Veritas™ Dynamic Multi-Pathing 管理指南

Linux

6.0.1

Symantec™

Veritas™ Dynamic Multi-Pathing 管理指南

本手册所述软件是根据许可协议而提供，仅可按该协议的条款使用。

产品版本：6.0.1

文档版本：6.0.1 Rev 0

法律声明

Copyright © 2012 Symantec Corporation. © 2012 Symantec Corporation 版权所有。All rights reserved. 保留所有权利。

Symantec、Symantec 徽标、Veritas、Veritas Storage Foundation、CommandCentral、NetBackup、Enterprise Vault 和 LiveUpdate 是 Symantec Corporation 或其附属公司在美国和其他国家/地区的商标或注册商标。“Symantec”和“赛门铁克”是 Symantec Corporation 在中国的注册商标。其他名称可能为其各自所有者的商标，特此声明。

本文档中介绍的产品根据限制其使用、复制、分发和反编译/逆向工程的授权许可协议分发。未经 Symantec Corporation 及其特许人（如果存在）事先书面授权，不得以任何方式任何形式复制本文档的任何部分。

本文档按“现状”提供，对于所有明示或暗示的条款、陈述和保证，包括任何适销性、针对特定用途的适用性或无侵害知识产权的暗示保证，均不提供任何担保，除非此类免责声明的范围在法律上视为无效。Symantec Corporation 不对任何与提供、执行或使用本文档相关的伴随或后果性损害负责。本文档所含信息如有更改，恕不另行通知。

根据 FAR 12.212 中的定义，授权许可的软件和文档被视为“商业计算机软件”，受 FAR 第 52.227-19 节“Commercial Computer Software - Restricted Rights”（商业计算机软件受限权利）和 DFARS 第 227.7202 节“Rights in Commercial Computer Software or Commercial Computer Software Documentation”（商业计算机软件或商业计算机软件文档权利）中的适用规定，以及所有后续法规中规定的权利的制约。美国政府仅可根据本协议的条款对授权许可的软件和文档进行使用、修改、发布复制、执行、显示或披露。

Symantec Corporation
350 Ellis Street
Mountain View, CA 94043

http://www.symantec.com

技术支持

Symantec 技术支持具有全球性支持中心。技术支持的主要任务是响应有关产品特性和功能的特定查询。技术支持小组还负责编写我们的联机知识库文章。技术支持小组与 Symantec 内的其他职能部门相互协作，及时解答您的问题。例如，技术支持小组与产品工程和 Symantec 安全响应中心协作，提供警报服务和病毒定义更新服务。

Symantec 提供的维护服务包括：

- 一系列支持服务，使您能为任何规模的单位选择适用的支持服务
- 通过电话和 Web 支持快速响应并提供最新信息
- 升级保证可保证软件顺利升级
- 全天候提供全球支持
- 高级功能，包括“客户管理服务”

有关 Symantec 维护计划的更多信息，请访问我们的网站：

http://www.symantec.com/zh/cn/support/index.jsp

与技术支持联系

具有有效维护协议的客户可以通过以下网址访问技术支持信息：

http://www.symantec.com/zh/cn/support/index.jsp

在联系技术支持之前，请确保您的计算机符合产品文档中所列的系统要求。而且您应当坐在发生问题的计算机旁边，以便需要时重现问题。

联系技术支持时，请准备好以下信息：

- 产品版本信息
- 硬件信息
- 可用内存、磁盘空间和 NIC 网卡信息
- 操作系统
- 版本和补丁程序级别
- 网络结构
- 路由器、网关和 IP 地址信息
- 问题说明：
  - 错误消息和日志文件
  - 联系 Symantec 之前执行过的故障排除操作

- 最近所做的软件配置更改和网络更改

授权许可与产品注册

如果您的 Symantec 产品需要注册或许可证密钥，请访问我们的技术支持网页：

https://licensing.symantec.com/

客户服务

可从以下网站获得客户服务信息：

http://www.symantec.com/zh/cn/support/index.jsp

客户服务可帮助您解决一些非技术性问题，例如以下几类问题：

- 有关产品许可或序列号的问题
- 产品注册更新（例如，更改地址或名称）
- 一般产品信息（功能、可用的语言、当地经销商）
- 有关产品更新和升级的最新信息
- 有关升级保障和维护合同的信息
- Symantec 采购计划的相关信息
- 有关 Symantec 技术支持选项的建议
- 非技术性的售前问题
- 与光盘或手册相关的问题

文档

介质中提供了 PDF 格式的产品指南。请确保您使用的是文档的最新版本。每个指南的第 2 页上提供了文档版本信息。从 Symantec 网站可以获取最新的产品文档。

https://sort.symantec.com/documents

您对产品文档的反馈对我们很重要。请发送改进建议和有关错误或疏漏的报告。请在您的报告中包括所报告的文本内容的文档标题和文档版本（位于第二页上）以及章节标题。请将反馈发送到：

doc_feedback@symantec.com

如需最新 HOWTO 文章、文档更新的信息，或询问有关产品文档的问题，请访问 Symantec Connect 上的 Storage and Clustering Documentation（存储和集群文档）论坛。

https://www-secure.symantec.com/connect/storage-management/forums/storage-and-clustering-documentation

关于 Symantec Connect

Symantec Connect 是为 Symantec 企业客户提供的点对点技术社区网站。 参与者可以与其他产品用户联络并共享信息，包括发布论坛帖子、文章、视频、下载、博客和提出建议，并可与 Symantec 产品团队和技术支持进行交流。 内容会由社区进行评分，成员可凭其贡献获得奖励积分。

http://www.symantec.com/connect/storage-management

维护协议资源

如果想就现有维护协议事宜联络 Symantec，请通过以下方式联络您所在地区的维护协议管理部门：

国家/地区	销售热线	电子邮件
中国大陆	800 810 8826	China-Sales@symantec.com
中国台湾	0080 1611 391	Taiwan-Sales@symantec.com
中国香港特别行政区	800 963 421	HongKong-Sales@symantec.com

目录

技术支持 3

第 1 章 了解 DMP 11

关于 Veritas Dynamic Multi-Pathing (DMP) 11
DMP 的工作原理 12
DMP 监视路径 I/O 的方式 15
负载平衡 16
集群环境中的 DMP 17
磁盘阵列的多条路径 17
设备发现 18
磁盘设备 18
DMP 中的磁盘设备命名 19
关于基于操作系统的命名机制 19
关于基于磁盘阵列的命名 19

第 2 章 设置 DMP 以管理本机设备 25

关于设置 DMP 以管理本机设备 25
将 LVM 卷组迁移到 DMP 26
从 EMC PowerPath 迁移到 DMP 27
从 Hitachi Data Link Manager (HDLM) 迁移到 DMP 28
从 Linux Device Mapper Multipath 迁移到 DMP 29
将 Dynamic Multi-Pathing (DMP) 设备与 Oracle 自动存储管理 (ASM) 一起使用 30
支持动态多径处理 (DMP) 设备与 Oracle 自动存储管理 (ASM) 结合使用 30
从 Oracle 自动存储管理 (ASM) 磁盘的列表中删除 Dynamic Multi-Pathing (DMP) 设备 32
将操作系统设备上的 Oracle 自动存储管理 (ASM) 磁盘组迁移到 Dynamic Multi-Pathing (DMP) 设备 32
将 DMP 设备添加到现有的 LVM 卷组或者创建新的 LVM 卷组 36
显示本机多径处理配置 37
为本机设备删除 DMP 支持 38

第 3 章 管理 DMP 39

关于对控制器与存储处理器启用和禁用 I/O 39
关于显示 DMP 数据库信息 40
显示磁盘路径 40
设置 DMP 节点的自定义名称 42
使用 vxdmpadm 管理 DMP 43
检索有关 DMP 节点的信息 44
显示关于 DMP 节点的综合信息 45
显示 LUN 组的成员 47
显示由 DMP 节点、控制器、磁盘阵列或阵列端口控制的路径 47
显示控制器的信息 49
显示磁盘阵列的相关信息 50
显示磁盘阵列端口的信息 51
显示有关 TPD 控制的设备的信息 51
显示扩展设备属性 52
隐藏或包含 VxVM 控制中的设备 54
收集和显示 I/O 统计信息 55
设置磁盘阵列路径的属性 60
显示设备或磁盘阵列的冗余级别 61
指定最小活动路径数 62
显示 I/O 策略 63
指定 I/O 策略 63
对路径、控制器、阵列端口或 DMP 节点禁用 I/O 68
为路径、控制器、阵列端口或 DMP 节点启用 I/O 69
重命名磁盘阵列 70
配置对 I/O 故障的响应 71
配置 I/O 控制机制 72
配置子路径故障转移组 (SFG) 73
配置低影响路径探测 73
显示恢复选项值 74
配置 DMP 路径还原策略 75
停止 DMP 路径还原线程 76
显示 DMP 路径还原线程的状态 76
配置阵列策略模块 77

第 4 章 管理磁盘 79

关于磁盘管理 79
发现和配置新添加的磁盘设备 79
部分发现设备 80
发现磁盘和动态添加磁盘阵列 81
第三方驱动程序共存特性 83
如何管理设备发现层 84

更改磁盘设备的命名机制 95
显示磁盘命名机制 96
重新生成持久性设备名称 97
更改 TPD 控制的磁盘阵列的设备命名 97
发现基于磁盘阵列的磁盘名称与基于操作系统的磁盘名称之间的关联关系 99

第 5 章 动态重新配置设备 101

关于联机动态重新配置 101
联机重新配置受 DMP 控制的 LUN 101
从现有目标 ID 中动态删除 LUN 102
将新的 LUN 动态添加到新的目标 ID 104
动态替换现有目标 ID 中的 LUN 104
从阵列端更改 LUN 的特性 105
联机更换主机总线适配器 106
联机升级阵列控制器固件 107

第 6 章 事件监视 109

关于动态多径处理 (DMP) 事件源后台驻留程序 (vxesd) 109
架构监视和主动错误检测 109
iSCSI 和 SAN 光纤通道技术的动态多径处理 (DMP) 发现 111
DMP 事件日志记录 111
启动和停止动态多径处理 (DMP) 事件源后台驻留程序 111

第 7 章 性能监视和优化 113

关于使用模板调整 Veritas Dynamic Multi-Pathing (DMP) 113
DMP 调整模板 114
DMP 调整模板示例 115
通过配置属性模板调整 DMP 主机 118
管理 DMP 配置文件 119
将 DMP 可调参数和属性重置为默认值 119
模板支持的 DMP 可调参数和属性 120
DMP 可调参数 120

术语表 127

索引 135

1

了解 DMP

本章节包括下列主题：

- 关于 Veritas Dynamic Multi-Pathing (DMP)
- DMP 的工作原理
- 磁盘阵列的多条路径
- 设备发现
- 磁盘设备
- DMP 中的磁盘设备命名

关于 Veritas Dynamic Multi-Pathing (DMP)

Veritas Dynamic Multi-Pathing (DMP) 为在系统上配置的操作系统本机设备提供了多径处理功能。DMP 创建 DMP 元设备（也称为 DMP 节点）来表示指向同一物理 LUN 的所有设备路径。

DMP 也可作为单独产品使用，它扩展了 DMP 元设备，可支持 OS 本机逻辑卷管理器 (LVM)。可以在 DMP 元设备上创建 LVM 卷和卷组。

Veritas Dynamic Multi-Pathing 可以通过 Storage Foundation 产品单独授权。Veritas Volume Manager 和 Veritas File System 功能未附带 DMP 许可证。

DMP 功能附带 Storage Foundation (SF) Enterprise 许可证、SF HA Enterprise 许可证和 Storage Foundation 标准许可证。

Veritas Volume Manager (VxVM) 卷和磁盘组可以与 LVM 卷和卷组共存，但是每个设备只能支持其中一种类型。如果磁盘具有 VxVM 标签，则该磁盘对 LVM 不可用。同样，如果磁盘正由 LVM 使用，则该磁盘对 VxVM 不可用。

DMP 的工作原理

Veritas Dynamic Multi-Pathing (DMP) 通过使用路径故障转移和负载平衡来提供更高的可用性、可靠性和性能。该功能可用于不同供应商的多端口磁盘阵列。

磁盘阵列可以通过多个路径连接到主机系统。为了检测到磁盘的各条路径，DMP 使用一种特定于每个受支持的阵列的机制。DMP 还可以区分连接到同一主机系统的某个受支持阵列的不同磁盘阵列。

请参见第 79 页的“发现和配置新添加的磁盘设备”。

DMP 使用哪种多径处理策略取决于磁盘阵列的特性。

DMP 支持下列标准阵列类型：

主动/主动 (A/A)	允许将多条路径并行用于 I/O。这种阵列通过在到 LUN 的多条路径之间平均分配 I/O 负载，使 DMP 能够提供更大的 I/O 吞吐量。如果一条路径出现故障，DMP 自动将 I/O 路由到其他可用的路径。
非对称主动/主动 (A/A-A)	A/A-A 或非对称主动/主动式阵列可以通过辅助存储路径来访问，而性能几乎不会下降。行为与 ALUA 类似，但该类阵列不支持 ALUA 阵列所支持的 SCSI 命令。
非对称逻辑单元访问 (ALUA)	DMP 支持 ALUA 的所有变体。
主动/被动 (A/P)	在正常操作期间允许通过单个控制器（也称为访问端口或存储处理器）上的主（主动）路径访问其 LUN（逻辑单元；真实磁盘或用硬件创建的虚拟磁盘）。 在隐式故障转移模式（或自动侵入模式）下，如果主路径出现故障，A/P 阵列自动使用故障转移功能将 I/O 安排到不同控制器上的次级（被动）路径。除非主动端口出现故障，否则 I/O 不使用被动端口。在 A/P 阵列中，如果在主路径上出现 I/O 故障，就会为单个 LUN 执行路径的故障转移。 此策略通过指向某个控制器的多条主路径来支持并行 I/O 和负载平衡。此功能是由具有多个端口的控制器提供的，或是通过在阵列和控制器之间插入 SAN 交换机提供的。只有在所有的主动主路径都出现故障时才故障转移到次级（被动）路径上。

显式故障转移模式或非自动侵入模式 (A/P-F) 下的主动/被动阵列	必须对阵列使用相应命令以使 LUN 故障转移到次级路径。 此策略通过指向某个控制器的多条主路径来支持并行 I/O 和负载平衡。此功能是由具有多个端口的控制器提供的，或是通过在阵列和控制器之间插入 SAN 交换机提供的。只有在所有的主动主路径都出现故障时才故障转移到次级（被动）路径上。
具备 LUN 组故障转移 (A/P-G) 特性的主动/被动阵列	对于具备 LUN 组故障转移特性的主动/被动阵列（A/P-G 阵列）中，通过一个控制器连接的一组 LUN 被当作一个故障转移实体对待。与 A/P 阵列不同，这种阵列的故障转移发生在控制器级别，而不是针对单个 LUN 进行。主控制器和次级控制器各自连接到不同的 LUN 组。如果主控制器的 LUN 组有一个 LUN 发生故障，则该组中所有的 LUN 均故障转移到次级控制器上。 此策略通过指向某个控制器的多条主路径来支持并行 I/O 和负载平衡。此功能是由具有多个端口的控制器提供的，或是通过在阵列和控制器之间插入 SAN 交换机提供的。只有在所有的主动主路径都出现故障时才故障转移到次级（被动）路径上。

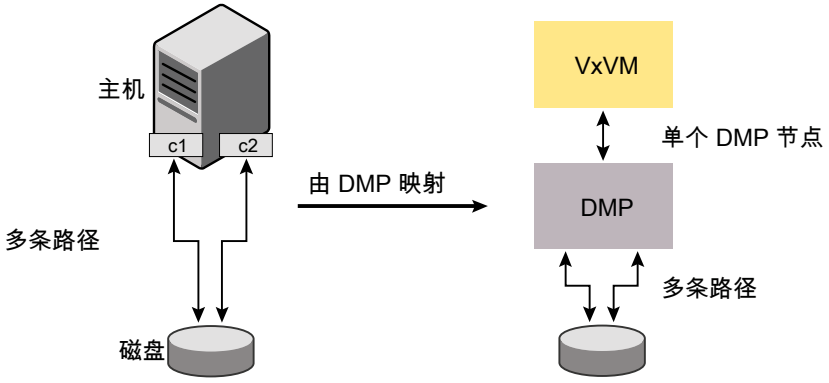
阵列策略模块 (APM) 可以为 DMP 定义除 DMP 支持的标准阵列类型以外的其他阵列类型。

Veritas Dynamic Multi-Pathing 使用 DMP 元节点（DMP 节点）访问连接到系统的磁盘设备。对于受支持的阵列中的每个磁盘，DMP 将一个节点映射到该磁盘连接的一组路径。另外，DMP 将磁盘阵列的相应多径处理策略与该节点关联。

对于不受支持的阵列中的磁盘，DMP 将单独的节点映射到与磁盘连接的每条路径。节点的原始设备和块设备是在目录 `/dev/vx/rdmp` 和 `/dev/vx/dmp` 中分别创建的。

图 1-1 显示 DMP 如何为支持的磁盘阵列的磁盘建立节点。

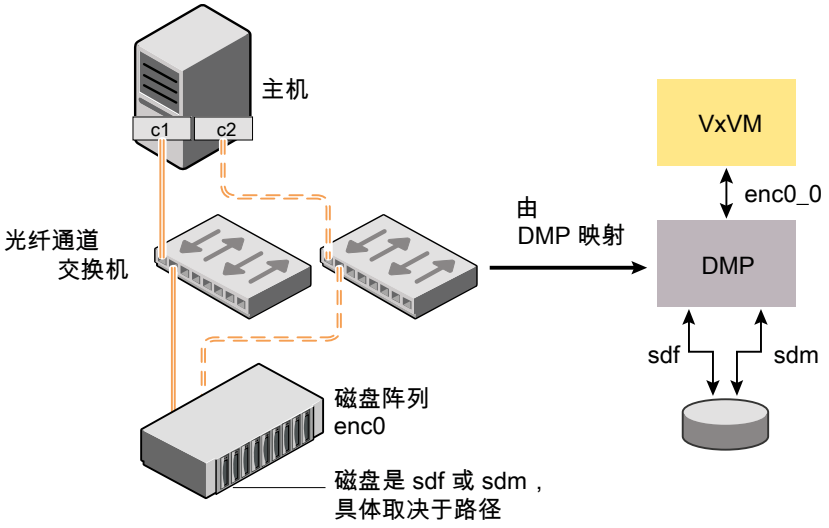
图 1-1 DMP 如何将磁盘的多条物理路径表示为一个节点



DMP 采用一种便于识别磁盘所属阵列的磁盘设备命名机制。

图 1-2 显示了一个存在于磁盘阵列的单个磁盘上、拥有两条路径 sdf 与 sdm，但 VxVM 使用单个 DMP 节点 enc0_0 进行访问的示例。

图 1-2 SAN 环境中的磁盘阵列多径处理示例



请参见第 19 页的“关于基于磁盘阵列的命名”。

请参见第 95 页的“更改磁盘设备的命名机制”。

请参见第 79 页的“发现和配置新添加的磁盘设备”。

DMP 监视路径 I/O 的方式

在 5.0 版之前的 VxVM 中，DMP 有一个进行错误处理的内核后台驻留程序 (`errord`) 以及另一个执行路径还原活动的内核后台驻留程序 (`restored`)。

从版本 5.0 起，DMP 提供一个内核线程池，用来执行错误处理、路径还原、统计信息收集以及 SCSI 请求回叫等任务。`vxdmpadm gettune` 命令可以用来提供线程相关信息。保留名称 `restored` 以便实现向后兼容性。

一个内核线程通过激活与路径对应的主机总线适配器的侦测来响应路径上的 I/O 故障。另一个线程然后根据 HBA 的响应再采取相应的措施。措施包括重试路径上的 I/O 请求；或者放弃该路径，重新安排备用路径上的 I/O。

还原内核任务将被定期唤醒（通常是间隔 5 分钟），以检查路径的运行状况以及恢复被还原的 I/O 路径。由于某些路径会出现间歇性的故障，因此请仅恢复在给定时间段内（默认情况下为 5 分钟）依然处于正常状态的路径上的 I/O。可以为 DMP 配置不同的路径检查策略。

请参见第 75 页的“配置 DMP 路径还原策略”。

收集统计信息的任务记录每个 I/O 请求的起止时间以及每个路径上的 I/O 故障数量以及重试次数。可以对 DMP 进行配置，使用该信息来防止 SCSI 驱动程序被 I/O 请求淹没。该功能称为 I/O 控制。

如果某个 I/O 请求与镜像卷相关，VxVM 就会指定 FAILFAST 标志。此时，DMP 并不会重试该路径上出现故障的 I/O 请求，而是把该路径上的磁盘标记为已经出现故障。

请参见第 15 页的“路径的故障转移机制”。

请参见第 16 页的“I/O 控制”。

路径的故障转移机制

DMP 在与具有多个路径的磁盘阵列结合使用时，可提高系统可用性。如果到磁盘阵列的某个路径丢失，无须管理员干预，DMP 就会自动为 I/O 请求选择下一条可用路径。

在修复或还原连接时，或者在完全引导系统后添加或删除设备时（假定操作系统能够正确识别设备），DMP 也会得到通知。

如果需要，可以将 DMP 对一条路径上 I/O 故障的响应调节为针对单个磁盘阵列的多条路径。可以将 DMP 配置为满足以下条件后停止 I/O 请求：给定时间后没有后续请求或者在路径上尝试给定次数后都告失败。

请参见第 71 页的“配置对 I/O 故障的响应”。

子路径故障转移组 (SFG)

子路径故障转移组 (SFG) 表示可能会一起出现故障并还原的一组路径。在 SFG 中某个路径上遇到I/O错误时，DMP还会在该SFG的其他路径上执行主动式路径探查。此行为会极大地提高路径故障转移的性能，从而提高I/O性能。当前，DMP按以下规范形成子路径故障转移组：将相同端点内的从主机到阵列的路径绑定到一个逻辑存储故障转移组中。

请参见第 73 页的“配置子路径故障转移组 (SFG)”。

低影响路径探测 (LIPP)

DMP 中的还原后台驻留程序会定期保持探测 LUN 路径。此行为可帮助 DMP 使路径状态始终保持最新，即使路径中不存在 I/O 活动也是如此。还原后台驻留程序更新路径状态时，低影响路径探测会向还原后台驻留程序添加逻辑以优化执行的探测的数量。在逻辑子路径故障转移组的帮助下可实现此优化。添加LIPP逻辑后，DMP会仅探查子路径故障转移组 (SFG) 内的有限数量的路径，而不是探查 SFG 中的所有路径。基于这些探测结果，DMP 可确定该 SFG 中所有路径的状态。

请参见第 73 页的“配置低影响路径探测”。

I/O 控制

如果启用了 I/O 控制，响应迟缓的路径上等待处理的 I/O 请求就会越积越多，可以对 DMP 进行配置以便在满足如下条件时禁止向此路径发送新的 I/O 请求：等待处理的I/O请求数量达到给定的值或者给定时间内该路径上均未出现成功的I/O请求。对路径应用控制后，该路径上的新I/O请求将被安排到其他可用路径上。如果 HBA 不再报告该路径存在任何错误或者该路径上等待处理的一个 I/O 请求成功，控制就被解除。

请参见第 72 页的“配置 I/O 控制机制”。

负载平衡

默认情况下，Veritas Dynamic Multi-Pathing (DMP) 使用最小队列 I/O 策略在主动/主动 (A/A)、主动/被动 (A/P)、具有显式故障转移功能的主动/被动 (A/P-F) 以及具有组故障转移功能的主动/被动(A/P-G)磁盘阵列的路径之间进行负载平衡。负载平衡通过使用所有可用路径的总带宽来提高 I/O 吞吐量。 沿未完成的 I/O 最小的路径发送 I/O。

对于 A/P 磁盘阵列，沿主路径发送 I/O。如果所有主路径都出故障，I/O 将切换到可用的辅助路径。由于将 LUN 的所有权不断地从一个控制器转给另一个控制器会导致 I/O 速度严重下降，因此不对 A/P 磁盘阵列执行主路径和次级路径之间的负载平衡，除非它们支持并行 I/O。

对于 A/P、A/P-F 和 A/P-G 阵列，负载平衡在当前所有的主动路径之间进行，与 A/A 磁盘阵列一样。

您可以更改通往磁盘阵列的路径的 I/O 策略。

请参见第 63 页的“指定 I/O 策略”。

集群环境中的 DMP

注意：要使用 Veritas Volume Manager (VxVM) 的集群功能，您需要有附加许可证。仅 VxVM 支持集群功能。

在由多台主机共享一些主动/被动 (A/P) 类型的磁盘阵列的集群环境中，集群中的所有节点必须通过同一个物理存储控制器端口来访问磁盘。通过多条路径同时访问磁盘会严重降低 I/O 性能（有时称为乒乓效应）。单个集群节点上的路径故障转移也在集群上进行协调，以使所有节点继续共享同一物理路径。

在 VxVM 4.1 版之前，集群和 DMP 功能不会在路径还原时处理 A/P 阵列中的自动故障回复，并且不支持显式故障转移模式阵列的故障回复。只能在纠正路径故障之后通过在各个集群节点上运行 `vxdctl enable` 命令来手动执行故障回复。在 4.1 版中，故障回复现在是由主节点协调的一项集群范围内的自动操作。另外，显式故障转移模式阵列的自动故障回复也是通过发出相应的低级命令进行处理的。

注意：支持 A/P 阵列的自动故障回复需要在系统上安装适当的阵列支持库 (ASL)。也可能需要阵列策略模块 (APM)。

请参见第 81 页的“发现磁盘和动态添加磁盘阵列”。

对于主动/主动类型的磁盘阵列，任何磁盘都可以通过其所有可用物理路径同时访问。在集群环境中，节点不需要通过同一物理路径访问一个磁盘。

请参见第 84 页的“如何管理设备发现层”。

请参见第 77 页的“配置阵列策略模块”。

关于启用或禁用使用共享磁盘组的控制器

在 5.0 版之前，Veritas Volume Manager (VxVM) 不允许启用或禁用与属于共享 Veritas Volume Manager 磁盘组的磁盘相连的路径或控制器。从 VxVM 5.0 起，支持在集群的共享 DMP 节点上执行此类操作。

