集, 启用，278

确

确定
- 设备，63
- 文件系统类型，304
- 系统中的磁盘（如何），190
- 已挂载文件系统，307
- 主 USB 音频设备（如何），141

热

热插拔
- PCI 设备（概述），91
- （概述），76
- 连接 SCSI 控制器（如何），85
- 配置 SCSI 控制器（如何），83
- 配置 SCSI 设备（如何），83
- 配置 USB 设备（如何），146
- 取消配置 SCSI 设备（如何），82
- 取消配置 USB 设备（如何），145
- 删除 PCI 适配卡（如何），93
- 使用 `cfgadm` 命令断开 SCSI 控制器（如何），84
- 添加 PCI 适配卡（如何），94
- 添加 SCSI 设备至 SCSI 总线（如何），86
- 移除 SCSI 设备（如何），88
- 以逻辑方式断开 USB 设备连接（如何），146
- 以逻辑方式断开 USB 设备子树连接（如何），147
- 以逻辑方式连接 USB 设备（如何），146
- 在 SCSI 控制器上更换相同 SCSI 设备（如何），87

删

删除
- PCI 适配卡（如何），93
- 搜索的 iSCSI 目标 (如何)，238
- 现有 IB 端口、HCA_SVC 或 VPPA 通信服务（如何），161

设

设备, 访问，69

设备名称, 备份，342–343

设备驱动程序
- 定义，60
- 添加，69

设置
- ZFS 文件系统磁盘（概述）
  - x86，214–221
- ZFS 文件系统的磁盘（概述）
  - SPARC，206

搜

搜索域
- iSNS
  - 创建, CLI，278–279
  - 添加, CLI，279
  - 显示状态, CLI，281
  - 指定客户端, CLI，279–280
- 客户端
  - 删除，CLI，281
- 删除，282

搜索域集
- iSNS
  - 创建, CLI，278
  - 显示状态, CLI，281
- 禁用
  - 显示状态, CLI，282
- 删除，282

添

添加
- PCI 适配卡（如何），94
- SCSI 设备至 SCSI 总线（如何），86
- USB 海量存储设备（如何），126
- USB 相机（如何），127
- USB 音频设备（如何），141
- VPPA 通信服务（如何），161
- 交换至 `vfstab`，323
- 添加新的可移除介质驱动器（如何），36
- 项至 `/etc/vfstab` 文件（如何），313

停

停止
- 访问文件系统的所有进程（如何），316
- 所有进程访问文件系统（如何），316
- 中止正在访问可移除介质的进程（如何），38

维

维护磁带机，345

伪

伪文件系统，（概述），297

为

为 ZFS 根文件系统安装引导块（如何），
SPARC，211

文

文件
/etc/default/fs，304
/etc/dfs/fstypes，304
共享，302–303
使用 tar 命令在磁带中检索（如何），332
位于 /proc 目录中，298
文件系统
4.3 Tahoe，295
BSD Fat Fast，295
CTFS，298
DOS，295
/export/home，300
FDFS，298
FIFOFS，298
High Sierra，295
ISO 9660，295
MNTFS，300
NAMEFS，298
OBJFS，298
/opt，300
PCFS，295
/proc，300
PROCFS，（概述），298
SHAREFS，299
SPECFS，299
SWAPFS，299
TMPFS，297
UFS，295
UNIX，295
/var，300
ZFS，295
查找类型，304
文件系统（续）
创建（概述）
回送 (LOFS)，306
共享，302–303
挂载表，301
基于磁盘，295
基于网络的，297
进程，（概述），298
类型，295
缺省的 SunOS，300
使有用（概述），306
停止所有进程访问（如何），316
伪，（概述），297
文件系统类型，295

物

物理设备名称
定义，70

显

显示
HCA 的内核 IB 客户机（如何），158
IB 通信服务（如何），160
InfiniBand 设备信息（如何），154
PCI 插槽配置信息（如何），92
SATA 设备（如何），96
USB 设备信息（如何），129
USB 总线信息（如何），144
磁盘分片信息（概述），194
交换空间，324–325
可移除介质用户（如何），38
系统的配置信息，65
系统配置信息，62
有关 SCSI 设备的信息，82
显示设置, iSNS 服务器，277

限

限制, 可移除介质访问（如何），46

写
写入, 数据 CD 和 DVD 以及音频 CD（概述），45

卸
卸载, USB 海量存储设备（如何），136

修
修复, 可移除介质上的坏块（如何），32
修改, iSCSI 启动器和目标参数（如何），263

虚
虚拟内存存储, 定义，319

验
验证, `nfsd` 守护进程是否正在运行，40

疑
疑难解答
EFI 磁盘标签，180
USB 海量存储设备，138
USB 音频设备问题，142

移
移除
SCSI 设备（如何），88
USB 海量存储设备（如何），128

以
以逻辑方式断开
USB 设备连接（如何），146
USB 设备子树连接（如何），147
以逻辑方式连接, USB 设备（如何），146

用
用于 ZFS 根文件系统的 `installgrub` 命令，219

原
原始磁盘设备接口，71, 72

在
在 USB 海量存储设备中修改, 分区和创建 PCFS 文件系统（如何），132

指
指定磁盘分片，72

中
中止, 正在访问可移除介质的进程（如何），38

重
重置, USB 设备（如何），147

注
注册, 在 DAT 静态注册表中注册服务提供者（如何），163
注销, 在 DAT 静态注册表中注册服务提供者（如何），163

装
装入, 磁盘（如何），29

自

自动挂载, 和 /home, 303
自动配置过程, 61