磁盘阵列的多条路径

有些磁盘阵列提供用于访问其磁盘设备的多个端口。 这些端口与主机总线适配器 (HBA) 控制器以及阵列的任何本地数据总线或 I/O 处理器相结合，组成了多个用于访问磁盘设备的硬件路径。这种磁盘阵列称为多径处理磁盘阵列。此类型的磁盘阵列可以通过许多不同的配置（如连接到单个主机上不同控制器的多个端口、通过主

机上的单个控制器形成的端口链，或同时连接到不同主机的端口）连接到主机系统上。

请参见第 12 页的“DMP 的工作原理”。

设备发现

设备发现是用来描述发现挂接到主机上的磁盘这一过程的术语。 此功能对于 DMP 很重要，因为它需要支持越来越多的来自很多供应商的磁盘阵列。 通过将设备发现服务与发现主机挂接的设备的功能结合使用，您可以增加对新磁盘阵列的支持。设备发现使用一个称作“设备发现层 (DDL)”的工具。

借助 DDL，您可以增加对新磁盘阵列的支持，而无需重新启动。

请参见第 84 页的“如何管理设备发现层”。

磁盘设备

设备名称（有时称为 devname 或磁盘访问名称）用于定义操作系统已知的磁盘设备的名称。

这类设备通常（但并非总是）位于 `/dev` 目录中。某些供应商的硬件专用的设备可能使用自己的路径名称约定。

VxVM 支持由操作系统提供的磁盘分区机制。设备名称的语法为 `hdx[N]` 或 `sdx[N]`，其中 *x* 是一个字母，用于指示操作系统可查看到的 EIDE (`hd`) 或 SCSI (`hd`) 磁盘的顺序，*N* 是范围 1 到 15 中可选的分区编号。设备名称的示例为 `sda7`，它引用了第一个 SCSI 磁盘上的第 7 个分区。如果省略分区编号，则设备名称表示的是整个磁盘。

如果设备是特定于某些供应商提供的硬件，可能会具有不同的路径名称。例如，COMPAQ SMART 和 SMARTII 控制器使用 `/dev/ida/cXdXpX` 和 `/dev/cciss/cXdXpX` 形式的设备名称。

Dynamic Multi-Pathing (DMP) 使用设备名称在 `/dev/vx/[r]dmp` 目录下创建元设备。DMP 使用元设备（或 DMP 节点）来表示可通过一条或多条物理路径（可能通过不同的控制器）访问的磁盘。可用的访问路径数取决于磁盘是单个磁盘，还是属于系统连接的多端口磁盘阵列。

可以使用 `vxdisk` 实用程序显示 DMP 元设备包含的路径，并显示每条路径的状态（如，启用还是禁用）。

请参见第 12 页的“DMP 的工作原理”。

设备名称还将重映射为基于磁盘阵列的名称。

请参见第 19 页的“DMP 中的磁盘设备命名”。

DMP 中的磁盘设备命名

磁盘的设备名称根据指定给 DMP 的命名机制进行分配。对于不同的磁盘类别，设备名称的格式可能会有所不同。

请参见第 82 页的“磁盘类别”。

设备名称可以使用下列命名机制之一：

- 基于操作系统的命名。
  请参见第 19 页的“关于基于操作系统的命名机制”。
- 基于磁盘阵列的命名。
  请参见第 19 页的“关于基于磁盘阵列的命名”。

设备名称长度超过 31 个字符的设备始终使用基于磁盘阵列的名称。

默认情况下，DMP 使用基于磁盘阵列的命名。可根据需要更改磁盘设备命名机制。

请参见第 95 页的“更改磁盘设备的命名机制”。

关于基于操作系统的命名机制

在基于操作系统的命名机制中，所有磁盘设备都使用 `hdx[N]` 格式或 `sdx[N]` 格式进行命名，其中 *x* 是一个字母，用于指示操作系统可看到的 EIDE (`hd`) 或 SCSI (`sd`) 磁盘的顺序， *N* 是一个可选的分区编号，在 1 到 15 的范围之内。

DMP 基于多条指向磁盘的路径来指派 DMP 元设备的名称（磁盘访问名称）。DMP 按字母顺序对名称排序，并选择第一个名称。例如，选择 sdd 而不是 sdc。此行为更便于将设备与底层存储关联。

如果 CVM 集群是对称的，则集群中每个节点都访问同一磁盘集。此命名机制使对称集群中各节点上的命名一致。

默认情况下，基于 OS 的名称不会永久保持不变，在系统配置对操作系统可识别的设备名称进行更改时会重新生成。如果不希望基于 OS 的名称在重新启动后发生更改，请为命名机制设置 persistence 属性。

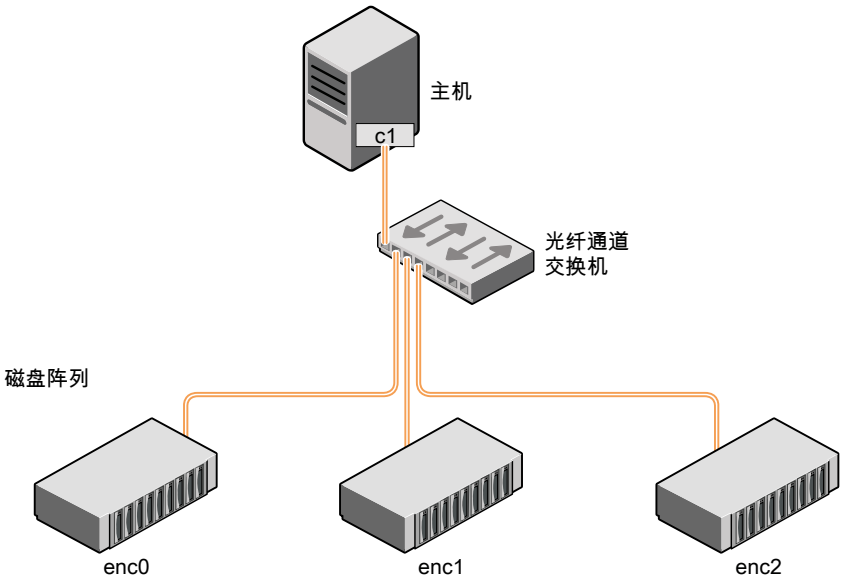
请参见第 95 页的“更改磁盘设备的命名机制”。

关于基于磁盘阵列的命名

基于磁盘阵列的命名机制为基于操作系统的设备命名机制提供了另一种选择。在使用光纤通道交换机的存储区域网络 (SAN) 中，由操作系统提供的有关磁盘位置的信息可能无法正确指出磁盘的物理位置。基于磁盘阵列的命名允许 DMP 将磁盘阵列作为单独的物理实体来进行访问。通过在单独的磁盘阵列上配置数据的冗余副本，可以在一个或多个磁盘阵列发生故障时确保数据的安全。

图 1-3 显示了一个典型的 SAN 环境，其中主机控制器通过光纤通道交换机连接到多个磁盘阵列。

图 1-3 通过光纤通道交换机连接的磁盘阵列配置的示例



在这样的配置中，可以使用基于磁盘阵列的命名机制来指示磁盘阵列中的每个磁盘。例如，磁盘阵列 enc0 中磁盘的设备被命名为 enc0_0、enc0_1 等。使用该机制的一个主要的优点是，可以迅速确定某个磁盘在大型 SAN 配置中的物理位置。

在大多数磁盘阵列中，可以使用基于硬件的存储管理功能，将几个物理磁盘作为一个 LUN 提供给操作系统。在这种情况下，VxVM 也会看到单一逻辑磁盘设备，而不是其组成磁盘。因此，当引用某个磁盘阵列中的一个磁盘时，该磁盘可能是物理设备，也可能是 LUN。

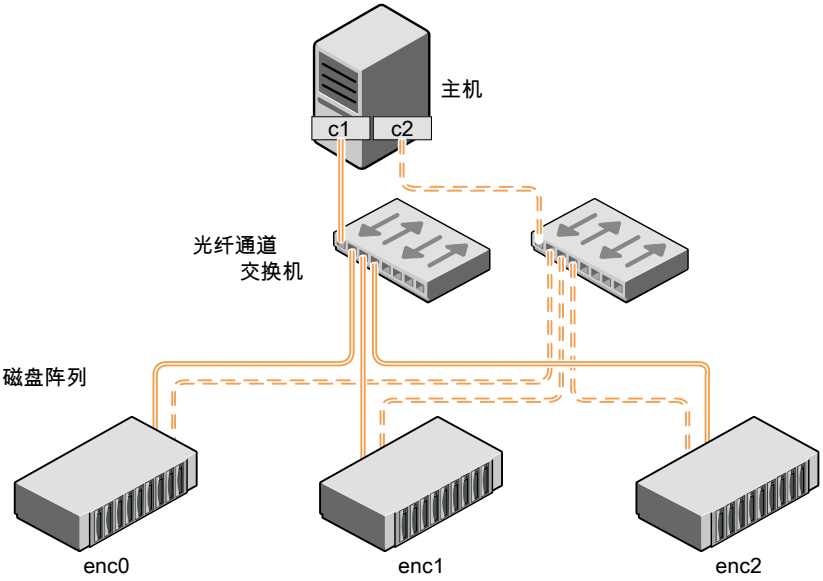
基于磁盘阵列的命名的另一个重要优点是，它使 VxVM 能够 避免在同一磁盘阵列中放置冗余的数据副本。之所以称为优点是因为：这样可以将每个磁盘阵列视为单独的故障域。例如，如果只在磁盘阵列 enc1 的磁盘上配置了镜像卷，则交换机与磁盘阵列之间的电缆故障可能导致整个卷不可用。

如果需要，可以用一个对配置更有意义的名称代替 DMP 分配给磁盘阵列的默认名称。

请参见第 70 页的“重命名磁盘阵列”。

图 1-4 显示了一个高可用性(HA)配置，在该配置中，通过将主机上的独立控制器连接到具有独立磁盘阵列路径的各个交换机，可以实现对存储器的冗余循环访问。

图 1-4 使用多个交换机提供冗余循环访问的 HA 配置的示例



这种配置可防范其中一个主机控制器（c1 与 c2）发生故障，或主机与任一交换机之间的电缆发生故障。在该示例中，对于所有可用来访问磁盘的路径，VxVM 都使用同一名称来标识每个磁盘。例如，磁盘设备 enc0_0 表示单个磁盘，对于该磁盘，操作系统使用两条不同的路径，如 sdf 与 sdm。

请参见第 19 页的“DMP 中的磁盘设备命名”。

请参见第 95 页的“更改磁盘设备的命名机制”。

要在配置数据冗余时将故障域纳入考虑范围，可以控制在磁盘阵列间布置镜像卷的方式。

基于磁盘阵列的命名摘要

默认情况下，DMP 使用基于磁盘阵列的命名。

基于磁盘阵列的命名的工作方式如下所示：

- 受支持的磁盘阵列中的所有架构式磁盘或非架构式磁盘都使用 *enclosure_name_#* 格式进行命名。例如，受支持的磁盘阵列 enggdept 中的磁盘被命名为 enggdept_0、enggdept_1、enggdept_2 等。
  可以使用 vxdmpadm 命令来管理磁盘阵列名称。
  请参见第 70 页的“重命名磁盘阵列”。
  请参见 vxdmpadm(1M) 手册页。

- DISKS 类别中的磁盘（JBOD 磁盘）以 Disk_# 格式命名。
- 磁盘分区通过在名称后面追加 *s#* 来表示，其中 # 分区编号。例如，Disk_0s5 和 Disk_0s6 指示分片磁盘 Disk_0 的专用区域和公共区域所使用的扩展分区。ACME_0s5 表示简单磁盘 ACME_0 的扩展分区。对于 CDS 磁盘，分区 3 同时用于专用区域和公共区域。
- OTHER_DISKS 类别中的磁盘（未经过 DMP 多径处理的磁盘）以 hdx[*N*] 格式或 sdx[*N*] 格式进行命名。
- 已封装的根磁盘始终使用 hdx[*N*] 格式或 sdx[*N*] 格式进行命名。

默认情况下，基于磁盘阵列的名称为永久的，因此重新启动后不会更改。

如果 CVM 集群是对称的，则集群中每个节点都访问同一磁盘集。基于磁盘阵列的名称提供了一个一致的命名系统，因此设备名称在每个节点上都是相同的。

要显示 DMP 磁盘（如 mydg01）的本机 OS 设备名称，请使用以下命令：

```
# vxdisk path | grep diskname
```

请参见第 70 页的“重命名磁盘阵列”。

请参见第 82 页的“磁盘类别”。

使用阵列卷标识符 (AVID) 的基于磁盘阵列的命名机制

默认情况下，DMP 使用称为阵列卷 ID (AVID) 的阵列特定属性将基于磁盘阵列的名称分配给 DMP 元设备。AVID 为由阵列提供的 LUN 提供唯一标识符。 与该阵列对应的 ASL 则提供 AVID 属性。 在磁盘阵列内，DMP 在 DMP 元节点名称中使用阵列卷标识符 (AVID) 作为索引。 DMP 元节点名称采用格式 *enclosureID_AVID*。

随着 AVID 在 EBN 命名机制中的引入，标识存储设备变得容易多了。阵列卷标识符 (AVID) 使您可以对连接到同一存储的多个节点采用一致的设备命名机制。磁盘访问名称永不改变，因为它基于阵列本身定义的名称。

注意：DMP 不支持将 AVID 与 PowerPath 名称一起使用。

如果 DMP 无法获得设备的 AVID，则会检索称为 LUN 序列号的另一个唯一的 LUN 标识符。DMP 根据 LUN 序列号 (LSN) 对设备进行排序，然后分配索引编号。所有主机看到的是同一组设备，因此所有主机将有排序相同的列表，从而导致设备索引在集群中是一致的。这种情况下，DMP 元节点名称采用格式 *enclosureID_index*。

DMP 也支持一个可伸缩的框架，该框架使您可以应用将自定义名称与机柜和 LUN 序列号相关联的设备命名文件，从而充分地在主机上自定义设备名称。

如果 CVM 集群是对称的，则集群中每个节点都访问同一磁盘集。 基于磁盘阵列的名称提供了一个一致的命名系统，因此设备名称在每个节点上都是相同的。

可使用DMP实用程序（如`vxdisk list`）来显示DMP元节点名称，包括显示AVID属性。使用AVID可将DMP元节点名称关联到阵列管理界面（GUI或CLI）中显示的LUN。

例如，在EMC CX阵列上，若磁盘阵列为emc_clariion0且ASL提供的阵列卷ID为91，则DMP元节点名称是emc_clariion0_91。以下示例输出显示了DMP元节点名称：

```
$ vxdisk list
emc_clariion0_91  auto:cdsdisk  emc_clariion0_91   dg1  online shared
emc_clariion0_92  auto:cdsdisk  emc_clariion0_92   dg1  online shared
emc_clariion0_93  auto:cdsdisk  emc_clariion0_93   dg1  online shared
emc_clariion0_282 auto:cdsdisk  emc_clariion0_282  dg1  online shared
emc_clariion0_283 auto:cdsdisk  emc_clariion0_283  dg1  online shared
emc_clariion0_284 auto:cdsdisk  emc_clariion0_284  dg1  online shared

# vxddladm get namingscheme
NAMING_SCHEME       PERSISTENCE    LOWERCASE      USE_AVID
============================================================
Enclosure Based     Yes            Yes            Yes
```

2

设置 DMP 以管理本机设备

本章节包括下列主题：

- 关于设置 DMP 以管理本机设备
- 将 LVM 卷组迁移到 DMP
- 从 EMC PowerPath 迁移到 DMP
- 从 Hitachi Data Link Manager (HDLM) 迁移到 DMP
- 从 Linux Device Mapper Multipath 迁移到 DMP
- 将 Dynamic Multi-Pathing (DMP) 设备与 Oracle 自动存储管理 (ASM) 一起使用
- 将 DMP 设备添加到现有的 LVM 卷组或者创建新的 LVM 卷组
- 显示本机多径处理配置
- 为本机设备删除 DMP 支持

关于设置 DMP 以管理本机设备

可以使用 DMP 而不是第三方驱动程序进行高级存储管理。本节介绍了如何设置 DMP 以管理本机 LVM 设备和在这些设备上运行的任何逻辑卷。

在安装 DMP 后，设置 DMP 以用于 LVM。要设置 DMP 以用于 LVM，请打开 dmp_native_support 可调参数。打开此可调参数后，DMP 将在没有 VxVM 标签且不受任何第三方多径处理(TPD)软件控制的任意设备上启用对 LVM 的支持。此外，打开 dmp_native_support 可调参数会将未使用的任何 LVM 卷组迁移到 DMP 设备上。

dmp_native_support 可调参数为 LVM 启用 DMP 支持，如下所示：

LVM 卷组	如果未使用 LVM 卷组，则打开本机支持会将卷组迁移到 DMP 设备。 如果正在使用 LVM 卷组，则执行关闭卷组和将卷组迁移到 DMP 的步骤。
Veritas Volume Manager (VxVM) 设备	不为具有 VxVM 标签的任何设备启用本机支持。要使设备可用于 LVM，请删除 VxVM 标签。 VxVM 设备可以在 DMP 控制下与本机设备共存。
使用第三方驱动程序 (TPD) 进行多径处理的设备	如果磁盘已经使用第三方驱动程序 (TPD) 进行多径处理，那么，除非删除 TPD 支持，否则 DMP 不管理该设备。在删除 TPD 支持后，打开 `dmp_native_support` 可调参数以迁移设备。 如果已通过 TPD 设备构建 LVM 卷组，则执行将 LVM 卷组迁移到 DMP 设备的步骤。

要打开 dmp_native_support 可调参数，请使用以下命令：

```
# vxdmpadm settune dmp_native_support=on
```

首次执行此操作时，该命令会报告是否正在使用卷组，但不迁移这些卷组。要将卷组迁移到 DMP 上，请停止该卷组，然后再次执行 `vxdmpadm settune` 命令以将卷组迁移到 DMP 上。

如果无法禁用卷（例如由 `/var` 或 `/home` 文件系统使用的卷），则重新启动应当会将卷组置于 DMP 设备上。

要验证 dmp_native_support 可调参数的值，请使用以下命令：

```
# vxdmpadm gettune dmp_native_support
Tunable                     Current Value   Default Value
--------------------------  --------------  ---------------
dmp_native_support          on              off
```

将 LVM 卷组迁移到 DMP

可以使用 DMP 而不是第三方驱动程序进行高级存储管理。本节介绍了如何设置 DMP 以管理 LVM 卷组和运行在其上的文件系统。

要设置 DMP，请将设备从现有的第三方设备驱动程序迁移到 DMP。

表 2-1 显示了支持的本机解决方案和迁移路径。

表 2-1 支持的迁移路径

操作系统	本机解决方案	迁移过程
Linux	EMC PowerPath	请参见第 27 页的“从 EMC PowerPath 迁移到 DMP”。
Linux	Hitachi Data Link Manager (HDLM)	请参见第 28 页的“从 Hitachi Data Link Manager (HDLM) 迁移到 DMP”。
Linux	Linux Device Mapper Multipath	请参见第 29 页的“从 Linux Device Mapper Multipath 迁移到 DMP”。

从 EMC PowerPath 迁移到 DMP

此过程介绍了如何使设备脱离 EMC PowerPath 的控制并在设备上启用 DMP。

规划下列过程的系统停机时间。

迁移步骤涉及到主机的系统停机时间，原因如下：

- 需要停止应用程序
- 如果使用 VCS，需要停止 VCS 服务

使设备脱离 EMC PowerPath 的控制并启用 DMP

1 停止使用 PowerPath 元设备的应用程序。

   在 VCS 环境中，停止应用程序的 VCS 服务组，此时将停止该应用程序。

2 卸载使用 PowerPath 设备上的卷组的任何文件系统。

3 停止使用 PowerPath 设备的 LVM 卷组。

```
# lvchange -a n lvpath
```

4 从 VxVM 中删除 PowerPath 设备的磁盘访问名称。

```
# vxdisk rm emcpowerXXXX
```

   其中： *emcpowerXXXX* 是设备的名称。

5 使设备脱离 PowerPath 的控制：

```
# powermt unmanage dev=pp_device_name
or
# powermt unmanage class=array_class
```

6 验证 PowerPath 设备是否已脱离了 PowerPath 的控制。

```
# powermt display dev=all
```

7 运行设备扫描使设备处于 DMP 的控制之下：

```
# vxdisk scandisks
```

8 为 LVM 卷组启用 DMP 支持。

```
# vxdmpadm settune dmp_native_support=on
```

9 装入文件系统。

10 重新启动应用程序。

从 Hitachi Data Link Manager (HDLM) 迁移到 DMP

此过程介绍了如何使设备脱离 HDLM 的控制并在设备上启用 DMP。

注意： DMP 不能与 HDLM 共存；必须从系统中删除 HDLM。

规划下列过程的系统停机时间。

迁移步骤涉及到主机的系统停机时间，原因如下：

- 需要停止应用程序
- 如果使用 VCS，需要停止 VCS 服务
- 这个过程涉及到一次或多次主机重新启动

使设备脱离 Hitachi Data Link Manager (HDLM) 的控制并启用 DMP

1 停止使用 HDLM 元设备的应用程序

2 卸载使用 HDLM 设备上的卷组的任何文件系统。

3 停止使用 HDLM 设备的 LVM 卷组。

```
# lvchange -a n lvpath
```

4 卸载 HDLM 软件包。

5 为 LVM 卷组启用 DMP 支持。

```
# vxdmpadm settune dmp_native_support=on
```

上述命令还会启用 DMP 根支持。

6 重新启动系统。

7 在重新启动后，DMP将控制设备。如果HDLM设备上存在任何LVM卷组，则将它们迁移到DMP设备上。

8 装入文件系统。

9 重新启动应用程序。

从 Linux Device Mapper Multipath 迁移到 DMP

此过程介绍了如何使设备脱离 Linux Device Mapper Multipath 的控制并在设备上启用 DMP。

规划下列过程的系统停机时间。

迁移步骤涉及到主机的系统停机时间，原因如下：

- 需要停止应用程序
- 如果使用 VCS，需要停止 VCS 服务
- 这个过程涉及到一次或多次主机重新启动

使设备脱离 Device Mapper Multipath 的控制并启用 DMP

1 停止使用 Device Mapper Multipath 设备的应用程序。

2 卸载使用 Device Mapper Multipath 设备的所有文件系统。

3 禁用 Device Mapper Multipath 设备上的所有卷。

```
# lvchange -a n lvname
```

4 更新 `/etc/multipath.conf` 文件以将所有 Device Mapper 设备列入黑名单。此步骤将为所有设备禁用多径处理功能。

```
# Blacklist all devices by default.
blacklist {
        devnode "*"
}
```

5 重新启动 `multipathd` 以删除所有的 `/dev/dm-*` 和 `/dev/mpath/*` 设备条目。

```
# service multipathd restart
```

6 停止 `multipathd` 后台驻留程序。

```
# service multipathd stop
```

7 禁止 `multipathd` 在重新启动后启动。

```
# chkconfig multipathd off
```

8 为 LVM 卷组启用 DMP 支持。

```
# vxdmpadm settune dmp_native_support=on
```

9 运行设备扫描使设备处于 DMP 的控制之下：

```
# vxdisk scandisks
```

10 装入文件系统。

11 重新启动应用程序。

将 Dynamic Multi-Pathing (DMP) 设备与 Oracle 自动存储管理 (ASM) 一起使用

此版本的 DMP 支持将 DMP 设备与 Oracle 自动存储管理 (ASM) 一起使用。DMP 支持下列操作：

- 请参见第 30 页的“支持动态多径处理 (DMP) 设备与 Oracle 自动存储管理 (ASM) 结合使用”。
- 请参见第 32 页的“从 Oracle 自动存储管理 (ASM) 磁盘的列表中删除 Dynamic Multi-Pathing (DMP) 设备”。
- 请参见第 32 页的“将操作系统设备上的 Oracle 自动存储管理 (ASM) 磁盘组迁移到 Dynamic Multi-Pathing (DMP) 设备”。

支持动态多径处理 (DMP) 设备与 Oracle 自动存储管理 (ASM) 结合使用

启用对 Oracle 自动存储管理 (ASM) 的 DMP 支持，这可以使 DMP 设备对 ASM 显示为可用磁盘。对 ASM 的 DMP 支持可用于字符设备 (`/dev/vx/rdmp/*`) 和块设备 (`/dev/vx/dmp`)。

对原始的（字符）设备 (`/dev/raw/*`) 的支持也可用。 Oracle ASM 要求 Voting 磁盘使用与 DMP 设备对应的 `/dev/raw/*` 设备条目。数据磁盘可以使用 DMP 块设备条目 `/dev/vx/dmp/*`

使 DMP 设备对 ASM 可见

1 在 ASM 中，确保 ASM_DISKSTRING 设置为正确的值。

该值可以为 /dev/vx/dmp/*、/dev/vx/rdmp/* 或 /dev/raw/*。

例如：

```
SQL> show parameter ASM_DISKSTRING;
 NAME                 TYPE        VALUE
 -------------------- ----------- ---------------
 asm_diskstring       string      /dev/vx/dmp/*
```

2 以 root 用户身份启用 DMP 设备以用于 ASM。

```
# vxdmpraw enable usernamegroupnamemode [devicename ...]
```

其中： *username* 表示运行 ASM 实例的 ASM 用户， *groupname* 表示指定的用户 ID 的 UNIX/Linux 组名称， *mode* 表示设备上设置的权限。如果您指定一个或多个 *devicenames*，将为这些设备启用对 ASM 的 DMP 支持。如果您不指定 *devicename*，则会为系统中具有 ASM 签名的所有设备都启用 DMP 支持。

例如：

```
# vxdmpraw enable oracle dba 765 eva4k6k0_1
```

ASM 支持已启用。DMP 设备的访问权限设置为由 *mode* 指定的权限。 更改在重新启动后保持不变。

3 如果 ASM _DISKSTRING 设置为 /dev/raw/*，则 vxdmpraw enable 命令将在与 DMP 设备对应的 /dev/raw/* 目录下创建原始设备。查询主要编号为 201 的设备的原始设备，使该原始设备与对应的分区关联。

4 在 ASM 中，确认 ASM 可看到这些新的设备。

```
SQL> select name,path,header_status from v$asm_disk;

NAME  PATH                      HEADER_STATUS
---------------------------------------------
...   .......                   ....
      /dev/vx/dmp/eva4k6k0_1    CANDIDATE
...   .......                   ....
```

从 Oracle 自动存储管理 (ASM) 磁盘的列表中删除 Dynamic Multi-Pathing (DMP) 设备

要从 ASM 磁盘列表中删除 DMP 设备，请从设备中禁用对 ASM 的 DMP 支持。无法从位于 ASM 磁盘组中的设备中删除对 ASM 的 DMP 支持。

从 ASM 磁盘列表中删除 DMP 设备

1 如果设备是任何 ASM 磁盘组的一部分，请从该 ASM 磁盘组中删除该设备。

2 以 root 用户身份禁止将 DMP 设备用于 ASM。

```
# vxdmpraw disable diskname
```

例如：

```
# vxdmpraw disable eva4k6k0_1
```

将操作系统设备上的 Oracle 自动存储管理 (ASM) 磁盘组迁移到 Dynamic Multi-Pathing (DMP) 设备

当现有 ASM 磁盘组将操作系统的本机设备用作磁盘时，可以将这些设备迁移到 Veritas Dynamic Multi-Pathing 的控制之下。如果 OS 设备由其他多径处理驱动程序控制，此操作将需要系统停机才能将设备迁移到 DMP 的控制之下。

完成此过程后，ASM 磁盘组将使用迁移的 DMP 设备作为其磁盘。

“在 ASM 中”表示您以运行 ASM 实例的用户身份执行此步骤。

“以 root 用户身份”表示您以 root 用户身份执行此步骤。

将操作系统设备上的 ASM 磁盘组迁移到 DMP 设备

1 在 ASM 中，标识要迁移的 ASM 磁盘组，并标识受其控制的磁盘。

2 在 ASM 中，卸载 ASM 磁盘组。

3 如果设备由其他多径处理驱动程序控制，则可将这些设备迁移到 DMP 的控制之下。请以 root 用户身份执行这些步骤。

   从 PowerPath 或 Device Mapper Multipath 进行迁移。

   请参见第 25 页的“关于设置 DMP 以管理本机设备”。

4 以 root 用户身份使用 `raw` 命令删除为特定 OS 设备创建的原始设备。

5 以 root 用户身份为步骤 1 中标识的 ASM 磁盘组启用 DMP 支持。

```
# vxdmpraw enable usernamegroupnamemode [devicename ...]
```

其中： *username* 表示运行 ASM 实例的 ASM 用户， *groupname* 表示指定的用户 ID 的 UNIX/Linux 组名称， *mode* 表示设备上设置的权限。如果您指定一个或多个 *devicenames*，将为这些设备启用对 ASM 的 DMP 支持。如果您不指定 *devicename*，则会为系统中具有 ASM 签名的所有设备都启用 DMP 支持。

6 在 ASM 中，适当地设置 ASM_DISKSTRING。首选设置为 `/dev/vx/dmp/*`

7 在 ASM 中，确认设备对 ASM 可用。

8 在 ASM 中，装入 ASM 磁盘组。磁盘组将装入到 DMP 设备中。

在 Linux 上，如果将 ASM_STRING 设置为 /dev/raw/*，则 `vxdmpraw enable` 命令还会在与 DMP 设备相对应的 `/dev/raw` 目录下创建原始设备。

例如：将操作系统设备上的 ASM 磁盘组迁移到 DMP 设备

1 在 ASM 中，标识要迁移的 ASM 磁盘组，并标识受其控制的磁盘。

```
SQL> select name, state from v$asm_diskgroup;
NAME                           STATE
------------------------------ -----------
ASM_DG1                        MOUNTED

SQL> select path , header_status from v$asm_disk where
header_status='MEMBER';

NAME         PATH             HEADER_STATUS
-------------------------------------------
ASM_DG1_0000 /dev/vx/dmp/sda  MEMBER
ASM_DG1_0001 /dev/vx/dmp/sdc  MEMBER
ASM_DG1_0002 /dev/vx/dmp/sdd  MEMBER
```

2 在 ASM 中，卸载 ASM 磁盘组。

```
SQL> alter diskgroup ASM_DG1 dismount;
Diskgroup altered.

SQL> select name , state from v$asm_diskgroup;
NAME                           STATE
------------------------------ -----------
ASM_DG1                        DISMOUNTED
```

3 如果设备由其他多径处理驱动程序控制，则可将这些设备迁移到 DMP 的控制之下。请以 root 用户身份执行这些步骤。

注意：此步骤可能需要安排计划内系统停机时间。

请参见第 25 页的“关于设置 DMP 以管理本机设备”。

4 如果 /dev/raw 设备用于 ASM，则执行此步骤。以 root 用户身份使用 `raw` 命令删除为特定 OS 设备创建的原始设备。

通过观察设备相应的主要和次要编号来确定关联关系。例如，以下 `ls` 命令的输出显示 sdc3 已绑定到编号为 (8,35) 的设备。

```
# ls -l /dev/sdc3
brw-r----- 1 root disk 8, 35 Aug 31 20:42 /dev/sdc3
```

以下 `raw` 命令的输出表示 /dev/raw/raw1 是相应的原始设备：

```
# raw -q /dev/raw/raw1
/dev/raw/raw1:  bound to major 8, minor 35
```

要删除此映射，请将此原始设备与 (0,0) 设备编号相关联：

```
# raw /dev/raw/raw1 0 0
/dev/raw/raw1:  bound to major 0, minor 0

/bin/sync
```

5 以 root 用户身份通过以下方式之一为步骤 1 中标识的 ASM 磁盘组启用 DMP 支持：

- 要迁移所选 ASM 磁盘组，请使用 `vxdmpadm` 命令确定 OS 设备所对应的 DMP 节点。

```
# vxdmpadm getdmpnode nodename= sdd
NAME        STATE    ENCLR-TYPE   PATHS   ENBL  DSBL  ENCLR-NAME
=============================================================
EVA4k6k0_0 ENABLED EVA4K6K      4       4     0     EVA4k6k0
```

在以下命令中使用设备名称：

```
# vxdmpraw enable oracle dba 660 eva4k6k0_0 \
  eva4k6k0_9 emc_clariion0_243
```

- 如果您不指定 *devicename*，则会为磁盘组中具有 ASM 签名的所有设备都启用 DMP 支持。例如：

```
# vxdmpraw enable oracle dba 660
```

6 从 ASM 中，设置 ASM_DISKSTRING。

```
SQL> alter system set ASM_DISKSTRING='/dev/vx/dmp/*';
System altered.
SQL> show parameter ASM_DISKSTRING;
NAME                        TYPE      VALUE
--------------------------- --------- --------------------
asm_diskstring              string    /dev/vx/dmp/*
```

7 在 ASM 中，确认设备对 ASM 可用。

```
SQL> select path , header_status from v$asm_disk where
header_status='MEMBER';


NAME   PATH                            HEADER_STATUS
--------------------------------------------------
       /dev/vx/dmp/emc_clariion0_243  MEMBER
       /dev/vx/dmp/eva4k6k4k0_0       MEMBER
       /dev/vx/dmp/eva4k6k0_1         MEMBER
```

8 在 ASM 中，装入 ASM 磁盘组。磁盘组将装入到 DMP 设备中。

```
SQL> alter diskgroup ASM_DG1 mount;
Diskgroup altered.

SQL> select name, state from v$asm_diskgroup;
NAME                           STATE
------------------------------ -----------
ASM_DG1                        MOUNTED

SQL> select path , header_status from v$asm_disk where
header_status='MEMBER';

NAME          PATH                            HEADER_STATUS
------------------------------------------------------------
ASM_DG1_0002 /dev/vx/dmp/emc_clariion0_243   MEMBER
ASM_DG1_0000 /dev/vx/dmp/eva4k6k0_1          MEMBER
ASM_DG1_0001 /dev/vx/dmp/eva4k6k0_9          MEMBER
```

将 DMP 设备添加到现有的 LVM 卷组或者创建新的 LVM 卷组

dmp_native_support 为 ON 时，可以在可用 DMP 设备上创建新的 LVM 卷组。也可以将可用的 DMP 设备添加到现有的 LVM 卷组。当 LVM 卷组位于 DMP 设备上后，可以使用任一 LVM 命令管理卷组。

在 DMP 设备上创建新的 LVM 卷组或者将 DMP 设备添加到现有的 LVM 卷组

1 选择可供 LVM 使用的磁盘。`vxdisk list` 命令显示未被 VxVM 使用且 TYPE auto:none 和 STATUS Online 无效的磁盘。

```
# vxdisk list

DEVICE                TYPE      DISK  GROUP    STATUS
. . .
tagmastore-usp0_0035 auto:none -     -        online invalid
tagmastore-usp0_0036 auto:none -     -        online invalid
```

2 在 DMP 设备上创建新的 LVM 卷组。

使用 DMP 设备的完整路径名。

```
# pvcreate /dev/vx/dmp/tagmastore-usp0_0035
Physical volume "/dev/vx/dmp/tagmastore-usp0_0035" successfully
created
#
# vgcreate /dev/newvg /dev/vx/dmp/tagmastore-usp0_0035
  Volume group "newvg" successfully created
# vgdisplay -v newvg |grep Name
    Using volume group(s) on command line
    Finding volume group "newvg"
  VG Name                newvg
    PV Name                /dev/vx/dmp/tagmastore-usp0_0035s3
```

3 将 DMP 设备添加到现有的 LVM 卷组。

使用 DMP 设备的完整路径名。

```
# pvcreate /dev/vx/dmp/tagmastore-usp0_0036
Physical volume "/dev/vx/dmp/tagmastore-usp0_0036"
successfully created

# vgextend newvg /dev/vx/dmp/tagmastore-usp0_0036
  Volume group "newvg" successfully extended

# vgdisplay -v newvg |grep Name
    Using volume group(s) on command line
    Finding volume group "newvg"
  VG Name               newvg
  PV Name               /dev/vx/dmp/tagmastore-usp0_0035s3
  PV Name               /dev/vx/dmp/tagmastore-usp0_0036s3
```

4 运行以下命令以触发设备的 DMP 发现：

```
# vxdisk scandisks
```

5 在发现完成后，磁盘显示为正由 LVM 使用：

```
# vxdisk list

. . .
tagmastore-usp0_0035 auto:LVM   -          -    LVM
tagmastore-usp0_0036 auto:LVM   -          -    LVM
```

6 对于所有 LVM 卷条目，将 _netdev 添加到 /etc/fstab 中的装入选项。 此选项可确保在发现 DMP 设备后启用这些卷。

显示本机多径处理配置

为本机设备启用DMP后，dmp_native_support属性显示为ON。该可调参数为ON时，所有 DMP 磁盘均对本机卷可用，但有以下例外：

- 具有 VxVM 标签的设备
  如果您初始化一个磁盘以供 VxVM 使用，将会为该磁盘自动禁用本机多径处理功能。删除 VxVM 标签后，将启用本机多径处理。
- 使用第三方驱动程序进行多径处理的设备
  如果磁盘已使用第三方驱动程序 (TPD) 进行多径处理，则 DMP 不会管理这些设备，除非删除 TPD 支持。

显示是否启用了 DMP

1 显示属性 `dmp_native_support`。

```
# vxdmpadm gettune dmp_native_support
```

2 当 `dmp_native_support` 可调参数为 ON 时，使用 `vxdisk list` 命令可显示可用的卷。对 LVM 可用的卷显示为 TYPE auto:none。已由 LVM 使用的卷显示为 TYPE auto:LVM。

为本机设备删除 DMP 支持

dmp_native_support 可调参数在重新启动和rpm升级后保持不变。

如果为 VxVM 初始化单个设备，或者如果为该设备设置 TPD 多径处理，则可以使该设备脱离 LVM 的控制。

要从所有DMP设备中删除对本机设备的支持，请关闭dmp_native_support可调参数。

关闭 dmp_native_support 可调参数：

```
# vxdmpadm settune dmp_native_support=off
```

查看 dmp_native_support 可调参数的值：

```
# vxdmpadm gettune dmp_native_support
Tunable                Current Value       Default Value
--------------------- ----------------    --------------
dmp_native_support          off                off
```

3

管理 DMP

本章节包括下列主题：

- 关于对控制器与存储处理器启用和禁用 I/O
- 关于显示 DMP 数据库信息
- 显示磁盘路径
- 设置 DMP 节点的自定义名称
- 使用 vxdmpadm 管理 DMP

关于对控制器与存储处理器启用和禁用 I/O

DMP 允许您关闭通过 HBA 控制器或存储处理器磁盘阵列端口的 I/O，以便您可以执行管理操作。此功能可用于维护主机的 HBA 控制器或挂接到受 DMP 支持的磁盘阵列上的阵列端口。完成维护任务后，可以重新打开 HBA 控制器或阵列端口的 I/O 操作。可以使用 `vxdmpadm` 命令来完成这些操作。

对于主动/主动类型的磁盘阵列，通过 HBA 控制器或阵列端口禁用 I/O 后，I/O 在剩余路径上继续。对于主动/被动类型的磁盘阵列，如果禁用通过 HBA 控制器或阵列端口的 I/O 导致了所有主路径被禁用，DMP 将故障转移到次级路径且 I/O 将在这些次级路径上继续。

DMP 不支持操作针对将第三方驱动程序 (TPD) 用于多径处理的控制器启用 I/O 或禁用 I/O。

管理操作结束之后，可以使用 `vxdmpadm` 命令重新启用通过 HBA 控制器的路径。

请参见第 68 页的“对路径、控制器、阵列端口或 DMP 节点禁用 I/O”。

请参见第 69 页的“为路径、控制器、阵列端口或 DMP 节点启用 I/O”。

也可以动态地联机执行某些重新配置操作。

关于显示 DMP 数据库信息

可以使用 vxdmpadm 命令列出 DMP 数据库信息和执行其他管理任务。此命令允许您列出连接到磁盘的所有控制器和其他存储在 DMP 数据库的相关信息。可以使用此信息定位系统硬件，并帮助您决定需要启用或禁用哪个控制器。

vxdmpadm 命令还提供其他有用的信息，如磁盘阵列序列号、哪些 DMP 设备（磁盘）连接到了磁盘阵列以及哪些路径连接到了特定控制器、磁盘阵列或磁盘阵列端口等。

请参见第 43 页的“使用 vxdmpadm 管理 DMP”。

显示磁盘路径

vxdisk 命令用于显示特定元设备的多径处理信息。元设备是通过系统 HBA 控制器建立了多条物理路径的物理磁盘的设备表现形式。在 DMP 中，系统中的所有物理磁盘都表示为具有一条或多条物理路径的元设备。

显示系统上的多径处理信息

◆ 可以使用 vxdisk path 命令显示系统中的设备路径、磁盘访问名称、磁盘介质名称和磁盘组之间的关系，如下所示：

```
# vxdisk path

SUBPATH      DANAME     DMNAME     GROUP     STATE
sda          sda        mydg01     mydg      ENABLED
sdi          sdi        mydg01     mydg      ENABLED
sdb          sdb        mydg02     mydg      ENABLED
sdj          sdj        mydg02     mydg      ENABLED
  .
  .
  .
```

这显示了两个磁盘 mydg01 和 mydg02 中的每一个都存在两条路径，并且也指示每个磁盘都处于 ENABLED 状态。

查看特定元设备的多径处理信息

1 请使用以下命令：

```
# vxdisk list devicename
```

例如，要查看 sdl 设备的多径处理信息，请使用以下命令：

```
# vxdisk list sdl
```

vxdisk list 命令的输出显示了多径处理信息，如下例所示：

```
Device:    sdl
devicetag: sdl
type:      sliced
hostid:    sys1
.
.
.
Multipathing information:
numpaths:   2
sdl   state=enabled     type=secondary
sdp   state=disabled    type=primary
```

numpaths 行显示有 2 条到设备的路径。Multipathing informatio 部分中的后面两行显示其中一条路径是活动的 (state=enabled)，而另一条路径已失败 (state=disabled)。

对于 EMC CLARiiON、Hitachi HDS 9200 与 9500、Sun StorEdge 6xxx 以及 Sun StorEdge T3 阵列等主动/被动类型磁盘阵列中的磁盘，将显示 type 字段。该字段指明磁盘的主路径和次级路径。

对于 EMC Symmetrix、Hitachi HDS 99xx 和 Sun StorEdge 99xx 系列以及 IBM ESS 系列的主动/主动类型磁盘阵列，将不显示 type 字段。这些阵列没有主路径和次级路径的概念。

2 或者，可以使用以下命令查看多径处理信息：

```
# vxdmpadm getsubpaths dmpnodename=devicename
```

例如，要查看 emc_clariion0_893 的多径处理信息，请使用以下命令：

```
# vxdmpadm getsubpaths dmpnodename=emc_clariion0_893
```

vxdmpadm getsubpaths 命令的典型输出如下：

```
NAME      STATE[A]   PATH-TYPE[M] CTLR-NAME  ENCLR-TYPE   ENCLR-NAME     ATTRS
==========================================================================
sdbc      ENABLED(A)  PRIMARY      c3        EMC_CLARiiON  emc_clariion0     -
sdbm      ENABLED     SECONDARY    c3        EMC_CLARiiON  emc_clariion0     -
sdbw      ENABLED(A)  PRIMARY      c3        EMC_CLARiiON  emc_clariion0     -
sdck      ENABLED(A)  PRIMARY      c2        EMC_CLARiiON  emc_clariion0     -
sdcu      ENABLED     SECONDARY    c2        EMC_CLARiiON  emc_clariion0     -
sdde      ENABLED(A)  PRIMARY      c2        EMC_CLARiiON  emc_clariion0     -
```

设置 DMP 节点的自定义名称

DMP 节点名称是表示多条指向磁盘的路径的元设备名称。DMP 节点名称是根据 DMP 命名机制从设备名称生成的。

请参见第 19 页的“DMP 中的磁盘设备命名”。

可以为 DMP 节点指定一个自定义的名称。用户指定的名称是持久性的，即使已禁用名称持久性。

不能指派已被设备使用的自定义名称。但是，如果指派的名称遵循与DDL生成的名称相同的命名约定，则添加设备时可能会发生名称冲突。如果 DMP 设备的用户定义名称与另一个 DMP 设备的 DDL 生成的名称相同，则 vxdisk list 命令输出中会将其中一个设备显示为 error。

为 DMP 节点指定自定义名称

◆ 请使用以下命令：

```
# vxdmpadm setattr dmpnode dmpnodename name=name
```

也可以从输入文件指定名称。这样便可以在系统上为 DMP 节点自定义有意义的名称。

从文件中指定 DMP 节点

1 使用 vxgetdmpnames 脚本从配置中的设备上获得已填充数据的示例文件。该示例文件显示了所需的格式并可用作指定自定义名称的模板。

2 若要指定名称，请使用以下命令：

```
# vxddladm assign names file=pathname
```

清除自定义名称

◆ 要清除自定义名称并使用默认的 OSN 或 EBN 名称，请使用以下命令：

```
# vxddladm -c assign names
```

使用 vxdmpadm 管理 DMP

vxdmpadm 实用程序是 DMP 的命令行管理界面。

可以使用 vxdmpadm 实用程序执行下列任务：

- 检索与特定路径对应的 DMP 设备的名称。
  请参见第 44 页的“检索有关 DMP 节点的信息”。
- 显示关于 DMP 节点的综合信息。
  请参见第 45 页的“显示关于 DMP 节点的综合信息”。
- 显示 LUN 组的成员。
  请参见第 47 页的“显示 LUN 组的成员”。
- 列出 DMP 设备节点、HBA 控制器、磁盘阵列或阵列端口下的所有路径。
  请参见第 47 页的“显示由 DMP 节点、控制器、磁盘阵列或阵列端口控制的路径”。
- 显示主机上 HBA 控制器的信息。
  请参见第 49 页的“显示控制器的信息”。
- 显示磁盘阵列的信息。
  请参见第 50 页的“显示磁盘阵列的相关信息”。
- 显示接入磁盘阵列存储处理器的磁盘阵列端口的信息。
  请参见第 51 页的“显示磁盘阵列端口的信息”。
- 显示由第三方多径处理驱动程序控制的设备的信息。
  请参见第 51 页的“显示有关 TPD 控制的设备的信息”。
- 显示扩展的设备属性。
  请参见第 52 页的“显示扩展设备属性”。

- 请参见第 54 页的“隐藏或包含 VxVM 控制中的设备”。
  在 DMP 控制中隐藏或包括设备。
- 收集 DMP 节点、磁盘阵列、路径或控制器的 I/O 统计信息。
  请参见第 55 页的“收集和显示 I/O 统计信息”。
- 配置磁盘阵列路径的属性。
  请参见第 60 页的“设置磁盘阵列路径的属性”。
- 显示设备或磁盘阵列的冗余级别
  请参见第 61 页的“显示设备或磁盘阵列的冗余级别”。
- 指定最小活动路径数
  请参见第 62 页的“指定最小活动路径数”。
- 显示或设置用于磁盘阵列路径的 I/O 策略。
  请参见第 63 页的“指定 I/O 策略”。
- 为系统上的路径、HBA 控制器或磁盘阵列端口启用或禁用 I/O 路径。
  请参见第 68 页的“对路径、控制器、阵列端口或 DMP 节点禁用 I/O”。
- 重命名磁盘阵列。
  请参见第 70 页的“重命名磁盘阵列”。
- 配置 DMP 响应 I/O 请求故障的方式。
  请参见第 71 页的“配置对 I/O 故障的响应”。
- 配置 I/O 限制机制。
  请参见第 72 页的“配置 I/O 控制机制”。
- 控制 DMP 路径还原线程的操作。
  请参见第 75 页的“配置 DMP 路径还原策略”。
- 配置阵列策略模块
  请参见第 77 页的“配置阵列策略模块”。
- 获取或设置由 DMP 使用的各种可调参数的值。
  请参见第 120 页的“DMP 可调参数”。

以下各节详细讨论这些任务以及示例输出。

请参见 `vxdmpadm`(1M) 手册页。

检索有关 DMP 节点的信息

以下命令显示控制特定物理路径的 DMP 节点：

vxdmpadm getdmpnode nodename=*pathname*

物理路径由 `nodename` 属性的参数指定，它必须是在设备目录中列出的有效路径。

设备目录是 /dev 目录。

此命令显示的输出类似于下面的示例输出。

```
# vxdmpadm getdmpnode nodename=sdbc

NAME               STATE   ENCLR-TYPE   PATHS ENBL DSBL ENCLR-NAME
===================================================================
emc_clariion0_89 ENABLED EMC_CLARiiON 6     6    0    emc_clariion0
```

使用 -v 选项可显示 LUN 序列号和阵列卷 ID。

```
# vxdmpadm -v getdmpnode nodename=sdbc

NAME              STATE  ENCLR-TYPE  PATHS ENBL DSBL ENCLR-NAME SERIAL-NO ARRAY_VOL_ID
=================================================================================
emc_clariion0_89 ENABLED EMC_CLARiiON 6   6    0   emc_clariion0 600601601 893
```

将 enclosure 属性和 getdmpnode 一起使用可以获得指定磁盘阵列的所有 DMP 节点的列表。

```
# vxdmpadm getdmpnode enclosure=enc0

NAME     STATE    ENCLR-TYPE PATHS  ENBL  DSBL ENCLR-NAME
=========================================================
sdm      ENABLED  ACME       2      2     0    enc0
sdn      ENABLED  ACME       2      2     0    enc0
sdo      ENABLED  ACME       2      2     0    enc0
sdp      ENABLED  ACME       2      2     0    enc0
```

将 dmpnodename 属性与 getdmpnode 一起使用可显示给定 DMP 节点的 DMP 信息。

```
# vxdmpadm getdmpnode dmpnodename=emc_clariion0_158

NAME               STATE   ENCLR-TYPE   PATHS ENBL DSBL ENCLR-NAME
===================================================================
emc_clariion0_158 ENABLED EMC_CLARiiON 1     1    0    emc_clariion0
```

显示关于 DMP 节点的综合信息

vxdmpadm list dmpnode 命令可显示 DMP 节点的详细信息。这些信息包括磁盘阵列名称、LUN 序列号、端口 ID 信息、设备属性等。

以下命令可显示系统中所有 DMP 节点的综合信息：

```
# vxdmpadm list dmpnode all
```

在 `list dmpnode` 中使用 `enclosure` 属性可以获得指定磁盘阵列所有 DMP 节点的列表。

vxdmpadm list dmpnode enclosure=*enclosure name*

例如，以下命令可显示 `enc0` 磁盘阵列中所有 DMP 节点上的综合信息。

vxdmpadm list dmpnode enclosure=enc0

在 `list dmpnode` 命令中使用 `dmpnodename` 属性可以显示指定 DMP 节点的 DMP 信息。DMP 节点可以通过名称来指定，也可以通过指定路径名称来指定。 指定 DMP 节点的详细信息包括所列 DMP 节点每条子路径的路径信息。

在由于故障而禁用的路径以及出于管理目的而手动禁用的路径之间，路径状态不同。使用 `vxdmpadm disable` 命令手动禁用的路径显示为 `disabled(m)`。

vxdmpadm list dmpnode dmpnodename=*dmpnodename*

例如，以下命令显示 DMP 节点 emc_clariion0_158 的综合信息。

vxdmpadm list dmpnode dmpnodename=emc_clariion0_158

```
dmpdev = emc_clariion0_158
state       = enabled
enclosure   = emc_clariion0
cab-sno     = CK200070400359
asl         = libvxCLARiiON.so
vid         = DGC
pid         = DISK
array-name  = EMC_CLARiiON
array-type  = CLR-A/PF
iopolicy    = MinimumQ
avid        = 158
lun-sno     = 600601601A141B001D4A32F92B49DE11
udid        = DGC%5FDISK%5FCK200070400359%5F600601601A141B001D4A32F92B49DE11
dev-attr    = lun
###path     = name state type transport ctlr hwpath aportID aportWWN attr
path        = sdck enabled(a) primary FC c2 c2 A5 50:06:01:61:41:e0:3b:33 -
path        = sdde enabled(a) primary FC c2 c2 A4 50:06:01:60:41:e0:3b:33 -
path        = sdcu enabled secondary FC c2 c2 B4 50:06:01:68:41:e0:3b:33 -
path        = sdbm enabled secondary FC c3 c3 B4 50:06:01:68:41:e0:3b:33 -
path        = sdbw enabled(a) primary FC c3 c3 A4 50:06:01:60:41:e0:3b:33 -
path        = sdbc enabled(a) primary FC c3 c3 A5 50:06:01:61:41:e0:3b:33 -
```

显示 LUN 组的成员

下面的命令显示与指定的 DMP 节点位于同一 LUN 组中的各个 DMP 节点：

```
# vxdmpadm getlungroup dmpnodename=sdq

NAME      STATE    ENCLR-TYPE  PATHS     ENBL    DSBL   ENCLR-NAME
==============================================================
sdo       ENABLED  ACME        2         2       0       enc1
sdp       ENABLED  ACME        2         2       0       enc1
sdq       ENABLED  ACME        2         2       0       enc1
sdr       ENABLED  ACME        2         2       0       enc1
```

显示由 DMP 节点、控制器、磁盘阵列或阵列端口控制的路径

vxdmpadm getsubpaths 命令列出 DMP 已知的全部路径。vxdmpadm getsubpaths 命令还提供了一些选项，用于列出通过特定 DMP 节点、控制器、磁盘阵列或阵列端口的子路径。要列出通过某个阵列端口的路径，请指定磁盘阵列名称和阵列端口 ID 的组合，或指定阵列端口 WWN。

列出 DMP 已知的全部子路径：

```
# vxdmpadm getsubpaths

NAME      STATE[A]   PATH-TYPE[M] DMPNODENAME  ENCLR-NAME   CTLR   ATTRS
==============================================================================
sdaf      ENABLED(A) PRIMARY      ams_wms0_130 ams_wms0     c2       -
sdc       ENABLED    SECONDARY    ams_wms0_130 ams_wms0     c3       -
sdb       ENABLED(A)   -          disk_24      disk         c0       -
sda       ENABLED(A)   -          disk_25      disk         c0       -
sdav      ENABLED(A) PRIMARY      emc_clariion0_1017 emc_clariion0 c3       -
sdbf      ENABLED    SECONDARY    emc_clariion0_1017 emc_clariion0 c3       -
```

组合使用 dmpnodename 属性的 vxdmpadm getsubpaths 命令显示由 /dev/vx/rdmp 目录下的指定 DMP 节点名控制的 LUN 的所有路径：

```
# vxdmpadm getsubpaths dmpnodename=sdu

NAME     STATE[A]  PATH-TYPE[M] CTLR-NAME ENCLR-TYPE ENCLR-NAME ATTRS
=================================================================
sdu      ENABLED(A) PRIMARY     c2        ACME       enc0       -
sdt      ENABLED    PRIMARY     c1        ACME       enc0       -
```

对于 A/A 阵列，所有可用于 I/O 的已启用路径均显示为 ENABLED(A)。

对于 I/O 策略设置为 singleactive 的 A/P 阵列，其中只有一条路径显示为 ENABLED(A)。其他路径已启用，但不能用于 I/O。如果 I/O 策略未设置为 singleactive，则 DMP 可以将一组路径（全部为主路径或全部为次级路径）用于 I/O，这些路径都显示为 ENABLED(A)。

请参见第 63 页的“指定 I/O 策略”。

处于 DISABLED 状态的路径不可用于 I/O 操作。

由系统管理员手动禁用的路径显示为DISABLED(M)。失败的路径显示为DISABLED。

可以使用 getsubpaths 获得连接到特定 HBA 控制器的所有路径的信息：

```
# vxdmpadm getsubpaths ctlr=c2

NAME   STATE[-]    PATH-TYPE[-] CTLR-NAME ENCLR-TYPE ENCLR-NAME ATTRS
=================================================================
sdk    ENABLED(A)  PRIMARY      sdk       ACME       enc0       -
sdl    ENABLED(A)  PRIMARY      sdl       ACME       enc0       -
sdm    DISABLED    SECONDARY    sdm       ACME       enc0       -
sdn    ENABLED     SECONDARY    sdn       ACME       enc0       -
```

可以使用 getsubpaths 命令来获得接入磁盘阵列的某个端口的所有路径的信息。磁盘阵列端口可以通过磁盘阵列名称与磁盘阵列端口ID或者磁盘阵列端口的WWN标识符来指定：

```
# vxdmpadm getsubpaths enclosure=enclosure portid=portid
# vxdmpadm getsubpaths pwwn=pwwn
```

例如，要列出通过阵列端口和磁盘阵列的子路径以及阵列端口 ID：

```
# vxdmpadm getsubpaths enclosure=emc_clariion0 portid=A5

NAME          STATE[A]    PATH-TYPE[M] DMPNODENAME   ENCLR-NAME    CTLR     ATTRS
================================================================================
sdav          ENABLED(A)  PRIMARY      emc_clariion0_1017 emc_clariion0 c3        -
sdcd          ENABLED(A)  PRIMARY      emc_clariion0_1017 emc_clariion0 c2        -
sdau          ENABLED(A)  PRIMARY      emc_clariion0_1018 emc_clariion0 c3        -
sdcc          ENABLED(A)  PRIMARY      emc_clariion0_1018 emc_clariion0 c2        -
```

例如，要列出通过某个阵列端口和 WWN 的子路径：

```
# vxdmpadm getsubpaths pwwn=50:06:01:61:41:e0:3b:33

NAME          STATE[A]    PATH-TYPE[M] CTLR-NAME   ENCLR-TYPE    ENCLR-NAME     ATTRS
================================================================================
sdav          ENABLED(A)   PRIMARY      c3          EMC_CLARiiON  emc_clariion0      -
```

```
sdcd          ENABLED(A)  PRIMARY      c2          EMC_CLARiiON  emc_clariion0     -
sdau          ENABLED(A)  PRIMARY      c3          EMC_CLARiiON  emc_clariion0     -
sdcc          ENABLED(A)  PRIMARY      c2          EMC_CLARiiON  emc_clariion0     -
```

vxdmpadm getsubpaths pwwn=20:00:00:E0:8B:06:5F:19

可以使用 `getsubpaths` 获取有关磁盘阵列所有子路径的信息。

vxdmpadm getsubpaths enclosure=*enclosure_name* [ctlr=*ctlrname*]

列出某个磁盘阵列的全部子路径：

vxdmpadm getsubpaths enclosure=emc_clariion0

```
NAME          STATE[A]    PATH-TYPE[M] DMPNODENAME  ENCLR-NAME   CTLR    ATTRS
================================================================================
sdav          ENABLED(A) PRIMARY      emc_clariion0_1017 emc_clariion0 c3    -
sdbf          ENABLED    SECONDARY    emc_clariion0_1017 emc_clariion0 c3    -
sdau          ENABLED(A) PRIMARY      emc_clariion0_1018 emc_clariion0 c3    -
sdbe          ENABLED    SECONDARY    emc_clariion0_1018 emc_clariion0 c3    -
```

列出某个磁盘阵列上某个控制器的全部子路径：

vxdmpadm getsubpaths enclosure=Disk ctlr=c1

默认情况下，`vxdmpadm getsubpaths` 命令的输出按磁盘阵列名称、DMP 节点名称及其中的路径名排序。

要根据路径名、DMP 节点名称、磁盘阵列名称或主机控制器名称对输出进行排序，请使用 `-s` 选项。

要对子路径信息进行排序，请使用以下命令：

**vxdmpadm -s {path | dmpnode | enclosure | ctlr} getsubpaths **
**[all | ctlr=*ctlr_name* | dmpnodename=*dmp_device_name* | **
**enclosure=*enclr_name* [ctlr=*ctlr_name* | portid=*array_port_ID*] | **
pwwn=*port_WWN* | tpdnodename=*tpd_node_name*]

显示控制器的信息

以下命令列出系统上所有 HBA 控制器的属性：

vxdmpadm listctlr all

```
CTLR-NAME       ENCLR-TYPE        STATE          ENCLR-NAME
================================================================
c1              OTHER             ENABLED        other0
c2              X1                ENABLED        jbod0
```

```
c3                ACME                ENABLED         enc0
c4                ACME                ENABLED         enc0
```

该输出表明控制器 c1 连接到了不属于任何可识别 DMP 类别的磁盘，因为磁盘阵列的类型为 OTHER。

其他控制器已连接到属于可识别 DMP 类别的磁盘。

所有控制器均处于 ENABLED 状态，表明可用于 I/O 操作。

状态 DISABLED 用来指明控制器不可用于 I/O 操作。该不可用性可能是由于硬件故障导致，或者是由于已使用 vxdmpadm disable 命令禁用了该控制器上的 I/O 操作导致。

以下格式的命令列出了属于指定磁盘阵列或磁盘阵列类型的控制器：

```
# vxdmpadm listctlr enclosure=enc0
```

或

```
# vxdmpadm listctlr type=ACME
```

```
CTLR-NAME          ENCLR-TYPE           STATE         ENCLR-NAME
=================================================================
c2                 ACME                 ENABLED       enc0
c3                 ACME                 ENABLED       enc0
```

使用 vxdmpadm getctlr 命令可以显示 HBA 供应商详细信息以及控制器 ID。对于 iSCSI 设备，控制器 ID 为基于 IQN 或 IEEE 格式的名称。对于 FC 设备，控制器 ID 为 WWN。由于 WWN 是从 ESD 获取的，因此如果未运行 ESD，则该字段将为空。ESD 是一个后台驻留程序进程，用于通知 DDL 有关事件的发生情况。显示为 “Controller ID (控制器 ID)” 的 WWN 会映射到与主机控制器关联的 HBA 端口的 WWN。

```
# vxdmpadm getctlr c5
```

```
LNAME      PNAME  VENDOR   CTLR-ID
=====================================================
c5         c5     qlogic       20:07:00:a0:b8:17:e1:37
```

显示磁盘阵列的相关信息

要显示指定磁盘阵列的属性，包括其磁盘阵列类型、磁盘阵列序列号、状态、阵列类型以及 LUN 数量，请使用下面的命令：

```
# vxdmpadm listenclosure enc0
```

```
ENCLR_NAME ENCLR_TYPE  ENCLR_SNO            STATUS  ARRAY_TYPE    LUN_COUNT
==========================================================================
enc0       A3          60020f20000001a90000 CONNECTED  A/P        30
```

以下命令列出系统中所有磁盘阵列的属性：

\# **vxdmpadm listenclosure all**

```
ENCLR_NAME ENCLR_TYPE ENCLR_SNO            STATUS   ARRAY_TYPE    LUN_COUNT
=================================================================================
Disk       Disk       DISKS                CONNECTED     Disk       6
ANA0       ACME       508002000001d660     CONNECTED     A/A       57
enc0       A3         60020f20000001a90000 CONNECTED     A/P       30
```

显示磁盘阵列端口的信息

使用本节中的命令可以显示有关阵列端口的信息。显示的磁盘端口信息包括其磁盘阵列名称、以及其 ID 与 WWN 标识符。

要显示可以通过路径、DMP 节点或 HBA 控制器访问的阵列端口的属性，请使用以下命令之一：

\# **vxdmpadm getportids path=*path-name***

\# **vxdmpadm getportids dmpnodename=*dmpnode-name***

\# **vxdmpadm getportids ctlr=*ctlr-name***

以下形式的命令显示指定磁盘阵列内所有阵列端口的相关信息：

\# **vxdmpadm getportids enclosure=*enclr-name***

下面的示例显示可以通过 DMP 节点 `sdg` 访问的阵列端口的相关信息：

\# **vxdmpadm getportids dmpnodename=sdg**

```
NAME       ENCLR-NAME  ARRAY-PORT-ID pWWN
==============================================================
sdg        HDS9500V0   1A            20:00:00:E0:8B:06:5F:19
```

显示有关 TPD 控制的设备的信息

通过第三方驱动程序 (TPD) 共存功能，可以在保留 DMP 监视功能的同时，使得由第三方多径处理驱动程序控制的 I/O 绕过 DMP。使用以下两条命令可分别显示 DMP 发现的、给定 TPD 设备的路径，以及对应于由 DMP 发现并受 TPD 控制的给定节点的 TPD 设备：

```
# vxdmpadm getsubpaths tpdnodename=TPD_node_name
# vxdmpadm gettpdnode nodename=TPD_path_name
```

请参见第 97 页的“更改 TPD 控制的磁盘阵列的设备命名”。

例如，请考虑 PowerPath 控制的 EMC Symmetrix 阵列中的以下磁盘，这些磁盘均为 DMP 已知的磁盘：

```
# vxdisk list

DEVICE          TYPE            DISK        GROUP     STATUS
emcpower10      auto:sliced     disk1       ppdg      online
emcpower11      auto:sliced     disk2       ppdg      online
emcpower12      auto:sliced     disk3       ppdg      online
emcpower13      auto:sliced     disk4       ppdg      online
emcpower14      auto:sliced     disk5       ppdg      online
emcpower15      auto:sliced     disk6       ppdg      online
emcpower16      auto:sliced     disk7       ppdg      online
emcpower17      auto:sliced     disk8       ppdg      online
emcpower18      auto:sliced     disk9       ppdg      online
emcpower19      auto:sliced     disk10      ppdg      online
```

以下命令显示了 DMP 已发现且对应于 PowerPath 控制的节点 `emcpower10` 的路径：

```
# vxdmpadm getsubpaths tpdnodename=emcpower10

NAME     TPDNODENAME   PATH-TYPE[-]DMP-NODENAME   ENCLR-TYPE ENCLR-NAME
=====================================================================
sdq      emcpower10s2 -            emcpower10     PP_EMC     pp_emc0
sdr      emcpower10s2 -            emcpower10     PP_EMC     pp_emc0
```

与此相反，下一个命令显示对应于 DMP 发现的路径 `sdq` 的 PowerPath 节点的信息：

```
# vxdmpadm gettpdnode nodename=sdq

NAME          STATE     PATHS   ENCLR-TYPE     ENCLR-NAME
=====================================================================
emcpower10s2 ENABLED   2       PP_EMC         pp_emc0
```

显示扩展设备属性

设备发现层 (DDL) 扩展属性是与 DDL 发现的 VxVM 或 DMP LUN 或磁盘对应的属性或标志。这些属性将 LUN 识别为特定的硬件类别。

类别列表包括：

硬件 RAID 类型	显示 LUN 所属的存储 RAID 组的种类
精简置备发现和回收	显示 LUN 的精简回收功能
设备介质类型	显示介质类型 - 是否为 SSD（固态磁盘）
基于存储的快照/克隆	显示 LUN 为 PRIMARY LUN 的快照还是克隆
基于存储的复制	显示 LUN 是否为远程站点中的复制组的一部分
传输	显示用于连接到此 LUN 的 HBA 的种类（FC、SATA、iSCSI）

每个 LUN 可以具有一个或多个上述扩展属性。在阵列支持库 (ASL) 中执行设备发现时，DDL 会发现扩展属性。如果已存在 Veritas Operations Manager (VOM)，则 DDL 也可以从配置为托管主机的主机的 VOM 管理服务器中获得扩展属性。

`vxdisk -p list` 命令显示 DDL 扩展属性。例如，以下命令显示此 LUN 的 std、fc 和 RAID_5 的属性：

```
# vxdisk -p list
DISK           : tagmastore-usp0_0e18
DISKID         : 1253585985.692.rx2600h11
VID            : HITACHI
UDID           : HITACHI%5FOPEN-V%5F02742%5F0E18
REVISION       : 5001
PID            : OPEN-V
PHYS_CTLR_NAME : 0/4/1/1.0x50060e8005274246
LUN_SNO_ORDER  : 411
LUN_SERIAL_NO  : 0E18
LIBNAME        : libvxhdsusp.sl
HARDWARE_MIRROR: no
DMP_DEVICE     : tagmastore-usp0_0e18
DDL_THIN_DISK  : thick
DDL_DEVICE_ATTR: std fc RAID_5
CAB_SERIAL_NO  : 02742
ATYPE          : A/A
ARRAY_VOLUME_ID: 0E18
ARRAY_PORT_PWWN: 50:06:0e:80:05:27:42:46
ANAME          : TagmaStore-USP
TRANSPORT      : FC
```

`vxdisk -x attribute -p list` 命令显示一行列表，列出了属性列表和属性。以下示例显示两个 Hitachi LUN，这两个 LUN 通过 `hdprclm` 属性支持精简回收：

```
# vxdisk -x DDL_DEVICE_ATTR -p list
DEVICE                    DDL_DEVICE_ATTR
tagmastore-usp0_0a7a      std fc RAID_5
tagmastore-usp0_065a      hdprclm fc
tagmastore-usp0_065b      hdprclm fc
```

用户可以在同一命令中指定多个 -x 选项以显示多个条目。例如：

```
# vxdisk -x DDL_DEVICE_ATTR -x VID -p list

DEVICE                 DDL_DEVICE_ATTR  VID
tagmastore-usp0_0a7a std fc RAID_5     HITACHI
tagmastore-usp0_0a7b std fc RAID_5     HITACHI
tagmastore-usp0_0a78 std fc RAID_5     HITACHI
tagmastore-usp0_0a79 std fc RAID_5     HITACHI
tagmastore-usp0_065a hdprclm fc        HITACHI
tagmastore-usp0_065b hdprclm fc        HITACHI
tagmastore-usp0_065c hdprclm fc        HITACHI
tagmastore-usp0_065d hdprclm fc        HITACHI
```

使用 `vxdisk -e list` 命令可以显示名为 ATTR 的最后一列中的 DLL_DEVICE_ATTR 属性。

```
# vxdisk -e list
DEVICE                 TYPE  DISK  GROUP  STATUS  OS_NATIVE_NAME  ATTR
tagmastore-usp0_0a7a auto  -     -      online  c10t0d2         std fc RAID_5
tagmastore-usp0_0a7b auto  -     -      online  c10t0d3         std fc RAID_5
tagmastore-usp0_0a78 auto  -     -      online  c10t0d0         std fc RAID_5
tagmastore-usp0_0655 auto  -     -      online  c13t2d7         hdprclm fc
tagmastore-usp0_0656 auto  -     -      online  c13t3d0         hdprclm fc
tagmastore-usp0_0657 auto  -     -      online  c13t3d1         hdprclm fc
```

有关支持扩展属性的 ASL 列表以及这些属性的说明，请参考位于以下 URL 的硬件兼容性列表 (HCL)：

http://www.symantec.com/docs/TECH170013

隐藏或包含 VxVM 控制中的设备

使用 `vxdmpadm exclude` 命令可根据指定的条件隐藏 VxVM 中的设备。当设备被隐藏时，DMP 不声明该设备，因此该设备不可用于 VxVM。可以使用 `vxdmpadm include` 命令将设备重新添加到 VxVM 控制中。可以根据 VID:PID 组合、路径、控制器或磁盘来包含或排除设备。使用叹号 (!) 可以排除或包含任何路径或控制器，已指定的除外。

不能隐藏根磁盘。如果外部磁盘的 VID:PID 与根磁盘的 VID:PID 相同，且根磁盘封装在 VxVM 下，则操作会失败。

注意：在某些 shell 中，! 字符是特殊字符。以下语法显示了如何在 Bash Shell 中将其转义。

```
# vxdmpadm exclude { all | product=VID:PID |
ctlr=[\!]ctlrname | dmpnodename=diskname [ path=[\!]pathname] }
```

```
# vxdmpadm include { all | product=VID:PID |
ctlr=[\!]ctlrname | dmpnodename=diskname [ path=[\!]pathname] }
```

其中：

all - 所有设备

产品=*VID:PID* – 所有具有指定 VID:PID 的设备

ctlr=*ctlrname* – 所有通过给定控制器的设备

dmpnodename=*diskname* - DMP 节点下的所有路径

dmpnodename=*diskname* path=\!*pathname* - DMP 节点下的所有路径，指定路径除外。

收集和显示 I/O 统计信息

可以使用 `vxdmpadm iostat` 命令收集和显示指定 DMP 节点、磁盘阵列、路径或控制器的 I/O 统计信息。

要启用统计信息收集功能，请输入以下命令：

```
# vxdmpadm iostat start [memory=size]
```

要将 I/O 计数器重置为零，请使用以下命令：

```
# vxdmpadm iostat reset
```

`memory` 属性可用于限制为每个 CPU 记录 I/O 统计数据时所用的最大内存量。默认的限制是每个 CPU `32k`（32 KB）。

要定期显示一段时间中累积的统计信息，请使用以下命令：

```
# vxdmpadm iostat show {all | ctlr=ctlr-name \
| dmpnodename=dmp-node \
| enclosure=enclr-name [portid=array-portid ] \
| pathname=path-name | pwwn=array-port-wwn } \
[interval=seconds [count=N]]
```

此命令可显示所有路径 (`all`) 的 I/O 统计数据，或显示指定控制器、DMP 节点、磁盘阵列、路径或端口 ID 的 I/O 统计数据。显示的统计数据包括每个 CPU 用于累积统计数据的 CPU 用量和内存量、读/写操作次数、读/写字节数以及读/写每 KB 的平均时间（以毫秒为单位）。

`interval` 和 `count` 属性可分别用于指定两次显示 I/O 统计数据之间间隔的时间（以秒为单位）以及要显示的行数。如果没有足够的内存来记录这些统计信息，则实际间隔可能小于这一指定值。

要禁用统计信息收集功能，请输入以下命令：

```
# vxdmpadm iostat stop
```

vxdmpadm iostat 命令的用法示例

以下是使用 `vxdmpadm iostat` 命令的示例会话。第一个命令启用 I/O 统计数据的收集：

```
# vxdmpadm iostat start
```

第二个命令显示所有路径上的当前统计数据，包括累计的总读写操作次数和读写的 KB 数。

```
# vxdmpadm -u k iostat show all
                    cpu usage = 7952us    per cpu memory = 8192b
             OPERATIONS           BYTES          AVG TIME(ms)
PATHNAME READS     WRITES    READS     WRITES   READS   WRITES
sdf         87          0    44544k         0    0.00     0.00
sdk          0          0        0          0    0.00     0.00
sdg         87          0    44544k         0    0.00     0.00
sdl          0          0        0          0    0.00     0.00
sdh         87          0    44544k         0    0.00     0.00
sdm          0          0        0          0    0.00     0.00
sdi         87          0    44544k         0    0.00     0.00
sdn          0          0        0          0    0.00     0.00
sdj         87          0    44544k         0    0.00     0.00
sdo          0          0        0          0    0.00     0.00
sdj         87          0    44544k         0    0.00     0.00
sdp          0          0        0          0    0.00     0.00
```

以下命令更改 `vxdmpadm` 可用来累积统计数据的内存量：

```
# vxdmpadm iostat start memory=4096
```

显示的统计数据可以按路径名、DMP 节点名和磁盘阵列名进行过滤（注意：基于 CPU 的内存在上一个命令运行后已发生更改）：

```
# vxdmpadm -u k iostat show pathname=sdk
                    cpu usage = 8132us    per cpu memory = 4096b
               OPERATIONS             BYTES             AVG TIME(ms)
PATHNAME    READS     WRITES      READS     WRITES     READS   WRITES
sdk             0          0          0          0      0.00     0.00

# vxdmpadm -u k iostat show dmpnodename=sdf
                    cpu usage = 8501us    per cpu memory = 4096b
               OPERATIONS             BYTES             AVG TIME(ms)
PATHNAME    READS     WRITES      READS     WRITES     READS   WRITES
sdf          1088          0     557056k          0      0.00     0.00

# vxdmpadm -u k iostat show enclosure=Disk
                   cpu usage = 8626us    per cpu memory = 4096b
               OPERATIONS             BYTES             AVG TIME(ms)
PATHNAME    READS     WRITES      READS     WRITES     READS   WRITES
sdf          1088          0     557056k          0      0.00     0.00
```

还可以指定显示统计数据的次数和时间间隔。下面的命令以两秒钟两次的频率显示路径的增量式统计数据：

```
# vxdmpadm iostat show pathname=sdk interval=2 count=2
                   cpu usage = 9621us    per cpu memory = 266240b
               OPERATIONS            BLOCKS           AVG TIME(ms)
PATHNAME    READS    WRITES     READS    WRITES     READS    WRITES
sdk         0         0         0         0          0.00      0.00

sdk         0         0         0         0          0.00      0.00
```

显示排队的或错误的 I/O 的统计数据

使用 `vxdmpadm iostat show` 命令与 `-q` 选项可显示指定的 DMP 节点上或指定的路径或控制器上在 DMP 中排队的 I/O。对于 DMP 节点，`-q` 选项可显示指定的 DMP 节点上已发送到基础层的 I/O。如果指定了路径或控制器，则 `-q` 选项将显示已发送到指定路径或控制器上但未返回到 DMP 的 I/O。

有关 `vxdmpadm iostat` 命令的详细信息，请参阅 `vxdmpadm`(1m) 手册页。

显示 DMP 节点上排队的 I/O 计数：

 # vxdmpadm -q iostat show [*filter*] [interval=*n* [count=*m*]]

例如：

```
# vxdmpadm -q iostat show dmpnodename=emc_clariion0_352

cpu usage = 338us    per cpu memory = 102400b
                        QUEUED I/Os      PENDING I/Os
DMPNODENAME          READS    WRITES
emc_clariion0_352    0        0         0
```

显示 DMP 节点、路径或控制器上返回了错误的 I/O 计数：

```
# vxdmpadm -e iostat show [filter]
[interval=n [count=m]]
```

例如，显示路径上返回了错误的 I/O 计数：

```
# vxdmpadm -e iostat show pathname=sdo

cpu usage = 637us    per cpu memory = 102400b
                  ERROR I/Os
PATHNAME      READS    WRITES
sdo           0         0
```

显示累积的 I/O 统计数据

使用 `vxdmpadm iostat` 命令的 `groupby` 子句可按 DMP 节点、控制器、阵列端口 ID、主机阵列控制器对和磁盘阵列来显示累积的 I/O 统计数据列表。如果未指定 `groupby` 子句，则按路径显示统计数据。

默认情况下，读取/写入时间以毫秒为单位显示，并最大精确到 2 个小数位。 吞吐量数据以“块”为单位显示，并且输出按比例确定，即小值以小单位显示，大值以大单位显示，从而使有效数位保持不变。 可以指定显示统计数据所采用的单位。 `-u` 选项接受以下选项：

`h or H`	以可能的最高单位显示吞吐量。
`k`	以 KB 为单位显示吞吐量。
`m`	以 MB 为单位显示吞吐量。
`g`	以 GB 显示吞吐量。
`bytes| b`	显示以确切字节数为单位的吞吐量。
`us`	显示以微秒为单位的平均读/写时间。

按 DMP 节点分组：

**# vxdmpadm [-u unit] iostat show groupby=*dmpnode* **
[all | dmpnodename=*dmpnodename* | enclosure=*enclr-name*]

按控制器分组：

 # vxdmpadm [-u unit] iostat show groupby=ctlr [all | ctlr=*ctlr*]

例如：

 # vxdmpadm iostat show groupby=ctlr ctlr=c5

```
             OPERATIONS         BLOCKS        AVG TIME(ms)
CTLRNAME   READS    WRITES   READS   WRITES   READS    WRITES
c5         224       14      54       7        4.20     11.10
```

按阵列端口分组：

**# vxdmpadm [-u unit] iostat show groupby=arrayport [all **
| pwwn=*array_pwwn* | enclosure=*enclr* portid=*array-port-id*]

例如：

**# vxdmpadm -u m iostat show groupby=arrayport **
enclosure=HDS9500-ALUA0 portid=1A

```
             OPERATIONS         BYTES        AVG TIME(ms)
PORTNAME   READS    WRITES   READS   WRITES   READS    WRITES
1A         743      1538     11m     24m      17.13    8.61
```

按磁盘阵列分组：

**# vxdmpadm [-u unit] iostat show groupby=enclosure [all **
| enclosure=*enclr*]

例如：

 # vxdmpadm -u h iostat show groupby=enclosure enclosure=EMC_CLARiiON0

```
                OPERATIONS        BLOCKS        AVG TIME(ms)
ENCLRNAME     READS   WRITES   READS   WRITES   READS    WRITES
EMC_CLARiiON 743      1538     11392k  24176k 17.13    8.61
```

您也可以过滤出所有数据条目为零的实体。此选项在包含大量故障转移设备的集群环境中特别有用。您可以只显示活动路径的统计数据。

从 `iostat show` 命令的输出中过滤所有零条目：

vxdmpadm [-u unit] -z iostat show [all|ctlr=*ctlr_name* | dmpnodename=*dmp_device_name* | enclosure=*enclr_name* [portid=*portid*] | pathname=*path_name*|pwwn=*port_WWN*][interval=*seconds* [count=*N*]]

例如：

vxdmpadm -z iostat show dmpnodename=emc_clariion0_893

```
cpu usage = 9852us    per cpu memory = 266240b
             OPERATIONS       BLOCKS           AVG TIME(ms)
PATHNAME  READS  WRITES   READS   WRITES   READS    WRITES
sdbc      32     0        258     0         0.04      0.00
sdbw      27     0        216     0         0.03      0.00
sdck      8      0        57      0         0.04      0.00
sdde      11     0        81      0         0.15      0.00
```

显示以微秒为单位的平均读/写时间。

vxdmpadm -u us iostat show pathname=sdck

```
cpu usage = 9865us    per cpu memory = 266240b
              OPERATIONS       BLOCKS           AVG TIME(us)
PATHNAME   READS    WRITES  READS    WRITES   READS    WRITES
sdck       8        0       57       0        43.04      0.00
```

设置磁盘阵列路径的属性

可以使用 `vxdmpadm setattr` 命令设置磁盘阵列路径的属性。

为路径设置的属性是持久性的，它们存储在 `/etc/vx/dmppolicy.info` 文件中。

可设置以下属性：

`active` 将备用（故障转移）路径更改为活动路径。以下示例为阵列指定了一个活动路径：

vxdmpadm setattr path sde pathtype=active

`nomanual` 还原路径的原始主要属性或次要属性。本示例还原指向 JBOD 磁盘的路径：

vxdmpadm setattr path sdm pathtype=nomanual

`nopreferred`

还原路径的正常优先级。下面的示例还原路径的默认优先级：

```
# vxdmpadm setattr path sdk \
 pathtype=nopreferred
```

`preferred [priority=N]`

指定首选路径并为其分配优先级编号。如果指定，优先级编号必须是大于或等于 1 的整数。优先级编号越大，表示路径的 I/O 负载携带能力越强。

请参见第 63 页的“指定 I/O 策略”。

以下示例先将主动/主动磁盘阵列的 I/O 策略设置为 `priority`，然后使用指定的优先级 2 指定首选路径：

```
# vxdmpadm setattr enclosure enc0 \
 iopolicy=priority
# vxdmpadm setattr path sdk pathtype=preferred \
 priority=2
```

`primary`

将路径定义为 JBOD 磁盘阵列的主路径。以下示例为 JBOD 磁盘阵列指定了一个主路径：

```
# vxdmpadm setattr path sdm pathtype=primary
```

`secondary`

将路径定义为 JBOD 磁盘阵列的次级路径。下面的示例为 JBOD 磁盘阵列指定一个次级路径：

```
# vxdmpadm setattr path sdn pathtype=secondary
```

`standby`

将备用（故障转移）路径标记为不用于常规的 I/O 计划。如果没有可用于 I/O 的活动路径，则使用此路径。下一个示例为 A/P-C 磁盘阵列指定一个备用路径：

```
# vxdmpadm setattr path sde pathtype=standby
```

显示设备或磁盘阵列的冗余级别

使用 `vxdmpadm getdmpnode` 命令列出冗余级别小于所需级别的设备。

要列出指定磁盘阵列上启用路径数少于指定数量的设备，请使用以下命令：

```
# vxdmpadm getdmpnode enclosure=encl_name redundancy=value
```

例如，要列出启用路径数少于 3 个的设备，请使用以下命令：

```
# vxdmpadm getdmpnode enclosure=EMC_CLARiiON0 redundancy=3

NAME               STATE    ENCLR-TYPE   PATHS ENBL DSBL ENCLR-NAME
===================================================================
emc_clariion0_162 ENABLED  EMC_CLARiiON 3     2    1    emc_clariion0
emc_clariion0_182 ENABLED  EMC_CLARiiON 2     2    0    emc_clariion0
emc_clariion0_184 ENABLED  EMC_CLARiiON 3     2    1    emc_clariion0
emc_clariion0_186 ENABLED  EMC_CLARiiON 2     2    0    emc_clariion0
```

要显示某个特定设备的最小冗余级别，请使用 `vxdmpadm getattr` 命令，如下所示：

```
# vxdmpadm getattr enclosure|arrayname|arraytype \
component-name redundancy
```

例如，要显示磁盘阵列 HDS9500-ALUA0 的最小冗余级别：

```
# vxdmpadm getattr enclosure HDS9500-ALUA0 redundancy

ENCLR_NAME   DEFAULT  CURRENT
============================================
 HDS9500-ALUA0    0      4
```

指定最小活动路径数

可以为设备或磁盘阵列设置最小冗余级别。最小冗余级别是设备或磁盘阵列的最小活动路径数。如果磁盘阵列的路径数小于最小冗余级别，则会发送一条消息到系统控制台，并记录到 DMP 日志文件中。另外，也会向 `vxnotify` 客户端发送通知。

最小冗余级别的设置值存储在 `dmppolicy.info` 文件中，并持久保留。如果未设置最小冗余级别，则默认值是 0。

可以使用 `vxdmpadm setattr` 命令设置最小冗余级别。

指定最小活动路径数

◆ 使用 `vxdmpadm setattr` 命令，其中冗余属性如下所示：

```
# vxdmpadm setattr enclosure|arrayname|arraytype component-name
redundancy=value
```

其中：*value* 为活动路径数。

例如，要设置磁盘阵列 HDS9500-ALUA0 的最小冗余级别：

```
# vxdmpadm setattr enclosure HDS9500-ALUA0 redundancy=2
```

显示 I/O 策略

要显示磁盘阵列、阵列或阵列类型的 I/O 策略的当前设置和默认设置，请使用 vxdmpadm getattr 命令。

下面的示例显示 JBOD 磁盘的默认及当前 iopolicy 设置：

```
# vxdmpadm getattr enclosure Disk iopolicy

ENCLR_NAME        DEFAULT         CURRENT
---------------------------------------
Disk              MinimumQ        Balanced
```

下一个示例显示磁盘阵列 enc0 的 partitionsize 设置，在该磁盘阵列上，已使用 2MB 的分区大小设置了 balanced I/O 策略：

```
# vxdmpadm getattr enclosure enc0 partitionsize

ENCLR_NAME      DEFAULT         CURRENT
---------------------------------------
enc0            512             4096
```

指定 I/O 策略

可以使用 vxdmpadm setattr 命令更改在磁盘阵列的多条路径之间分摊 I/O 负载的 I/O 策略。可以为某个磁盘阵列（例如 HDS01）、特定类型的所有磁盘阵列（例如 HDS）或者特定阵列类型的所有磁盘阵列（A/A 表示主动/主动，或 A/P 表示主动/被动）设置策略。

警告： I/O 策略记录在 /etc/vx/dmppolicy.info 文件中，并且在系统重新启动后持久保留。

请不要自己编辑此文件。

可以设置下列策略：

`adaptive`

此策略通过动态调度路径上的I/O，尝试最大限度地提高磁盘的总 I/O 吞吐量。如果 I/O 负载随时间变化，建议使用此策略。例如，数据库的I/O可能同时表现为慢速传输（表扫描）和快速传输（随机查找）。对于不同路径可能有不同跃点数的 SAN 环境，此策略也很有用。由于此策略是由 DMP 自动管理的，因此无法进行进一步的配置。

在此示例中，为磁盘阵列 `enc1` 设置了 adaptive I/O 策略：

```
# vxdmpadm setattr enclosure enc1 \
 iopolicy=adaptive
```

`adaptiveminq`

它与 `adaptive` 策略非常相似，但根据每个路径中的 I/O 队列的长度进行 I/O 调度。队列路径越短，为其分配的优先级越高。

`balanced [partitionsize=size]`

该策略是专为优化磁盘驱动器以及 RAID 控制器中缓存的使用而设计的。缓存大小通常在 120KB 到 500KB 或更大范围内变化，具体取决于特定硬件的特性。在正常操作期间，在逻辑上将磁盘（或 LUN）分为若干个区域（或分区），并且仅在其中一条主动路径上发送给定区域的I/O。如果该路径失败，则自动在其余路径之间重新分配工作负载。

可以使用 partitionsize 属性的大小参数来指定分区大小。可以 2 的幂次方（从 2 到 231）为块单位来调整分区大小。不是 2 的幂的值自动向下舍入为最接近的可接受值。

将分区大小指定为 0 相当于指定默认分区大小。

分区大小的默认值为 512 块 (256k)。将分区大小指定为 0 相当于默认分区大小为 512 块 (256k)。

可以调整 `dmp_pathswitch_blks_shift` 可调参数的值来更改默认值。

请参见第 120 页的“DMP 可调参数”。

注意：如果将该值设置为大于缓存大小，就会失去此策略的优点。

例如，对于主要由顺序读写组成的 I/O 活动模式，建议将 Hitachi HDS 9960 A/A 阵列的分区大小设置为 32,768 到 131,072 块（16 MB 到 64 MB）。

下一个示例在磁盘阵列 enc0 上使用 4096 块 (2MB) 的分区大小设置 balanced I/O 策略：

```
# vxdmpadm setattr enclosure enc0 \
 iopolicy=balanced partitionsize=4096
```

`minimumq`

此策略在 LUN 队列中未完成的 I/O 请求数最少的路径上发送 I/O。由于 DMP 自动确定队列最短的路径，因此无法进行进一步的配置。

下面的示例将 JBOD 的 I/O 策略设置为 `minimumq`。

```
# vxdmpadm setattr enclosure Disk \
 iopolicy=minimumq
```

这是所有阵列的默认 I/O 策略。

`priority`

当 SAN 中的路径具有不同的性能，并且您想手动执行负载平衡时，此策略很有用。您可以根据可用路径的配置和性能特性以及系统其他特性，指定每条路径的优先级。

请参见第 60 页的“设置磁盘阵列路径的属性”。

在此示例中，将所有 SENA 阵列的 I/O 策略设置为 `priority`：

```
# vxdmpadm setattr arrayname SENA \
 iopolicy=priority
```

`round-robin`

此策略按循环顺序在路径之间平均分配 I/O。例如，如果有三条路径，则第一个 I/O 请求使用一条路径，第二个 I/O 请求使用另一条路径，第三个 I/O 请求沿着剩下的那条路径发送，第四个 I/O 请求沿着第一条路径发送，依此类推。由于此策略是由 DMP 自动管理的，因此无法进行进一步的配置。

下一个示例将所有主动/主动磁盘阵列的 I/O 策略设置为 `round-robin`：

```
# vxdmpadm setattr arraytype A/A \
 iopolicy=round-robin
```

`singleactive`

此策略沿一条活动路径路由 I/O。对于每个控制器只有一条活动路径、而其他路径则用于故障转移的 A/P 阵列，可以配置此策略。如果为 A/A 阵列配置了此策略，那么各路径之间不会有负载平衡，而且备用路径仅用于提供高可用性 (HA)。如果当前的活动路径出现故障，I/O 就会被切换到备用活动路径。由于 DMP 选择单个活动路径，因此无法进行进一步的配置。

下面的示例将 JBOD 磁盘的 I/O 策略设置为 `singleactive`：

```
# vxdmpadm setattr arrayname Disk \
 iopolicy=singleactive
```

安排非对称主动/主动阵列路径或 ALUA 阵列路径上的 I/O

可以结合 adaptive、balanced、minimumq、priority 与 round-robin I/O 策略指定 use_all_paths 属性，从而指定是否将 I/O 请求排定到非对称主动/主动 (A/A-A) 阵列或 ALUA 阵列的主路径以及辅助路径上。根据阵列的特性，随之改进的负载平衡可以提升总的 I/O 吞吐量。但只能根据阵列供应商的建议启用该功能。它对 A/A-A 或 ALUA 以外的阵列类型不起作用。

例如，以下命令利用 4096 块 (2MB) 的分区大小在磁盘阵列 enc0 上设置 balanced I/O 策略，并允许在辅助路径上排定 I/O 请求：

```
# vxdmpadm setattr enclosure enc0 iopolicy=balanced \
  partitionsize=4096 use_all_paths=yes
```

该属性的默认设置为 use_all_paths=no。

可为磁盘阵列、阵列名称或阵列类型显示 use_all_paths 的当前设置。要执行此操作，请指定 use_all_paths 选项的情况下执行 vxdmpadm gettattr 命令。

```
# vxdmpadm getattr enclosure HDS9500-ALUA0 use_all_paths


ENCLR_NAME    ATTR_NAME     DEFAULT CURRENT
============================================
HDS9500-ALUA0 use_all_paths no      yes
```

use_all_paths 属性只适用于 A/A-A 阵列和 ALUA 阵列。对于其他阵列，上述命令将显示以下消息：

```
Attribute is not applicable for this array.
```

在 SAN 中应用负载平衡的示例

此示例介绍了如何在 SAN 环境（其中有多条通过多个 SAN 交换机到达主动/被动设备的主路径）中配置负载平衡。

如 vxdisk list 命令的此示例输出所示，设备 sdm 有 8 条主路径：

```
# vxdisk list sdq


Device: sdq
    .
    .
    .
numpaths: 8
sdj state=enabled type=primary
sdk state=enabled type=primary
```

```
sdl state=enabled type=primary
sdm state=enabled type=primary
sdn state=enabled type=primary
sdo state=enabled type=primary
sdp state=enabled type=primary
sdq state=enabled type=primary
```

此外，设备位于磁盘组 mydg 的磁盘阵列 ENC0 中，并且包含一个简单的连续卷 myvol1。

第一步是启用 DMP 统计信息收集功能：

```
# vxdmpadm iostat start
```

下一步使用 dd 命令应用卷中的输入工作负载：

```
# dd if=/dev/vx/rdsk/mydg/myvol1 of=/dev/null &
```

通过运行 vxdmpadm iostat 命令来显示设备的 DMP 统计信息，可以看出，所有 I/O 都被定向到一条路径，即 sdq：

```
# vxdmpadm iostat show dmpnodename=sdq interval=5 count=2
    .
    .
    .
cpu usage = 11294us per cpu memory = 32768b
             OPERATIONS           KBYTES         AVG TIME(ms)
PATHNAME   READS   WRITES   READS    WRITES   READS      WRITES
sdj        0       0        0        0         0.00        0.00
sdk        0       0        0        0         0.00        0.00
sdl        0       0        0        0         0.00        0.00
sdm        0       0        0        0         0.00        0.00
sdn        0       0        0        0         0.00        0.00
sdo        0       0        0        0         0.00        0.00
sdp        0       0        0        0         0.00        0.00
sdq        10986   0        5493     0         0.41        0.00
```

使用 vxdmpadm 命令显示包含该设备的磁盘阵列的 I/O 策略：

```
# vxdmpadm getattr enclosure ENC0 iopolicy

ENCLR_NAME      DEFAULT           CURRENT
============================================
ENC0            MinimumQ       Single-Active
```

此信息表明磁盘阵列的策略被设置为 `singleactive`，这就是所有 I/O 都发生在一条路径上的原因。

为了在多条主路径之间平衡 I/O 负载，该策略设置为 `round-robin`，如下所示：

```
# vxdmpadm setattr enclosure ENC0 iopolicy=round-robin
# vxdmpadm getattr enclosure ENC0 iopolicy

ENCLR_NAME     DEFAULT             CURRENT
============================================
ENC0           MinimumQ            Round-Robin
```

现在重置 DMP 统计信息：

```
# vxdmpadm iostat reset
```

由于工作负载还在运行，现在可以看到将 I/O 策略更改为在主路径之间平衡负载的效果。

```
# vxdmpadm iostat show dmpnodename=sdq interval=5 count=2
     .
     .
     .
cpu usage = 14403us per cpu memory = 32768b
             OPERATIONS            KBYTES         AVG TIME(ms)
PATHNAME    READS    WRITES    READS     WRITES   READS      WRITES
sdj         2041     0         1021      0         0.39        0.00
sdk         1894     0         947       0         0.39        0.00
sdl         2008     0         1004      0         0.39        0.00
sdm         2054     0         1027      0         0.40        0.00
sdn         2171     0         1086      0         0.39        0.00
sdo         2095     0         1048      0         0.39        0.00
sdp         2073     0         1036      0         0.39        0.00
sdq         2042     0         1021      0         0.39        0.00
```

通过输入下面的命令，可将磁盘阵列的策略返回到单个活动 I/O 策略：

```
# vxdmpadm setattr enclosure ENC0 iopolicy=singleactive
```

对路径、控制器、阵列端口或 DMP 节点禁用 I/O

禁用通过路径、HBA 控制器、阵列端口或 DMP 节点的 I/O 将阻止 DMP 通过指定路径或者与指定控制器、阵列端口或 DMP 节点连接的路径发出 I/O 请求。在通过路径发布的所有暂停的 I/O 请求全部完成之前，该命令将一直被阻止。

DMP 不支持通过该操作对使用第三方驱动程序 (TPD) 进行多径处理的控制器禁用 I/O。

要对一个或多个路径禁用 I/O，请使用以下命令：

```
# vxdmpadm [-c|-f] disable path=path_name1[,path_name2,path_nameN]
```

要对连接到一个或多个 HBA 控制器的路径禁用 I/O，请使用以下命令：

```
# vxdmpadm [-c|-f] disable ctlr=ctlr_name1[,ctlr_name2,ctlr_nameN]
```

要禁用连接到磁盘阵列端口的路径的 I/O，请使用以下某个命令：

```
# vxdmpadm [-c|-f] disable enclosure=enclr_name portid=array_port_ID
# vxdmpadm [-c|-f] disable pwwn=array_port_WWN
```

其中，磁盘阵列端口通过磁盘阵列名称与磁盘阵列端口 ID 或者磁盘阵列端口的 WWN 标识符指定。

以下示例说明如何禁用阵列端口上的 I/O：

```
# vxdmpadm disable enclosure=HDS9500V0 portid=1A
# vxdmpadm disable pwwn=20:00:00:E0:8B:06:5F:19
```

要对特定路径禁用 I/O，请指定控制器和端口 ID，它们表示光纤的两端：

```
# vxdmpadm [-c|-f] disable ctlr=ctlr_name enclosure=enclr_name \
portid=array_port_ID
```

要对特定的 DMP 节点禁用 I/O，请指定该 DMP 节点名称。

```
# vxdmpadm [-c|-f] disable dmpnodename=dmpnode
```

可以使用 -c 选项来检查是否只有一条活动路径通达磁盘。如果情况属实，`disable` 命令都将失败并显示一条错误消息，直到您使用`-f` 选项强制禁用该路径为止。

如果对通过单一路径连接到根磁盘的控制器发布`disable`命令且备用路径上没有配置根磁盘镜像，该命令就会失败。如果存在此类镜像，该命令就会成功。

为路径、控制器、阵列端口或 DMP 节点启用 I/O

启用控制器将允许以前被禁用的路径、HBA 控制器、阵列端口或 DMP 节点再次接受I/O。只有可通过路径、控制器、阵列端口或DMP节点访问主机且可在其上执行I/O操作时，此操作才会成功。在连接主动/被动磁盘阵列时，`enable`操作将使I/O故障回复到主路径。`enable`操作还可用于允许先前分离的系统板上的控制器的I/O。

注意：用于访问配置了集群共享磁盘组的磁盘阵列的控制器支持该操作。

DMP 不支持通过该操作为使用第三方驱动程序 (TPD) 进行多径处理的控制器启用 I/O。

要为一个或多个路径启用 I/O，请使用以下命令：

```
# vxdmpadm enable path=path_name1[,path_name2,path_nameN]
```

要为连接到一个或多个 HBA 控制器的路径启用 I/O，请使用以下命令：

```
# vxdmpadm enable ctlr=ctlr_name1[,ctlr_name2,ctlr_nameN]
```

要启用连接到磁盘阵列端口的路径的 I/O，请使用以下某个命令：

```
# vxdmpadm enable enclosure=enclr_name portid=array_port_ID
# vxdmpadm enable pwwn=array_port_WWN
```

其中，磁盘阵列端口通过磁盘阵列名称与磁盘阵列端口 ID 或者磁盘阵列端口的 WWN 标识符指定。

下面是采用命令来启用磁盘阵列端口上的 I/O 的示例：

```
# vxdmpadm enable enclosure=HDS9500V0 portid=1A
# vxdmpadm enable pwwn=20:00:00:E0:8B:06:5F:19
```

要为特定路径启用 I/O，请指定控制器和端口 ID，它们表示光纤的两端：

```
# vxdmpadm enable ctlr=ctlr_name enclosure=enclr_name \
 portid=array_port_ID
```

要为特定的 DMP 节点启用 I/O，请指定该 DMP 节点名称。

```
# vxdmpadm enable dmpnodename=dmpnode
```

重命名磁盘阵列

`vxdmpadm setattr` 命令可用于为现有磁盘阵列分配有意义的名称，例如：

```
# vxdmpadm setattr enclosure enc0 name=GRP1
```

该示例将磁盘阵列的名称由 `enc0` 更改为 `GRP1`。

注意：磁盘阵列名称前缀的最大长度为 23 个字符。

以下命令显示更改后的名称：

```
# vxdmpadm listenclosure all
```

```
ENCLR_NAME     ENCLR_TYPE     ENCLR_SNO                  STATUS
============================================================
other0         OTHER         OTHER_DISKS                CONNECTED
jbod0          X1            X1_DISKS                   CONNECTED
GRP1           ACME          60020f20000001a90000       CONNECTED
```

配置对 I/O 故障的响应

可配置 DMP 对指定磁盘阵列、磁盘阵列名称或阵列类型路径上出现故障的 I/O 请求的响应方式。默认情况下，DMP 配置为对单一路径最多重试五次出现故障的 I/O 请求。

要显示用来处理应用于磁盘阵列、阵列名称或阵列类型路径的 I/O 请求故障的当前设置，请使用 `vxdmpadm getattr` 命令。

请参见第 74 页的“显示恢复选项值”。

若要设置 DMP 在路径上重试发送 I/O 请求的次数限制，请使用如下命令：

```
# vxdmpadm setattr \
 {enclosure enc-name|arrayname name|arraytype type} \
 recoveryoption=fixedretry retrycount=n
```

`retrycount` 的参数值指定重试多少次之后 DMP 才把 I/O 请求重新排定到另一条可用的路径上或者彻底地宣告请求失败。

作为有别于指定固定重试次数的另外一种选择，您可以指定 DMP 允许处理 I/O 请求的时间长度。如果在该时间内 I/O 请求没有成功，DMP 会宣告该 I/O 请求失败。要指定 `iotimeout` 值，请使用以下命令：

```
# vxdmpadm setattr \
 {enclosure enc-name|arrayname name|arraytype type} \
 recoveryoption=timebound iotimeout=seconds
```

默认的 `iotimeout` 值为 300 秒。对于某些应用程序（如 Oracle），需要将 `iotimeout` 设为较大的值。DMP 的 `iotimeout` 值应该大于基础操作系统层的 I/O 服务时间。

注意： `fixedretry` 与 `timebound` 设置相互排斥。

下例配置磁盘阵列 `enc0` 的限时恢复，并把 `iotimeout` 值设成 360 秒：

```
# vxdmpadm setattr enclosure enc0 recoveryoption=timebound \
 iotimeout=360
```

下例将所有主动/主动磁盘阵列路径的固定重试上限均设置为 10 次：

```
# vxdmpadm setattr arraytype A/A recoveryoption=fixedretry \
 retrycount=10
```

指定 `recoveryoption=default` 会把 DMP 重置为对应于 `recoveryoption=fixedretry retrycount=5` 的默认设置，例如：

```
# vxdmpadm setattr arraytype A/A recoveryoption=default
```

上述命令还具备将 I/O 控制配置为默认设置的效果。

请参见第 72 页的"配置 I/O 控制机制"。

注意：响应 I/O 故障的设置在重新启动系统后依然保留。

配置 I/O 控制机制

默认情况下，DMP 配置为关闭所有路径的 I/O 控制。要显示用来处理应用于磁盘阵列路径的 I/O 控制的当前设置，请使用 `vxdmpadm getattr` 命令。

请参见第 74 页的"显示恢复选项值"。

如果启用，由于收集统计信息的后台驻留程序的活动，I/O 控制会占用少量 CPU 与内存资源。如果禁用 I/O 控制，后台驻留程序不再收集统计信息，在重新启用 I/O 控制之前一直处于非活动状态。

要关闭 I/O 控制，请按如下格式使用 `vxdmpadm setattr` 命令：

```
# vxdmpadm setattr \
 {enclosure enc-name|arrayname name|arraytype type} \
 recoveryoption=nothrottle
```

下例显示如何为磁盘阵列 `enc0` 的路径禁用 I/O 控制：

```
# vxdmpadm setattr enclosure enc0 recoveryoption=nothrottle
```

`vxdmpadm setattr` 命令可以用来启用指定磁盘阵列、磁盘阵列名称或磁盘阵列类型路径上的 I/O 控制：

```
# vxdmpadm setattr \
 {enclosure enc-name|arrayname name|arraytype type}\
 recoveryoption=throttle [iotimeout=seconds]
```

如果指定了 `iotimeout` 属性，其参数将以秒为单位指定时间，DMP 等待待处理的 I/O 请求该时间之后就在该路径上实施 I/O 控制。默认的 `iotimeout` 值为 10 秒。将 `iotimeout` 设成较大值会导致 SCSI 驱动程序中有更多的 I/O 请求排队等候之后再实施 I/O 控制。

下面的示例将磁盘阵列 enc0 的 iotimeout 值设置为 60 秒：

```
# vxdmpadm setattr enclosure enc0 recoveryoption=throttle \
 iotimeout=60
```

指定 recoveryoption=default 会把 I/O 控制重置为默认设置，如下所示：

```
# vxdmpadm setattr arraytype A/A recoveryoption=default
```

上述命令配置与 recoveryoption=nothrottle 对应的默认行为。上述命令还配置对 I/O 故障做出响应时的默认行为。

请参见第 71 页的"配置对 I/O 故障的响应"。

注意：I/O 控制设置在重新启动系统后依然保留。

配置子路径故障转移组 (SFG)

可以使用可调参数 dmp_sfg_threshold 打开或关闭子路径故障转移组 (SFG) 功能。

要关闭该功能，请将可调参数 dmp_sfg_threshold 的值设置为 0：

```
# vxdmpadm settune dmp_sfg_threshold=0
```

要打开该功能，请将 dmp_sfg_threshold 的值设置为触发 SFG 所需的路径故障数。默认值为 1。

```
# vxdmpadm settune dmp_sfg_threshold=N
```

此可调参数的默认值为 1，表示该功能已打开。

要看到子路径故障转移组 ID，请使用以下命令：

```
# vxdmpadm getportids {ctlr=ctlr_name | dmpnodename=dmp_device_name \
  | enclosure=enclr_name | path=path_name}
```

配置低影响路径探测

可以使用 vxdmpadm settune 命令打开或关闭低影响路径探测 (LIPP) 功能：

```
# vxdmpadm settune dmp_low_impact_probe=[on|off]
```

通过探查连接到同一 HBA 和阵列端口的某个路径子集，路径探查将得到优化。路径子集的大小可以由 dmp_probe_threshold 可调参数控制。默认值设置为 5。

```
# vxdmpadm settune dmp_probe_threshold=N
```

显示恢复选项值

要显示用来处理应用于磁盘阵列、阵列名称或阵列类型路径的 I/O 请求故障的当前设置，请使用以下命令：

```
# vxdmpadm getattr \
{enclosure enc-name|arrayname name|arraytype type} \
recoveryoption
```

下例介绍了用来显示在磁盘阵列上设置的 `recoveryoption` 选项值的 `vxdmpadm getattr` 命令。

```
# vxdmpadm getattr enclosure HDS9500-ALUA0 recoveryoption
ENCLR-NAME      RECOVERY-OPTION  DEFAULT[VAL]    CURRENT[VAL]
================================================================
HDS9500-ALUA0   Throttle         Nothrottle[0]   Timebound[60]
HDS9500-ALUA0   Error-Retry      Fixed-Retry[5]  Timebound[20]
```

这里显示了默认的和当前的策略选项及其值。

表 3-1 总结了出错后重试 I/O 的可能恢复选项设置。

表 3-1 出错后重试 I/O 的恢复选项

恢复选项	可能的设置	说明
recoveryoption=fixedretry	Fixed-Retry (retrycount)	如果 I/O 出现故障，DMP 就对出现故障的 I/O 请求重试指定的次数。
recoveryoption=timebound	Timebound (iotimeout)	如果 I/O 出现故障，DMP 对出现故障的 I/O 请求重试指定的时间（以秒为单位）。

表 3-2 总结了控制 I/O 的可能恢复选项设置。

表 3-2 I/O 控制的恢复选项

恢复选项	可能的设置	说明
recoveryoption=nothrottle	无	未使用 I/O 控制。
recoveryoption=throttle	Timebound (iotimeout)	如果在指定时间内（以秒为单位）I/O 请求没有返回，DMP 就对路径实施限制。

配置 DMP 路径还原策略

DMP 提供一个内核线程，此线程以指定间隔重新检查路径的状况。 对路径进行何种分析取决于所配置的检查策略。

注意：DMP 路径还原线程不会通过已用 `vxdmpadm disable` 禁用的控制器来更改路径的禁用状态。

配置 DMP 路径还原策略时，必须停止路径还原线程，然后使用新属性重新启动它。

请参见第 76 页的"停止 DMP 路径还原线程"。

使用 `vxdmpadm settune dmp_restore_policy` 命令来配置以下某个还原策略。 在停止还原线程或者使用 `vxdmpadm settune` 命令更改值之前，该策略将一直有效。

- `check_all`

  路径还原线程将分析系统中的所有路径，重新启用返回联机状态的路径，并禁用不可访问的路径。 配置该策略的命令如下：

  # **vxdmpadm settune dmp_restore_policy=check_all**

- `check_alternate`

  路径还原线程检查是否至少有一个备用路径情况正常。如果不满足此条件，则生成一个通知。在有大量可用路径的情况下，该策略避免对所有正常路径运行查询命令，并且比 `check_all` 的开销小。如果每个 DMP 节点仅有两个路径，则该策略与 `check_all` 相同。 配置该策略的命令如下：

  # **vxdmpadm settune dmp_restore_policy=check_alternate**

- `check_disabled`

  这是默认的路径还原策略。路径还原线程检查先前由于硬件故障而禁用的路径的情况，如果这些路径已返回联机状态，则重新启用它们。 配置该策略的命令如下：

  # **vxdmpadm settune dmp_restore_policy=check_disabled**

- `check_periodic`

  路径还原线程在给定数目的循环中执行一次 `check_all`，而在其余的循环中执行 `check_disabled`。 如果有大量可用路径，则该策略可能导致速度周期性地变慢（由于执行 `check_all`）。 配置该策略的命令如下：

  # **vxdmpadm settune dmp_restore_policy=check_periodic**

  前后两次运行 `check_all` 策略之间的默认循环次数为 10。

dmp_restore_interval 可调参数指定路径还原线程检查路径的频率。例如，以下命令将轮询间隔设置为 400 秒：

```
# vxdmpadm settune dmp_restore_interval=400
```

将立即应用设置，并且在重新启动后保持不变。使用 vxdmpadm gettune 查看当前设置。

请参见第 120 页的“DMP 可调参数”。

如果在未指定策略或间隔的情况下发出 vxdmpadm start restore 命令，将使用先前由管理员利用 vxdmpadm settune 命令设置的持久性策略和间隔设置来启动路径还原线程。如果管理员没有设置策略或间隔，将使用系统默认值。系统默认还原策略为 check_disabled。系统默认间隔为 300 秒。

警告：将间隔减小到系统默认值以下可能会对系统性能产生负面影响。

停止 DMP 路径还原线程

使用此命令停止 DMP 路径还原线程：

```
# vxdmpadm stop restore
```

警告：如果停止路径还原线程，则自动路径故障回复将停止。

显示 DMP 路径还原线程的状态

使用 vxdmpadm gettune 命令显示的可调参数值可以显示出 DMP 路径还原线程的状态。这些可调参数包括：

dmp_restore_state　自动路径还原内核线程的状态。

dmp_restore_interval　DMP 路径还原线程的轮询间隔。

dmp_restore_policy　供 DMP 用来检查路径情况的策略。

显示 DMP 路径还原线程的状态

◆ 请使用以下命令：

```
# vxdmpadm gettune dmp_restore_state

# vxdmpadm gettune dmp_restore_interval

# vxdmpadm gettune dmp_restore_policy
```

配置阵列策略模块

阵列策略模块 (APM) 是一个可动态加载的内核模块（DMP 的插件），以与一个阵列结合使用。APM 定义特定于阵列的过程和命令以便：

- 当阵列内的磁盘有多条路径可用时，选择一条 I/O 路径。
- 选择路径故障转移机制。
- 一条路径出现故障时选择备用路径。
- 使路径更改生效。
- 响应 SCSI 保留或释放请求。

当注册阵列时，DMP 为这些功能提供默认过程。APM 可以修改 DMP 或其他版本的 APM 提供的全部或部分现有过程。

可以使用以下命令显示为某个系统配置的所有 APM：

```
# vxdmpadm listapm all
```

此命令的输出包括每个模块的文件名、支持的阵列类型、APM 名称、APM 版本和当前是否已加载并使用该模块。要查看个别模块的详细信息，请将该模块的名称指定为命令参数：

```
# vxdmpadm listapm module_name
```

若要添加和配置 APM，请使用以下命令：

```
# vxdmpadm -a cfgapm module_name [attr1=value1 \
  [attr2=value2 ...]]
```

可选配置属性及其值是阵列 APM 专用的。有关详细信息，请参阅阵列供应商提供的文档。

注意：默认情况下，DMP 使用可用的最新 APM。如果需要强制 DMP 使用早期版本的 APM，请指定 `-u` 选项而不是 `-a` 选项。仅在不使用当前版本的 APM 时才替换它。

通过指定 `-r` 选项，可以删除当前未加载的 APM：

```
# vxdmpadm -r cfgapm module_name
```

请参见 `vxdmpadm`(1M) 手册页。

4

管理磁盘

本章节包括下列主题：

- 关于磁盘管理
- 发现和配置新添加的磁盘设备
- 更改磁盘设备的命名机制
- 发现基于磁盘阵列的磁盘名称与基于操作系统的磁盘名称之间的关联关系

关于磁盘管理

Veritas Dynamic Multi-Pathing (DMP) 用于管理多端口磁盘阵列。

请参见第 12 页的“DMP 的工作原理”。

DMP 使用设备发现层 (DDL) 处理设备发现和磁盘阵列的配置。DDL 可发现 DMP 操作所需的磁盘及其属性。使用 `vxddladm` 实用程序可管理 DDL。

请参见第 84 页的“如何管理设备发现层”。

发现和配置新添加的磁盘设备

将新磁盘物理连接到主机或将新光纤通道设备分区到主机时，可以使用 `vxdctl enable` 命令来重建卷设备节点的目录，并更新 DMP 内部数据库以反映系统的新情况。

要重新配置 DMP 数据库，首先重新启动系统使得 Linux 可以识别这些新磁盘，然后调用 `vxdctl enable` 命令。

还可以使用 `vxdisk scandisks` 命令扫描操作系统设备树中的设备，并启动多径处理磁盘的动态重新配置操作。

如果希望 DMP 只扫描已添加到系统的新设备，而不扫描已启用或已禁用的设备，请为该命令指定 -f 选项，如下所示：

```
# vxdctl -f enable
# vxdisk -f scandisks
```

但是，如果因对下列项进行修改而修改了系统配置，则启动完整扫描：

- 安装的阵列支持库。
- 被 VxVM 排除使用的设备的列表。
- DISKS (JBOD)、SCSI3 或外部设备定义。

请参见 vxdctl(1M) 手册页。

请参见 vxdisk(1M) 手册页。

部分发现设备

动态多径处理(DMP)支持部分设备发现功能，即您可以在发现过程中包括或排除指向物理磁盘的路径。

vxdisk scandisks 命令重新扫描 OS 设备树中的设备，然后触发 DMP 重配置操作。可以为 vxdisk scandisks 命令指定参数以执行部分设备发现操作。 例如，使用以下命令可让 DMP 发现以前未知的新增设备：

```
# vxdisk scandisks new
```

下一个示例发现光纤设备：

```
# vxdisk scandisks fabric
```

下面的命令扫描设备 sdm 和 sdn：

```
# vxdisk scandisks device=sdm,sdn
```

或者，可以指定 ! 前缀字符来表示要扫描除已列出设备以外的所有设备。

注意：在某些 shell 中，! 字符是特殊字符。以下示例显示如何在 bash shell 中对其换码。

```
# vxdisk scandisks \!device=sdm,sdn
```

也可以扫描连接（或未连接）到一系列逻辑或物理控制器的设备。例如，以下命令发现并配置除指定逻辑控制器连接的设备之外的所有设备：

```
# vxdisk scandisks \!ctlr=c1,c2
```

下面的命令仅发现连接到指定物理控制器的那些设备：

```
# vxdisk scandisks pctlr=c1+c2
```

物理控制器的列表项之间用 + 字符分隔。

可以使用 `vxdmpadm getctlr all` 命令获取物理控制器列表。

只应为 `vxdisk scandisks` 命令指定一个选择参数。指定多个选项会导致出错。

请参见 `vxdisk`(1M) 手册页。

发现磁盘和动态添加磁盘阵列

DMP 使用阵列支持库 (ASL) 为多径处理提供特定于阵列的支持。阵列支持库 (ASL) 是一种可动态加载的共享库（DDL 的插件）。在设备发现期间 ASL 实现特定于硬件的逻辑以发现设备属性。DMP 提供设备发现层 (DDL)，用于确定应与每个磁盘阵列关联的 ASL。

在某些情况下，通过将 LUN 视为磁盘 (JBOD)，DMP 还可提供基本的多径处理和故障转移功能。

DMP 如何声明设备

为充分优化对任意阵列的支持以及对更复杂阵列类型的支持，DMP 要求使用特定于阵列的阵列支持库 (ASL)，该支持库可能与阵列策略模块 (APM) 紧密结合。ASL 和 APM 是特定于阵列的有效插件，允许 DMP 与任何特定阵列模型紧密结合。

请参见硬件兼容列表以了解受支持阵列的完整列表。

http://www.symantec.com/docs/TECH170013

在设备发现期间，DDL 检查每个设备上的安装 ASL，以查出哪个 ASL 声明了该设备。

如果未发现有哪个 ASL 声明了该设备，则 DDL 将检查对应的 JBOD 定义。可以为不受支持阵列添加 JBOD 定义，使 DMP 为阵列提供多径处理。如果找到 JBOD 定义，DDL 将在 DISKS 类别中声明设备，并将 LUN 添加到 DMP 使用的 JBOD（物理磁盘）设备列表中。如果 JBOD 定义包括一个压缩文件号，则 DDL 使用压缩文件号将 LUN 分组到磁盘阵列中。

请参见第 91 页的“将不支持的磁盘阵列添加到 DISKS 类别”。

即使没有 ASL 或 JBOD 定义，DMP 也可以为 ALUA 兼容的阵列提供基本的多径处理。DDL 将 LUN 声明为 aluadisk 磁盘阵列的一部分。阵列类型显示为 ALUA。通过添加 JBOD 定义还可以将 LUN 分组到磁盘阵列中。

磁盘类别

经过认证可用于 Veritas Dynamic Multi-Pathing 的磁盘阵列受阵列支持库 (ASL) 的支持，并按磁盘返回的供应商 ID 字符串（如 HITACHI）进行归类。

JBOD 中能用 DMP 进行多径处理的磁盘归入 DISKS 类别中。不支持的阵列中的磁盘也可以归入 DISKS 类别中。

请参见第 91 页的“将不支持的磁盘阵列添加到 DISKS 类别”。

JBOD 中不属于任何受支持的类别且无法用 DMP 进行多径处理的磁盘属于 OTHER_DISKS 类别。

添加对新磁盘阵列的支持

可以动态地添加对新磁盘阵列类型的支持。这种支持以 Symantec 所开发的阵列支持库 (ASL) 的形式出现。Symantec 通过更新 VRTSaslapm rpm来支持新磁盘阵列。要确定已更新的 VRTSaslapm rpm是否可以下载，请参考硬件兼容性列表技术说明。硬件兼容性列表提供了指向下载最新rpm的链接和安装 VRTSaslapm rpm的说明。当系统联机时可升级 VRTSaslapm rpm，且无需停止应用程序。

要访问硬件兼容性列表，请访问以下 URL：

http://www.symantec.com/docs/TECH170013

每个 VRTSaslapm rpm都特定于 Dynamic Multi-Pathing 版本。请务必安装支持 Dynamic Multi-Pathing 安装版本的 VRTSaslapm rpm。

在安装 VRTSaslapm rpm时，不需要先将新的磁盘阵列连接到系统。在 SLES11 系统上，如果新磁盘阵列中的任何一个磁盘随后与系统进行连接，并且如果 vxconfigd 正在运行，则 vxconfigd 会立即调用 DDL 设备发现，然后将新磁盘包含在 VxVM 设备列表中。对于其他风格的 Linux，重新启动系统以使 Linux 可以识别这些新磁盘，然后使用 vxdctl enable 命令将这些新磁盘包含在 VxVM 设备列表中。

如果需要删除最新的 VRTSaslapm rpm，可以恢复到原来的安装版本。有关详细过程，请参考《Veritas Storage Foundation and High Availability Solutions 故障排除指南》。

启用新磁盘阵列发现功能

vxdctl enable 命令会扫描所有磁盘设备及其属性，然后更新 DMP 设备列表，并使用新的设备数据库重新配置 DMP。不必重新启动主机。

警告：此命令可确保为阵列正确设置动态多径处理。否则，VxVM 将把磁盘的独立路径视为是单独的设备，从而导致数据损坏。

启用新磁盘阵列发现功能

◆ 键入以下命令：

```
# vxdctl enable
```

第三方驱动程序共存特性

通过 DMP 的第三方驱动程序 (TPD) 共存功能，可以在保留 DMP 的监视功能的同时，让某些第三方多径处理驱动程序控制的 I/O 绕过 DMP。只要提供且安装了合适的 ASL，您无需设置定义文件或运行特殊命令就可让系统发现使用 TPD 的设备。DMP 的 TPD 共存特性在不需要对第三方多径处理驱动程序做任何更改的情况下允许共存。

请参见第 97 页的“更改 TPD 控制的磁盘阵列的设备命名”。

请参见第 51 页的“显示有关 TPD 控制的设备的信息”。

自动发现 EMC Symmetrix 阵列

在 VxVM 4.0 中，可以通过两种方法来配置 EMC Symmetrix 阵列：

- 如果安装了 EMC PowerPath，则可将 EMC Symmetrix 阵列配置为外部设备。请参见第 94 页的“外部设备”。
- 如果没有安装 EMC PowerPath，则可使用 DMP 来执行多径处理。

将系统升级为 VxVM 4.1 或更高的版本之后，DDL 就可以发现现有的 EMC PowerPath 设备，并将它作为带 DMP 节点的自动配置的磁盘配置到 DMP 中，即使 PowerPath 正被用来执行多径处理。用户不需要将这类阵列作为外部设备进行配置。

表 4-1 显示了将 DMP 与 PowerPath 配合使用的方案。

ASL 全部包括在 ASL-APM rpm 中，该软件包在安装 Storage Foundation 产品时安装。

表 4-1 将 DMP 与 PowerPath 配合使用的方案

PowerPath	DMP	阵列配置模式
已安装。	`libvxpp` ASL 在内部处理 EMC Symmetrix 阵列与 DGC CLARiiON 请求。PowerPath 处理故障转移。	EMC Symmetrix - 任何 DGC CLARiiON - 主动/被动 (A/P)， 显式故障转移模式下的主动/被动 (A/P-F) 和 ALUA 显式故障转移

PowerPath	DMP	阵列配置模式
未安装；阵列为 EMC Symmetrix。	DMP 处理多径处理。 ASL 名称是 libvxemc。	主动/主动
未安装，阵列为 DGC CLARiioN (CXn00)。	DMP 处理多径处理。 ASL 名称是 libvxCLARiiON。	主动/被动 (A/P)、显式故障转移模式下的主动/被动 (A/P-F) 和 ALUA

如果将任意 EMCpower 磁盘配置为外部磁盘，请使用 vxddladm rmforeign 命令删除外来定义，如本例所示：

```
# vxddladm rmforeign blockpath=/dev/emcpowera10 \
 charpath=/dev/emcpowera10
```

为了让 DMP 能够收到正确的查询数据，必须将“通用序列号（C 位）Symmetrix 控制器”参数设置为启用。

如何管理设备发现层

设备发现层 (DDL) 允许动态添加磁盘阵列。DDL 可发现 DMP 操作所需的磁盘及其属性。

可以使用 vxddladm 实用程序管理 DDL 来执行下列任务：

- 列出 DDL 发现的所有设备（包括 iSCSI 设备）的层次结构。
- 列出包括 iSCSI 在内的所有主机总线适配器。
- 列出主机总线适配器上配置的端口。
- 列出从主机总线适配器配置的目标。
- 列出从主机总线适配器配置的设备。
- 获取或设置 iSCSI 操作参数。
- 列出支持的阵列类型。
- 将对阵列的支持添加到 DDL。
- 从 DDL 删除对阵列的支持。
- 列出排除的磁盘阵列的信息。
- 列出在 DISKS (JBOD) 类别中支持的磁盘。
- 将不同提供商的磁盘添加到 DISKS 类别。
- 从 DISKS 类别中删除磁盘。

■ 将磁盘添加为外部设备。

以下各节将详细阐述这些任务。

请参见 vxddladm(1M) 手册页。

列出包含 iSCSI 在内的所有设备

可以显示 DDL 搜索到的所有设备（包括 iSCSI 设备）的层次结构。

列出包括 iSCSI 在内的所有设备

◆ 键入以下命令：

```
# vxddladm list
```

以下是一个输出示例：

```
HBA fscsi0 (20:00:00:E0:8B:19:77:BE)
     Port fscsi0_p0 (50:0A:09:80:85:84:9D:84)
          Target fscsi0_p0_t0 (50:0A:09:81:85:84:9D:84)
               Device sda
. . .
HBA iscsi0 (iqn.1986-03.com.sun:01:0003ba8ed1b5.45220f80)
     Port  iscsi0_p0 (10.216.130.10:3260)
          Target  iscsi0_p0_t0 (iqn.1992-08.com.netapp:sn.84188548)
               Device sdb
               Device sdc
          Target  iscsi0_p0_t1 (iqn.1992-08.com.netapp:sn.84190939)
. . .
```

列出包括 iSCSI 在内的所有主机总线适配器

可以获取有关在系统上配置的所有主机总线适配器（包括 iSCSI 适配器）的信息。这些信息包括以下几项：

Driver	控制 HBA 的驱动程序。
Firmware	固件版本。
Discovery	针对目标使用的搜索方法。
State	设备是联机还是脱机。
Address	硬件地址。

列出包括 iSCSI 在内的所有主机总线适配器

◆ 使用以下命令可列出在系统上配置的所有 HBA（包括 iSCSI 设备）：

```
# vxddladm list hbas
```

列出主机总线适配器上配置的端口

可以获得在 HBA 上配置的所有端口的信息。 显示的信息包括下列几项：

HBA-ID	父 HBA。
State	设备是联机还是脱机。
Address	硬件地址。

列出主机总线适配器上配置的端口

◆ 使用以下命令获取 HBA 上配置的端口：

```
# vxddladm list ports
PortID   HBA-ID    State      Address
-------------------------------------------------------
c2_p0    c2        Online     50:0A:09:80:85:84:9D:84
c3_p0    c3        Online     10.216.130.10:3260
```

列出从主机总线适配器或端口配置的目标

可以获取有关从主机总线适配器或端口配置的所有目标的信息。这些信息包括以下几项：

Alias	别名（如果有）。
HBA-ID	父 HBA 或端口。
State	设备是联机还是脱机。
Address	硬件地址。

列出目标

◆ 要列出所有目标，请使用以下命令：

```
# vxddladm list targets
```

以下是一个输出示例：

```
TgtID      Alias HBA-ID State  Address
-----------------------------------------------------------------
c2_p0_t0   -     c2     Online 50:0A:09:80:85:84:9D:84
c3_p0_t1   -     c3     Online iqn.1992-08.com.netapp:sn.84190939
```

列出从主机总线适配器或端口配置的目标

◆ 可以使用下列命令根据 HBA 或端口进行过滤：

```
# vxddladm list targets [hba=hba_name|port=port_name]
```

例如，要获得从指定的 HBA 配置的目标：

```
# vxddladm list targets hba=c2
```

```
TgtID      Alias  HBA-ID   State      Address
-------------------------------------------------------------
c2_p0_t0   -      c2       Online     50:0A:09:80:85:84:9D:84
```

列出从主机总线适配器和目标配置的设备

可以获得关于从主机总线适配器配置的所有设备的信息。这些信息包括以下几项：

Device	设备名称。
Target-ID	父目标。
State	设备是联机还是脱机。
DDL status	设备是否由 DDL 索求。如果 DDL 索求设备，则输出同时会显示 ASL 名称。

列出从主机总线适配器配置的设备

◆ 要获得已配置的设备，请使用下列命令：

```
# vxddladm list devices

Device     Target-ID     State     DDL status (ASL)
-------------------------------------------------------
sda        fscsi0_p0_t0 Online    CLAIMED (libvxemc.so)
sdb        fscsi0_p0_t0 Online    SKIPPED (libvxemc.so)
sdc        fscsi0_p0_t0 Offline   ERROR
sdd        fscsi0_p0_t0 Online    EXCLUDED
sde        fscsi0_p0_t0 Offline   MASKED
```

要列出从主机总线适配器和目标配置的设备

◆ 要获得从特定的 HBA 和目标配置的设备，请使用下列命令：

```
# vxddladm list devices target=target_name
```

获取或设置 iSCSI 操作参数

DDL 提供一个接口用于设置并显示某些影响 iSCSI 设备路径性能的参数，但底层的操作系统框架必须支持设置这些值的功能。 如果操作系统不支持，则 `vxddladm set` 命令会返回错误。

表 4-2 iSCSI 设备的参数

参数	默认值	最小值	最大值
DataPDUInOrder	yes	no	yes
DataSequenceInOrder	yes	no	yes
DefaultTime2Retain	20	0	3600
DefaultTime2Wait	2	0	3600
ErrorRecoveryLevel	0	0	2
FirstBurstLength	65535	512	16777215
InitialR2T	yes	no	yes
ImmediateData	yes	no	yes
MaxBurstLength	262144	512	16777215
MaxConnections	1	1	65535

参数	默认值	最小值	最大值
MaxOutStandingR2T	1	1	65535
MaxRecvDataSegmentLength	8182	512	16777215

获取启动器上针对特定 iSCSI 目标设置的 iSCSI 操作参数

◆ 键入以下命令：

```
# vxddladm getiscsi target=tgt-id {all | parameter}
```

可以使用该命令获取所有 iSCSI 操作参数。

```
# vxddladm getiscsi target=c2_p2_t0
```

以下是一个输出示例：

```
PARAMETER                CURRENT  DEFAULT    MIN      MAX
-------------------------------------------------------
DataPDUInOrder               yes      yes     no      yes
DataSequenceInOrder          yes      yes     no      yes
DefaultTime2Retain            20       20      0     3600
DefaultTime2Wait               2        2      0     3600
ErrorRecoveryLevel             0        0      0        2
FirstBurstLength           65535    65535    512 16777215
InitialR2T                   yes      yes     no      yes
ImmediateData                yes      yes     no      yes
MaxBurstLength            262144   262144    512 16777215
MaxConnections                 1        1      1    65535
MaxOutStandingR2T              1        1      1    65535
MaxRecvDataSegmentLength    8192     8182    512 16777215
```

在启动器上设置特定 iSCSI 目标的 iSCSI 操作参数

◆ 键入以下命令：

```
# vxddladm setiscsi target=tgt-idparameter=value
```

列出所有支持的磁盘阵列

使用此过程可以获得与其他形式的 `vxddladm` 命令一起使用的 `vid` 和 `pid` 属性值。

列出所有支持的磁盘阵列

◆ 键入以下命令：

```
# vxddladm listsupport all
```

拒绝支持磁盘阵列库

您可以拒绝支持依赖于特定磁盘阵列库的磁盘阵列。此外，还可以拒绝支持来自特定供应商的磁盘阵列。

拒绝支持磁盘阵列库

◆ 要拒绝支持某一磁盘阵列库，请为下面的命令指定该阵列库。

```
# vxddladm excludearray libname=libvxemc.so
```

您还可以拒绝支持来自特定供应商的磁盘阵列，如下例所示：

```
# vxddladm excludearray vid=ACME pid=X1
```

重新支持被拒绝的磁盘阵列库

如果先前已拒绝支持依赖于特定磁盘阵列库的所有阵列，按照此过程操作可支持这些阵列。 此过程将该库从拒绝列表中删除。

重新支持被拒绝的磁盘阵列库

◆ 如果已经拒绝支持依赖于特定磁盘阵列库的所有阵列，则可以使用 `includearray` 关键字从拒绝列表中删除该项，如下例所示：

```
# vxddladm includearray libname=libvxemc.so
```

此命令将阵列库添加到数据库中，以便该库可以在设备发现中再次使用。如果 `vxconfigd` 正在运行，则可以使用 `vxdisk scandisks` 命令发现阵列并将其详细信息添加到数据库中。

列出被拒绝的磁盘阵列

列出当前被 VxVM 拒绝使用的所有磁盘阵列

◆ 键入以下命令：

```
# vxddladm listexclude
```

列出 DISKS 类别中受支持的磁盘

列出 DISKS (JBOD) 类别中支持的磁盘

◆ 键入以下命令：

```
# vxddladm listjbod
```

显示关于受支持阵列库的详细信息

利用 DMP，可以显示有关阵列支持库 (ASL) 的详细信息。

阵列支持库位于目录 /etc/vx/lib/discovery.d 中。

显示关于受支持阵列库的详细信息

◆ 键入以下命令：

```
# vxddladm listsupport libname=library_name.so
```

此命令显示阵列的供应商 ID (VID)、产品 ID (PID)、阵列类型（例如 A/A 或 A/P）和阵列名称。以下为示例输出。

```
# vxddladm listsupport libname=libvxfujitsu.so
ATTR_NAME            ATTR_VALUE
==================================================
LIBNAME              libvxfujitsu.so
VID                  vendor
PID                  GR710, GR720, GR730
                     GR740, GR820, GR840
ARRAY_TYPE           A/A, A/P
ARRAY_NAME           FJ_GR710, FJ_GR720, FJ_GR730
                     FJ_GR740, FJ_GR820, FJ_GR840
```

将不支持的磁盘阵列添加到 DISKS 类别

如果阵列无可用的阵列支持库 (ASL)，则磁盘阵列应作为 JBOD 设备添加。

除非另外指定，否则假定 JBOD 为主动/主动 (A/A) 阵列。如果合适的 ASL 不能使用，则 A/A-A、A/P 或 A/PF 阵列必须被归类为主动/被动 (A/P) JBOD，以避免路径延迟和出现 I/O 错误。如果 JBOD 符合 ALUA 标准，则添加为 ALUA 阵列。

请参见第 12 页的“DMP 的工作原理”。

警告：此过程可确保在 Veritas Volume Manager 不支持的阵列上正确设置动态多径处理 (DMP)。否则，Veritas Volume Manager 将把磁盘的独立路径视为是单独的设备，这可能会损坏数据。

将不支持的磁盘阵列添加到 DISKS 类别

1 使用以下命令标识阵列中磁盘的供应商 ID 和产品 ID：

```
# /etc/vx/diag.d/vxscsiinq device_name
```

其中：*device_name* 是阵列中某个磁盘的设备名称。记下此命令输出中的供应商 ID (VID) 和产品 ID (PID) 的值。对于 Fujitsu 磁盘，还要记下所显示的序列号的字符数。

下面的示例输出显示供应商 ID 为 `SEAGATE`，产品 ID 为 `ST318404LSUN18G`。

```
Vendor id (VID)      : SEAGATE
Product id (PID)     : ST318404LSUN18G
Revision             : 8507
Serial Number        : 0025T0LA3H
```

2 阻止所有应用程序（例如数据库）访问在阵列上配置的VxVM卷，并卸载在阵列上配置的所有文件系统和存储检查点。

3 如果阵列类型为 A/A-A、A/P 或 A/PF，则将其配置为自动侵入模式。

4 输入下面的命令以添加新的 JBOD 类别：

```
# vxddladm addjbod vid=vendorid [pid=productid] \
[serialnum=opcode/pagecode/offset/length] \
[cabinetnum=opcode/pagecode/offset/length] policy={aa|ap}]
```

其中：*vendorid* 和 *productid* 是您在上一步中找到的 VID 和 PID 值。例如，*vendorid* 可能为 `FUJITSU`、`IBM` 或 `SEAGATE`。对于 Fujitsu 设备，还必须指定序列号的字符数（如 10）作为 `length` 参数的参数。如果阵列类型为 A/A-A、A/P 或 A/PF，则还必须指定 `policy=ap` 属性。

继续前面的示例时，用于定义这种类型的磁盘阵列的命令是：

```
# vxddladm addjbod vid=SEAGATE pid=ST318404LSUN18G
```

5 使用 `vxdctl enable` 命令将阵列置于 VxVM 控制之下。

```
# vxdctl enable
```

请参见第 82 页的“启用新磁盘阵列发现功能”。

6 若要验证现在是否支持该阵列，请输入以下命令：

```
# vxddladm listjbod
```

下面是该阵列示例命令的示例输出：

```
VID      PID          SerialNum            CabinetNum   Policy
                  (Cmd/PageCode/off/len) (Cmd/PageCode/off/len)
==============================================================
SEAGATE ALL PIDs  18/-1/36/12           18/-1/10/11    Disk
SUN     SESS01    18/-1/36/12           18/-1/12/11    Disk
```

7 要验证是否识别该阵列，可使用 `vxdmpadm listenclosure` 命令，如下面的示例阵列的示例输出所示：

```
# vxdmpadm listenclosure
ENCLR_NAME ENCLR_TYPE ENCLR_SNO STATUS    ARRAY_TYPE LUN_COUNT
==============================================================
Disk       Disk       DISKS     CONNECTED Disk       2
```

阵列的磁盘阵列名称和类型都显示为 `Disk`。可以使用 `vxdisk list` 命令来显示阵列中的磁盘：

```
# vxdisk list
DEVICE   TYPE       DISK     GROUP      STATUS
Disk_0   auto:none  -        -          online invalid
Disk_1   auto:none  -        -          online invalid
...
```

8 要验证是否识别 DMP 路径，可使用 `vxdmpadm getdmpnode` 命令，如下面的示例阵列的示例输出所示：

```
# vxdmpadm getdmpnode enclosure=Disk
NAME   STATE   ENCLR-TYPE PATHS ENBL DSBL ENCLR-NAME
=====================================================
Disk_0 ENABLED Disk       2     2    0    Disk
Disk_1 ENABLED Disk       2     2    0    Disk
...
```

此示例中的输出表明阵列中的磁盘有两个路径。

有关更多信息，请输入 `vxddladm help addjbod` 命令。

请参见 `vxddladm`(1M) 手册页。

请参见 `vxdmpadm`(1M) 手册页。

从 DISKS 类别中删除磁盘

请按照本节中的步骤从 DISKS 类别中删除磁盘。

从 DISKS 类别中删除磁盘

◆ 使用带有 rmjbod 关键字的 vxddladm 命令。以下示例说明如何用命令删除供应商 ID 为 SEAGATE 的磁盘：

```
# vxddladm rmjbod vid=SEAGATE
```

外部设备

DDL 可能找不到由第三方驱动程序控制的一些设备，如提供多径处理或 RAM 磁盘功能的那些设备。对于这些设备，使用第三方驱动程序为一些阵列提供的多径处理功能，可能要比使用动态多径处理(DMP)更好一些。可以使用 vxddladm addforeign 命令将这种外部设备配置成简单磁盘，以供 VxVM 使用。这样还可以在处理 I/O 时避开 DMP。下面的示例说明如何在指定的目录中为块设备和字符设备添加对应的条目：

```
# vxddladm addforeign blockdir=/dev/foo/dsk \
 chardir=/dev/foo/rdsk
```

如果驱动程序不支持块或字符设备，则可以使用如下所示的命令将其忽略：

```
# vxddladm addforeign blockdir=/dev/foo/dsk
```

默认情况下，该命令隐藏与自动发现机制在操作系统维护的设备树中找到的匹配设备对应的任何条目。使用 -f 和 -n 选项可以覆盖此行为，详见 vxddladm(1M) 手册页。

为外部设备添加条目后，应使用 vxdisk scandisks 或 vxdctl enable 命令发现用作简单磁盘的设备。然后，这些磁盘就可像自动配置的磁盘那样运行。

VxVM 4.0 中引入了外部设备功能以支持非标准设备（如 RAM 磁盘）、一些固态磁盘和伪设备（如 EMC PowerPath）。

外部设备支持具有以下限制：

- 外部设备始终被视为是只有一条路径的磁盘。与自动发现的磁盘不同，外部设备没有 DMP 节点。
- 集群环境中的共享磁盘组不支持此机制。只支持独立的主机系统。
- 持久性组保留 (PGR) 操作不支持此机制。
- 此机制不受 DMP 控制，因此不能自动启用故障磁盘，并且 DMP 管理命令不适用。

- VxVM 不能使用磁盘阵列信息。这样会降低使用此类设备创建的磁盘组的可用性。
- 外部设备不支持 I/O 防护和集群文件系统功能。

若有合适的 ASL 可用于阵列且已安装，则可以消除这些限制。

请参见第 83 页的“第三方驱动程序共存特性”。

更改磁盘设备的命名机制

可以使用基于磁盘阵列的命名机制为磁盘命名，也可以使用操作系统的命名机制。DMP 命令根据当前命名机制显示设备名称。

默认命名机制是基于磁盘阵列的命名机制 (EBN)。

将 DMP 用于本机卷时，磁盘命名机制必须是 EBN，必须将 use_avid 属性设置为 on，并且必须将 persistence 属性设置为 yes。

更改磁盘命名机制

- 从 vxdiskadm 主菜单中选择“Change the disk naming scheme(**更改磁盘命名机制**)”，以便更改希望 DMP 使用的磁盘命名机制。出现提示后，输入 y 表示要更改命名机制。

  或

  从命令行更改命名机制。使用以下命令选择基于磁盘阵列的命名机制：

  # vxddladm set namingscheme=ebn [persistence={yes|no}] \ [use_avid=yes|no] [lowercase=yes|no]

  使用下列命令选择基于操作系统的命名机制：

  # vxddladm set namingscheme=osn [persistence={yes|no}] \ [lowercase=yes|no]

  可选 persistence 参数可用于选择由 DMP 显示的磁盘设备名称在重新配置磁盘硬件并重新启动系统后是否依然保持不变。默认情况下，基于磁盘阵列的命名机制永久有效， 基于操作系统的命名机制暂时有效。

  要仅更改命名持久性而不更改命名机制，请对当前命名机制运行 vxddladm set namingscheme 命令，并指定 persistence 属性。

  默认情况下，磁盘阵列的名称被转换为小写字母，而与ASL指定的名称大小写无关。因此，基于磁盘阵列的设备名称为小写字母。设置 lowercase=no 选项可以禁止转换为小写字母。

  对于基于磁盘阵列的命名机制，use_avid 选项指定是否在设备名称中将阵列卷 ID 用于索引号。默认情况下，use_avid=yes，表示将设备命名为 *enclosure_avid*。如果将 use_avid 设置为 no，则 DMP 设备命名为 *enclosure_index*。将设备按 LUN 序列号排序之后分配索引号。

  无论使用哪种方法，都会立即执行更改。

  请参见第 97 页的“重新生成持久性设备名称”。

显示磁盘命名机制

DMP 磁盘命名机制可以是基于操作系统的命名机制，也可以是基于磁盘阵列的命名机制。

以下命令可显示当前是否设置了 DMP 磁盘命名机制，它也可显示磁盘命名机制的属性，例如是否启用持久保留。

若要显示当前的磁盘命名机制及其运行模式，请使用以下命令：

```
# vxddladm get namingscheme
NAMING_SCHEME     PERSISTENCE LOWERCASE USE_AVID
==============================================
Enclosure Based  Yes          Yes        Yes
```

请参见第 19 页的“DMP 中的磁盘设备命名”。

重新生成持久性设备名称

持久性设备命名功能使磁盘设备的名称在系统重新启动后保持不变。DDL会根据持久性设备名称数据库来分配设备名称。

如果选择了基于操作系统的命名机制，每个磁盘名称通常会设置为磁盘的某条路径的名称。在重新配置然后重新启动硬件之后，操作系统可能为磁盘路径生成不同的名称。因此，持久性设备名称可能不再对应于实际路径。这不会让磁盘无法使用，但会中断磁盘名称与它的一条路径之间的关联关系。

同样，如果选择基于磁盘阵列的命名机制，则设备名称取决于磁盘阵列的名称和索引编号。如果硬件配置更改阵列中 LUN 的显示顺序，则持久性设备名称可能无法反映当前索引。

重新生成持久性设备名称

◆ 若要重新生成持久性名称储存库，请使用下列命令：

```
# vxddladm [-c] assign names
```

使用 -c 选项可清除所有用户指定的名称并将其替换为自动生成的名称。

如果未指定 -c 选项，则现有的用户指定名称保持不变，但重新生成基于 OS 的名称和基于磁盘阵列的名称。

磁盘名称现在对应于新路径名。

更改 TPD 控制的磁盘阵列的设备命名

默认情况下，TPD 控制的磁盘阵列使用基于 TPD 分配的节点名称的伪设备名称。如果将设备命名更改为本机，则设备的命名格式与 DMP 的其他设备的命名格式相同。设备将使用操作系统名称 (OSN) 或基于磁盘阵列的名称 (EBN)，具体取决于设置的命名机制。

请参见第 96 页的“显示磁盘命名机制”。

更改 TPD 控制的磁盘阵列的设备命名

◆ 对于由第三方驱动程序 (TPD) 控制的磁盘阵列，如果这些驱动程序的共存特性有合适的 ASL 支持，那么默认情况下将基于 TPD 分配的节点名为磁盘阵列分配设备名称。可以使用以下 vxdmpadm 命令在这些名称和操作系统识别的设备名称之间切换：

```
# vxdmpadm setattr enclosure enclosure_name tpdmode=native|pseudo
```

tpdmode 属性的参数所选择的名称基于操作系统使用的名称 (native)，或基于 TPD 分配的节点名称 (pseudo)。

下面的名为 EMC0 的磁盘阵列的示例说明了如何使用上述命令在基于 TPD 和基于操作系统的命名之间切换。在本示例中，设备命名机制设置为 OSN。

```
# vxdisk list

DEVICE         TYPE            DISK       GROUP     STATUS
emcpower10     auto:sliced     disk1      mydg      online
emcpower11     auto:sliced     disk2      mydg      online
emcpower12     auto:sliced     disk3      mydg      online
emcpower13     auto:sliced     disk4      mydg      online
emcpower14     auto:sliced     disk5      mydg      online
emcpower15     auto:sliced     disk6      mydg      online
emcpower16     auto:sliced     disk7      mydg      online
emcpower17     auto:sliced     disk8      mydg      online
emcpower18     auto:sliced     disk9      mydg      online
emcpower19     auto:sliced     disk10     mydg      online

# vxdmpadm setattr enclosure EMC0 tpdmode=native

# vxdisk list

DEVICE         TYPE            DISK       GROUP     STATUS
sdj            auto:sliced     disk1      mydg      online
sdk            auto:sliced     disk2      mydg      online
sdl            auto:sliced     disk3      mydg      online
sdm            auto:sliced     disk4      mydg      online
sdn            auto:sliced     disk5      mydg      online
sdo            auto:sliced     disk6      mydg      online
sdp            auto:sliced     disk7      mydg      online
sdq            auto:sliced     disk8      mydg      online
sdr            auto:sliced     disk9      mydg      online
sds            auto:sliced     disk10     mydg      online
```

如果 tpdmode 设置为 native，则显示设备编号最小的路径。

发现基于磁盘阵列的磁盘名称与基于操作系统的磁盘名称之间的关联关系

如果启用基于磁盘阵列的命名机制，vxprint 命令将使用基于磁盘阵列的磁盘设备名称（磁盘访问名称）而不是基于操作系统的名称来显示卷的结构。

发现基于磁盘阵列的磁盘名称与基于操作系统的磁盘名称之间的关联关系

◆ 要发现与给定基于磁盘阵列的磁盘名称关联的基于操作系统的名称，请使用以下命令之一：

```
# vxdisk list enclosure-based_name
# vxdmpadm getsubpaths dmpnodename=enclosure-based_name
```

例如，要查找与磁盘 ENC0_21 相关联的物理设备，正确的命令为：

```
# vxdisk list ENC0_21
# vxdmpadm getsubpaths dmpnodename=ENC0_21
```

要使用这些命令获得块磁盘设备和字符磁盘设备的完整路径名，可将显示的设备名称追加到 /dev/vx/dmp/ 或 /dev/vx/rdmp/ 后面。

5

动态重新配置设备

本章节包括下列主题：

- 关于联机动态重新配置
- 联机重新配置受 DMP 控制的 LUN
- 联机更换主机总线适配器
- 联机升级阵列控制器固件

关于联机动态重新配置

系统管理员和存储管理员可能需要修改精简置备到服务器的一组LUN。您可以动态更改 LUN 配置，重新配置后无需重新启动主机。

您可以执行以下几种联机动态重新配置：

- 联机重新配置受 DMP 控制的 LUN
  请参见第 101 页的“联机重新配置受 DMP 控制的 LUN”。
- 联机更换主机总线适配器 (HBA)
  请参见第 106 页的“联机更换主机总线适配器”。
- 更新阵列控制器固件，也称为不间断性升级
  请参见第 107 页的“联机升级阵列控制器固件”。

联机重新配置受 DMP 控制的 LUN

系统管理员和存储管理员可能需要修改精简置备到服务器的一组LUN。您可以动态更改 LUN 配置，重新配置后无需重新启动主机。

这些操作如下：

- 从现有目标 ID 中动态删除 LUN

请参见第 102 页的“从现有目标 ID 中动态删除 LUN”。

- 将新的 LUN 动态添加到新的目标 ID
  请参见第 104 页的“将新的 LUN 动态添加到新的目标 ID”。
- 替换现有目标 ID 中的 LUN
  请参见第 104 页的“动态替换现有目标 ID 中的 LUN”。
- 更改 LUN 特性
  请参见第 105 页的“从阵列端更改 LUN 的特性”。

从现有目标 ID 中动态删除 LUN

Veritas Dynamic Multi-Pathing (DMP) 可提供动态重新配置工具，以简化从现有目标 ID 中删除 LUN 的过程。将会取消从主机到每个 LUN 的映射。DMP 将发出操作系统设备扫描并清理操作系统设备树。

警告：在此设备操作完成前，请勿在动态重新配置工具外运行任何设备发现操作。

从现有目标 ID 中动态删除 LUN

1. 解除任何卷管理器对设备的使用。

   对于在 DMP 设备上使用 Linux LVM 的 LUN，请从 LVM 卷组中删除设备

   ```
   # vgreduce vgnamedevicepath
   ```

2. 启动 `vxdiskadm` 实用程序：

   ```
   # vxdiskadm
   ```

3. 从 `vxdiskadm` 菜单中选择“**Dynamic Reconfiguration operations (动态重新配置操作)**”选项。
4. 选择“**Remove LUNs (删除 LUN)**”选项。

5 键入 **list** 或按**回车键**将显示可用于删除的 LUN 列表。如果 LUN 不在磁盘组中且状态为 online、nolabel、online invalid 或 online thinrclm，则可以删除 LUN。

下面显示示例输出：

```
Select disk devices to remove: [<pattern-list>,all,list]: list
LUN(s) available for removeal:
eva4k6k0_0
eva4k6k0_1
eva4k6k0_2
eva4k6k0_3
eva4k6k0_4
emc0_017e
```

6 输入 LUN 的名称、逗号分隔的 LUN 列表或正则表达式，指定要删除的 LUN。

例如，输入 `emc0_017e`。

7 出现提示时，确认所做的 LUN 选择。

DMP 将从 VxVM 使用中删除 LUN。

8 在下面的提示中，从阵列中删除 LUN。

```
Enclosure=emc0 AVID=017E
   Device=emc0_017e Serial=830017E000
-----------------------------------------------------------------
     PATH=sda ctlr=c15 port=7e-a [50:01:43:80:12:08:3c:26]
     PATH=sdc ctlr=c17 port=7e-a [50:01:43:80:12:08:3a:76]
-----------------------------------------------------------------
Please remove LUNs with Above details from array and
press 'y' to continue removal (default:y):
```

9 返回动态重新配置工具，并选择 `y` 继续删除过程。

DMP 会完成从 VxVM 使用中删除设备的过程。将显示类似以下内容的输出：

```
Luns Removed
-------------------------
emc0_017e
```

DMP 将更新操作系统设备树和 VxVM 设备树。

10 选择“**exit (退出)**”可退出动态重新配置工具。

将新的 LUN 动态添加到新的目标 ID

动态重新配置工具简化了向现有目标 ID 添加新 LUN 的过程。一个或多个新 LUN 通过多个 HBA 端口映射到主机。此时会对要识别并添加到 DMP 控制的 LUN 执行操作系统设备扫描。

警告：在此设备操作完成前，请勿在动态重新配置工具外运行任何设备发现操作。

将新的 LUN 动态添加到新的目标 ID

1 启动 `vxdiskadm` 实用程序：

```
# vxdiskadm
```

2 从 `vxdiskadm` 菜单中选择“**Dynamic Reconfiguration operations (动态重新配置操作)**”选项。

3 选择“**Add LUNs (添加 LUN)**”选项。

此工具将发出设备发现命令。

4 显示提示时，添加阵列中的 LUN。

5 选择 **y** 继续向 DMP 添加 LUN。

此操作将触发设备扫描。新发现的设备现在为可见状态。

```
Luns Added
-----------------------------------------------------------------
Enclosure=emc0 AVID=017E
   Device=emc0_017e Serial=830017E000
     PATH=c15t0d6 ctlr=c15 port=7e-a [50:01:43:80:12:08:3c:26]
     PATH=c17t0d6 ctlr=c17 port=7e-a [50:01:43:80:12:08:3a:76]
```

6 选择“**exit (退出)**”可退出动态重新配置工具。

动态替换现有目标 ID 中的 LUN

Veritas Dynamic Multi-Pathing (DMP) 提供了一个动态重新配置工具来简化替换现有目标 ID 中的新 LUN 的过程。将会取消从主机到每个 LUN 的映射。DMP 将发出操作系统设备扫描并清理操作系统设备树。

警告：在此设备操作完成前，请勿在动态重新配置工具外运行任何设备发现操作。

动态替换现有目标 ID 中的 LUN

1 解除任何卷管理器对设备的使用。

对于在 DMP 设备上使用 Linux LVM 的 LUN，请从 LVM 卷组中删除设备

```
# vgreduce vgnamedevicepath
```

2 启动 `vxdiskadm` 实用程序：

```
# vxdiskadm
```

3 从 `vxdiskadm` 菜单中选择“**Dynamic Reconfiguration operations (动态重新配置操作)**”选项。

4 选择“**Replace LUNs (替换 LUN)**”选项。

输出将显示可供替换的 LUN 的列表。LUN 如果不在磁盘组中且状态为 online、nolabel、online invalid 或 online thinrclm，便可供替换。

5 选择一个或多个要替换的 LUN。

6 出现提示时，确认所做的 LUN 选择。

7 从阵列中删除该 LUN。

8 返回动态重新配置工具，并选择 `y` 继续进行删除。

删除成功完成后，动态重新配置工具将提示您添加 LUN。

9 显示提示时，添加阵列中的 LUN。

10 选择 **y** 继续向 DMP 添加 LUN。

此操作将触发设备扫描。新发现的设备现在为可见状态。

DMP 将更新操作系统设备树。

从阵列端更改 LUN 的特性

某些阵列提供了更改 LUN 特性的方法。例如，EMC Symmetrix 阵列中允许使用写保护（只读）的 LUN 和支持读写的 LUN。在更改 LUN 的属性前，必须先使设备脱离 Veritas Volume Manager (VxVM) 的控制。

更改 LUN 的属性

1 解除任何卷管理器对设备的使用。

对于在 DMP 设备上使用 Linux LVM 的 LUN，请从 LVM 卷组中删除设备

```
# vgreduce vgnamedevicepath
```

2 更改设备特性。

3 使用 DMP 执行设备扫描。对集群中的所有节点执行此命令。

```
# vxdisk scandisks
```

4 将设备重新添加回磁盘组。

```
# vxdg -g dgname adddisk daname
```

联机更换主机总线适配器

Veritas Dynamic Multi-Pathing (DMP) 提供了一个动态重新配置工具来简化从现有系统中删除主机总线适配器的过程。

联机更换主机总线适配器

1 启动 `vxdiskadm` 实用程序：

```
# vxdiskadm
```

2 从 `vxdiskadm` 菜单中选择“**Dynamic Reconfiguration operations (动态重新配置操作)**”选项。

3 选择“**Replace HBAs (替换 HBA)**”选项。

输出将显示可用于 DMP 的 HBA 的列表。

4 选择一个或多个要替换的 HBA。

5 出现提示时，确认所做的 HBA 选择。

6 更换该主机总线适配器。

7 返回动态重新配置工具，然后选择 `y` 继续执行替换过程。

DMP 将更新操作系统设备树。

联机升级阵列控制器固件

存储阵列子系统在升级修复、修补程序或功能时需要升级代码。装入文件系统并向存储提供 I/O 时，可以联机执行这些升级。

旧版存储子系统包含两个用于提供冗余的控制器。一次可以对一个控制器进行联机升级。第一个控制器进行联机控制器升级时，DMP 会将所有 I/O 故障转移到第二个控制器。第一个控制器在完全升级代码后，会重新启动、重置并使用新版本的代码进入联机状态。第二个控制器会经历相同的过程，I/O 会故障转移到第一个控制器。

注意：在整个过程中，应用程序 I/O 不受影响。

阵列供应商对此过程有不同的叫法。例如，对于 CLARiiON 阵列，EMC 将该过程称为不间断性升级 (NDU)。

在此联机升级过程中，A/A 类型的阵列不要求特殊处理。对于 A/P、A/PF 和 ALUA 类型的阵列，DMP 会在联机控制器代码升级期间通过特定于供应商的阵列策略模块 (APM) 执行特定于阵列的处理。

控制器在代码升级期间重置并重新启动时，DMP 会通过 SCSI 状态检测到此状态。DMP 会立即将所有 I/O 故障转移到下一个控制器。

如果该阵列不完全支持 NDU，控制器的所有路径可能在短时间内对 I/O 不可用。开始升级前，请为 `dmp_lun_retry_timeout` 可调参数设置一个时间段，该时间段大于您预计控制器对 I/O 不可用的时间。在 `dmp_lun_retry_timeout` 时间段结束前或者 I/O 成功前（以先发生的事件为准），DMP 不会使 I/O 失败。因此，您可以执行固件升级，而无需中断应用程序 I/O。

例如，如果您预计路径对 I/O 不可用的时间为 300 秒，请使用以下命令：

```
# vxdmpadm settune dmp_lun_retry_timeout=300
```

在 300 秒内或者 I/O 成功前，DMP 不会使 I/O 失败。

要验证哪些阵列支持联机控制器升级或 NDU，请参见以下 URL 上的硬件兼容性列表 (HCL)：

http://www.symantec.com/docs/TECH170013

6

事件监视

本章节包括下列主题：

- 关于动态多径处理 (DMP) 事件源后台驻留程序 (vxesd)
- 架构监视和主动错误检测
- iSCSI 和 SAN 光纤通道技术的动态多径处理 (DMP) 发现
- DMP 事件日志记录
- 启动和停止动态多径处理 (DMP) 事件源后台驻留程序

关于动态多径处理 (DMP) 事件源后台驻留程序 (vxesd)

事件源后台驻留程序 (`vxesd`) 是一个 Veritas Dynamic Multi-Pathing (DMP) 组件进程，可接收用于采取适当操作的任何设备相关事件的通知。`vxesd` 的优点包括：

- 监视 SAN 架构事件和主动错误检测（SAN 事件）
  请参见第 109 页的“架构监视和主动错误检测”。
- 记录 DMP 事件以用于故障排除（DMP 事件）
  请参见第 111 页的“DMP 事件日志记录”。
- 自动发现设备（OS 事件）
- 发现 SAN 组件和 HBA 阵列端口连接（光纤通道和 iSCSI）
  请参见第 111 页的“iSCSI 和 SAN 光纤通道技术的动态多径处理 (DMP) 发现”。

请参见第 111 页的“启动和停止动态多径处理 (DMP) 事件源后台驻留程序”。

架构监视和主动错误检测

DMP 在检测路径中的错误时主动发挥作用。DMP 事件源后台驻留程序 `vxesd` 使用全球网络存储工业协会 (SNIA) HBA API 库从 HBA 接收 SAN 架构事件。DMP 根据

SAN 事件中的信息检查可疑设备，即使没有活动的 I/O 也是如此。当 DMP 验证可疑的设备时，新 I/O 会定向到正常路径。

在启动期间，vxesd 查询 HBA（通过 SNIA 库）以获取 SAN 拓扑。vxesd 后台驻留程序确定与对操作系统可见的每个设备路径对应的端口全球名称 (PWWN)。vxesd 后台驻留程序获取拓扑后，vxesd 向 HBA 注册以得到 SAN 事件通知。如果 LUN 从 SAN 断开，则 HBA 会将 SAN 事件通知 vxesd，指定受影响的 PWWN。vxesd 后台驻留程序使用此事件信息并将其与以前的拓扑信息关联，以确定受影响的设备路径集。

vxesd 后台驻留程序将受影响的组发送到 vxconfigd 后台驻留程序 (DDL)，以便设备路径可以标记为可疑。当路径标记为可疑后，DMP 不会将新 I/O 发送到该路径，除非它是设备的最后一个路径。在其下一个定期循环中，DMP 还原任务使用 SCSI 查询探查在后台检查路径的可访问性。如果 SCSI 查询失败，DMP 禁用受影响 LUN 的路径，该路径也将记录在事件日志中。

如果以后重新连接 LUN，HBA 会将 SAN 事件通知 vxesd。当 DMP 还原任务运行其下一个测试循环时，会使用 SCSI 探查检查已禁用的路径，并在成功时重新启用这些路径。

注意： 如果 vxesd 接收到 HBA LINK UP 事件，则会重新启动 DMP 还原任务并立即运行 SCSI 探查，而不等待下一个定期循环。在 DMP 还原任务重新启动时，会开始一个新的定期循环。如果已禁用的路径在第一次 SCSI 探查时无法访问，则在下一个循环（默认情况下为 300 秒）中测试它们。

默认情况下，启用架构监视功能。dmp_monitor_fabric 可调参数的值在重新启动后保持不变。

要禁用架构监视功能，请使用以下命令：

```
# vxdmpadm settune dmp_monitor_fabric=off
```

要启用架构监视功能，请使用以下命令：

```
# vxdmpadm settune dmp_monitor_fabric=on
```

要显示 dmp_monitor_fabric 可调参数的当前值，请使用以下命令：

```
# vxdmpadm gettune dmp_monitor_fabric
```

iSCSI 和 SAN 光纤通道技术的动态多径处理 (DMP) 发现

vxesd 构建对主机可见的 iSCSI 和光纤通道设备的拓扑。vxesd 后台驻留程序使用 SNIA 光纤通道 HBA API 获取 SAN 拓扑。如果 IMA 不可用，则使用 iSCSI 管理 CLI 获取 iSCSI SAN 拓扑。

要显示光纤通道和 iSCSI 设备的分层列表，请使用以下命令：

```
# vxddladm list
```

请参见 vxddladm(1M) 手册页。

DMP 事件日志记录

DMP 将主要事件通知 vxesd，vxesd 将事件记录在日志文件 (/etc/vx/dmpevents.log) 中。这些事件包括：

- 标记已启用的路径或 DMP 节点
- 标记已禁用的路径或 DMP 节点
- 路径限制
- I/O 错误分析
- 主机总线适配器 (HBA) 和存储区网络 (SAN) 事件

日志文件位于 /var/adm/vx/dmpevents.log，但是通过符号方式链接到 /etc/vx/dmpevents.log。当文件达到 10,000 行时，会轮转日志。即，将 dmpevents.log 重命名为 dmpevents.log.X，并创建一个新的 dmpevents.log 文件。

使用可调参数 dmp_log_level，可以更改事件日志文件中的详细级别。有效值是 1 至 4。默认级别为 1。

```
# vxdmpadm settune dmp_log_level=X
```

可以使用以下命令显示 dmp_log_level 的当前值：

```
# vxdmpadm gettune dmp_log_level
```

有关各个日志级别的详细信息，请参见 vxdmpadm (1M) 手册页。

启动和停止动态多径处理 (DMP) 事件源后台驻留程序

默认情况下，DMP 在引导时启动 vxesd。

要停止 `vxesd` 后台驻留程序，请使用 `vxddladm` 实用程序：

vxddladm stop eventsource

要启动 `vxesd` 后台驻留程序，请使用 `vxddladm` 实用程序：

vxddladm start eventsource [logfile=*logfilename*]

要查看 `vxesd` 后台驻留程序的状态，请使用 `vxddladm` 实用程序：

vxddladm status eventsource

7

性能监视和优化

本章节包括下列主题：

- 关于使用模板调整 Veritas Dynamic Multi-Pathing (DMP)
- DMP 调整模板
- DMP 调整模板示例
- 通过配置属性模板调整 DMP 主机
- 管理 DMP 配置文件
- 将 DMP 可调参数和属性重置为默认值
- 模板支持的 DMP 可调参数和属性
- DMP 可调参数

关于使用模板调整 Veritas Dynamic Multi-Pathing (DMP)

Veritas Dynamic Multi-Pathing 具有多个可调参数和属性，可通过配置它们来获得最佳性能。DMP 提供了模板方法，这样通过单一操作便可更新多个可调参数和属性。模板表示完整或部分 DMP 配置，其中显示主机的参数和属性的值。

要查看和使用可调参数，您可以将 DMP 可调参数的配置值转储到一个文件。如有必要，对参数和属性进行编辑。然后，将模板文件加载到主机，就可以通过单一操作更新所有值。

可以将配置文件加载到同一个主机，或另一个类似的主机。模板方法对于下列情况十分有用：

- 使用最佳性能调整值配置多个类似主机。

配置一个主机以获取最佳性能。配置主机后，将可调参数和属性转储到模板文件。然后，您可以将模板文件加载到另一个具有类似要求的主机。Symantec 建议使用同一配置模板的主机具有相同的操作系统和类似的 I/O 要求。

- 定义多个专用模板以处理不同的 I/O 加载要求。
  当主机上的加载更改时，您可以加载另一个模板以获取最佳性能。此策略适用于 I/O 加载中可预见的临时变化。作为系统管理员，在定义系统的 I/O 加载行为后，您可以针对特定加载自定义调整模板。然后，您可以让调整自动进行，因为存在一个可在脚本或 cron 作业中使用的单一加载命令。

您可以随时重置配置，重置会将可调参数和属性的值恢复为 DMP 默认值。

您可以使用 `vxdmpadm config` 命令管理 DMP 配置文件。

请参见 `vxdmpadm`(1m) 手册页。

DMP 调整模板

借助模板机制，您可以通过将配置值转储到文件或标准输出来调整 DMP 参数和属性。

DMP 支持使用模板文件调整下列类型的信息：

- DMP 可调参数。
- 为磁盘阵列、阵列名称或阵列类型定义的 DMP 属性。
- Veritas 命名机制参数。

模板文件分为下列部分：

DMP 可调参数	应用于所有磁盘阵列和阵列。
Namingscheme	应用于所有磁盘阵列和阵列。
Arraytype	用于自定义阵列类型。应用于指定阵列类型的所有磁盘阵列。
Arrayname	适用于特定阵列需要自定义时（即可调参数与应用于阵列类型的可调参数不同时）。 此部分中的属性应用于指定阵列名称的所有磁盘阵列。
Enclosurename	应用于指定 Cab 序列号和阵列名称的磁盘阵列。 适用于特定磁盘阵列需要自定义时（即可调参数与应用于阵列类型和阵列名称的可调参数不同时）。

加载是此部分的基本操作。只有当每个部分中的所有属性都有效时，DMP 才加载每个部分。如果所有部分都已经过处理，则 DMP 会报告错误列表，并警告用户。DMP 不支持部分回滚。DMP 在加载过程中会验证可调参数和属性。不过，Symantec 建议您在尝试加载配置模板文件之前对其进行检查。进行所有必要的更正，直到正确验证配置文件。

加载模板时，将按照下列顺序为属性赋予优先级：

磁盘阵列部分 > 阵列名称部分 > 阵列类型部分

如果同一阵列类型的所有磁盘阵列都需要相同的设置，则从模板中删除相应的阵列名称和磁盘阵列名称部分。仅在阵列类型部分中定义设置。如果某些磁盘阵列或阵列名称需要自定义设置，则保留阵列名称或磁盘阵列的属性部分。如果磁盘阵列或阵列名称的条目使用的设置与为阵列类型定义的设置相同，您可以删除这些条目。

从主机转储配置文件时，该主机可能包含其他主机上不可见的一些阵列。将模板加载到不包括磁盘阵列、阵列类型或阵列名称的目标主机时，DMP 会忽略这些部分。

您可能不想将设置应用于目标主机上的非共享阵列或某些特定于主机的阵列。务必在模板中为这些阵列中的每个阵列定义一个磁盘阵列部分。将模板文件加载到目标主机时，磁盘阵列部分确定设置。否则，DMP 将应用各个阵列名称或阵列类型部分的设置。

DMP 调整模板示例

本节显示一个 DMP 调整模板示例。

```
DMP Tunables
        dmp_cache_open=on
        dmp_daemon_count=10
        dmp_delayq_interval=15
        dmp_restore_state=enabled
        dmp_fast_recovery=on
        dmp_health_time=60
        dmp_log_level=1
        dmp_low_impact_probe=on
        dmp_lun_retry_timeout=0
        dmp_path_age=300
        dmp_pathswitch_blks_shift=9
        dmp_probe_idle_lun=on
        dmp_probe_threshold=5
        dmp_restore_cycles=10
        dmp_restore_interval=300
        dmp_restore_policy=check_disabled
        dmp_retry_count=5
        dmp_scsi_timeout=20
```

```
        dmp_sfg_threshold=1
        dmp_stat_interval=1
        dmp_monitor_ownership=on
        dmp_monitor_fabric=on
        dmp_monitor_osevent=off
        dmp_native_support=off
Namingscheme
        namingscheme=ebn
        persistence=yes
        lowercase=yes
        use_avid=yes
Arraytype
        arraytype=CLR-A/PF
        iopolicy=minimumq
        partitionsize=512
        recoveryoption=nothrottle
        recoveryoption=timebound iotimeout=300
        redundancy=0
Arraytype
        arraytype=ALUA
        iopolicy=adaptive
        partitionsize=512
        use_all_paths=no
        recoveryoption=nothrottle
        recoveryoption=timebound iotimeout=300
        redundancy=0
Arraytype
        arraytype=Disk
        iopolicy=minimumq
        partitionsize=512
        recoveryoption=nothrottle
        recoveryoption=timebound iotimeout=300
        redundancy=0
Arrayname
        arrayname=EMC_CLARiiON
        iopolicy=minimumq
        partitionsize=512
        recoveryoption=nothrottle
        recoveryoption=timebound iotimeout=300
        redundancy=0
Arrayname
        arrayname=EVA4K6K
        iopolicy=adaptive
```

```
        partitionsize=512
        use_all_paths=no
        recoveryoption=nothrottle
        recoveryoption=timebound iotimeout=300
        redundancy=0
Arrayname
        arrayname=Disk
        iopolicy=minimumq
        partitionsize=512
        recoveryoption=nothrottle
        recoveryoption=timebound iotimeout=300
        redundancy=0
Enclosure
        serial=CK200051900278
        arrayname=EMC_CLARiiON
        arraytype=CLR-A/PF
        iopolicy=minimumq
        partitionsize=512
        recoveryoption=nothrottle
        recoveryoption=timebound iotimeout=300
        redundancy=0
        dmp_lun_retry_timeout=0
Enclosure
        serial=50001FE1500A8F00
        arrayname=EVA4K6K
        arraytype=ALUA
        iopolicy=adaptive
        partitionsize=512
        use_all_paths=no
        recoveryoption=nothrottle
        recoveryoption=timebound iotimeout=300
        redundancy=0
        dmp_lun_retry_timeout=0
Enclosure
        serial=50001FE1500BB690
        arrayname=EVA4K6K
        arraytype=ALUA
        iopolicy=adaptive
        partitionsize=512
        use_all_paths=no
        recoveryoption=nothrottle
        recoveryoption=timebound iotimeout=300
        redundancy=0
```

```
        dmp_lun_retry_timeout=0
Enclosure
        serial=DISKS
        arrayname=Disk
        arraytype=Disk
        iopolicy=minimumq
        partitionsize=512
        recoveryoption=nothrottle
        recoveryoption=timebound iotimeout=300
        redundancy=0
        dmp_lun_retry_timeout=0
```

通过配置属性模板调整 DMP 主机

可以使用模板文件将对 DMP 配置进行的一系列更改上传到同一个主机或另一个类似的主机。

Symantec 建议您将 DMP 模板加载到与作为可调参数值来源的主机类似的主机。

通过模板在主机上配置 DMP

1 将当前主机配置的内容转储到一个文件。

```
# vxdmpadm config dump file=filename
```

2 编辑该文件，以对模板中的可调参数进行任何必要的更改。

目标主机可能包括非共享阵列或特定于主机的阵列。为了避免用阵列名或阵列类型的设置更新这些阵列，请在模板中为这些阵列中的每个阵列定义一个磁盘阵列部分。将模板文件加载到目标主机时，磁盘阵列部分确定设置。否则，DMP 将应用各个阵列名称或阵列类型部分的设置。

3 验证 DMP 可调参数的值。

```
# vxdmpadm config check file=filename
```

如果配置检查成功，则 DMP 不显示输出。如果文件包含错误，则 DMP 显示相应错误。进行所有必要的更正，直到配置文件有效。例如，您可能会看到如下所示的错误：

```
VxVM vxdmpadm ERROR V-5-1-0 Template file 'error.file' contains
following errors:

Line No: 22  'dmp_daemon_count' can not be set to 0 or less
Line No: 44  Specified value for 'dmp_health_time' contains
non-digits
Line No: 64  Specified value for 'dmp_path_age' is beyond
the limit of its value
Line No: 76  'dmp_probe_idle_lun' can be set to either on or off
Line No: 281 Unknown arraytype
```

4 将文件加载到目标主机。

```
# vxdmpadm config load file=filename
```

在加载过程中，DMP 验证模板的每个部分。DMP 加载所有有效部分。DMP 不加载包含错误的部分。

管理 DMP 配置文件

您可以显示最近加载到主机的模板文件的名称。该信息包括 DMP 加载模板文件时的日期和时间。

显示主机当前使用的模板文件的名称

◆
```
# vxdmpadm config show
```

```
TEMPLATE_FILE      DATE             TIME
=============================================
/tmp/myconfig      Feb 09, 2011     11:28:59
```

将 DMP 可调参数和属性重置为默认值

DMP 保留 DMP 可调参数和属性的默认值。您可以随时将默认值还原到主机。将丢弃您通过模板文件对主机应用的任何更改。

将 MP 可调参数重置为默认值

◆ 请使用以下命令：

```
# vxdmpadm config reset
```

模板支持的 DMP 可调参数和属性

DMP 支持通过配置模板调整下列可调参数和属性。

DMP 可调参数	请参见第 120 页的“DMP 可调参数”。
为磁盘阵列、阵列名称或阵列类型定义的DMP属性。	■ iopolicy ■ partitionsize ■ use_all_paths ■ recoveryoption 属性（retrycount 或 iotimeout） ■ redundancy ■ dmp_lun_retry_timeout
命名机制属性：	■ naming scheme ■ persistence ■ lowercase ■ use_avid

模板不支持下列可调参数：

- OS 可调参数
- TPD 模式
- 磁盘阵列的故障转移属性 (failovermode)

DMP 可调参数

DMP 提供了各种可用于调整您的环境的参数。

表 7-1 显示了可以调整的 DMP 参数。您可以联机设置可调参数，无需重新启动。

表 7-1 DMP 可调参数

参数	说明
dmp_cache_open	如果该参数设成 on，阵列支持库 (ASL) 执行的首次设备开启将被缓存。该缓存可以最大限度地降低 ASL 后续开启造成的开销，从而提升了设备发现的性能。如果该参数设置为 off，则不执行缓存。 该属性的默认值为 on。
dmp_daemon_count	用于维护路径错误处理、路径还原和其他 DMP 管理任务的内核线程数量。 默认线程数量为 10。
dmp_delayq_interval	阵列故障转移到备用路径后，DMP 在重试 I/O 之前应该等待的时间。在故障转移后，某些磁盘阵列不能立即接受 I/O 请求。 默认值为 15 秒。
dmp_fast_recovery	DMP 是否应该直接从 HBA 接口尝试获取 SCSI 错误信息。如果 HBA 接口支持错误查询功能，将该值设置为 on 可以潜在提高错误恢复速度。如果该参数设置为 off，则不使用 HBA 接口。 默认设置为 on。
dmp_health_time	DMP 立即检查出现故障的路径，防止通过该路径来传送 I/O 请求。dmp_health_time 值表示路径必须处于正常运行状态的时间（以秒为单位）。如果路径的状态在此期间内由 enabled 变回 disabled，DMP 就会将此路径标记为间歇性故障，并且只是在dmp_path_age 秒之后才会重新为 I/O 启用该路径。 默认值为 60 秒。 值为 0 将阻止 DMP 检测间歇性故障路径。

参数	说明
dmp_log_level	为 DMP 控制台消息显示的详细级别。定义了下列级别值： 1 - 显示 5.0 之前的版本中存在的所有 DMP 日志消息。 2 - 显示 1 级消息以及与路径或磁盘添加或删除、SCSI 错误、IO 错误和 DMP 节点迁移相关的消息。 3 - 显示 1 级和 2 级消息以及与路径限制、可疑路径、空闲路径和错误的路径逻辑相关的消息。 4 - 显示 1 级、2 级与 3 级消息以及与设置或更改路径的属性和更改可调参数相关的消息。 默认值为 1。
dmp_low_impact_probe	确定是否优化还原后台驻留程序执行的路径探查。将其设置为 on 可启用优化，将其设置为 off 可禁用优化。仅当还原策略为 check_disabled 或处于 check_periodic 策略的 check_disabled 阶段时，才会优化路径探查。 默认值为 on。
dmp_lun_retry_timeout	指定一个重试时间段来处理 HBA 和 SCSI 驱动程序未处理的瞬态错误。 通常，不需要这样的特殊处理。因此，dmp_lun_retry_timeout 可调参数的默认值为 0。如果指向某个磁盘的所有路径都出现故障，DMP 将无法与应用程序进行 I/O。只对路径检查一次以确定连接是否正常。 在 DMP 需要处理瞬态错误的特殊情况下，可以将 DMP 配置为将应用程序的 I/O 故障推延很短的时间间隔。将 dmp_lun_retry_timeout 可调参数设置为非零值即可指定相应的时间间隔。如果指向 LUN 的所有路径都出现故障且需要检修 I/O，则 DMP 在指定的时间间隔内每五秒探查一次路径。如果路径在该时间间隔内还原，DMP 会检测到这一点并重试 I/O。在达到指定的 dmp_lun_retry_timeout 时间间隔之前或某个路径上的 I/O 成功之前（以先发生的事件为准），DMP 不会让磁盘的 I/O 随着所有出现故障的路径而出现故障。

参数	说明
dmp_monitor_fabric	确定事件源后台驻留程序 (vxesd) 是否使用全球网络存储工业协会 (SNIA) HBA API。此 API 允许 DDL 通过收集有关 SAN 拓扑的信息和监视光纤事件改善故障转移的性能。 如果该参数设置为 on，则 DDL 使用 SNIA HBA API。请注意，HBA 供应商特定的 HBA-API 库应该可用，以便使用此功能。 如果该参数设置为 off，则不使用 SNIA HBA API。 对于 5.0 以前通过安装修补程序支持此 DDL 功能的版本，默认设置为 off。对于 5.0 以及更高版本，默认设置为 on。
dmp_monitor_osevent	确定事件源后台驻留程序 (vxesd) 是否监视操作系统事件（如重新配置操作）。 如果将此参数设置为 on，vxesd 会监视诸如挂接操作系统设备等操作。 如果将此参数设置为 off，则 vxesd 不会监视操作系统操作。DMP 与 EMC PowerPath 共存时，Symantec 建议将此参数设置为 off 以避免出现任何问题。 默认设置为 on，除非已安装 EMC PowerPath。如果是在已安装 PowerPath 的系统上安装 DMP，则 DMP 会将 dmp_monitor_osevent 设置为 off。
dmp_monitor_ownership	确定是否对 ALUA 阵列启用了所有权监控。如果该可调参数设置为 on，则 DMP 轮询设备以查找 LUN 所有权更改。轮询时间间隔由 dmp_restore_interval 可调参数指定。默认值为 on。 如果 dmp_monitor_ownership 可调参数为 off，则 DMP 不会向设备轮询是否存在 LUN 所有权更改。
dmp_native_support	确定 DMP 是否将对本机设备执行多径处理。 将该可调参数设置为 on 可使 DMP 对本机设备执行多径处理。 当 Dynamic Multi-Pathing 作为另一 Veritas 产品的组件安装时，默认值为 off。 当 Veritas Dynamic Multi-Pathing 作为独立产品安装时，默认值为 on。

参数	说明
dmp_path_age	在 DMP 再次针对间歇性故障路径计划 I/O 请求之前，此路径需要按正常运行而被监视的时间。 默认值为 300 秒。 0 值将阻止 DMP 检测间歇性故障的路径。
dmp_pathswitch_blks_shift	在切换到下一个可用路径之前，沿着 DMP 路径发送到磁盘阵列的连续 I/O 块的默认数目。该值以 2 的指数函数的整数指数表示；例如，9 表示 512 块。 默认值为 9。在这种情况下，512 块 (256KB) 连续 I/O 在切换前经 DMP 路径送出。对于具有内部数据缓存的智能磁盘阵列，可以通过增加此可调参数的值来获得更好的吞吐量。例如，对于 HDS 9960 A/A 阵列，如果 I/O 活动模式主要由连续读取或写入组成，则最佳值在 15 和 17 之间。 该参数仅影响 balanced I/O 策略的行为。值为 0 将禁用对此策略的多径处理，如果使用 vxdmpadm 命令为阵列指定其他分区大小，则不会禁用。 请参见第 63 页的“指定 I/O 策略”。
dmp_probe_idle_lun	如果启用了 DMP 统计信息收集功能，将该可调参数设成 on（默认），让 DMP 路径还原线程探查空闲的 LUN； 将此可调参数值设成 off 则关闭此功能。（空闲 LUN 就是上面没有计划 I/O 请求的 VM 磁盘。）只有在启用了 DMP 统计信息收集功能时才解释该可调参数值。关闭统计信息收集功能同时禁用空闲 LUN 侦测。 默认值为 on。
dmp_probe_threshold	如果 dmp_low_impact_probe 设置为 on，则 dmp_probe_threshold 确定在决定更改同一子路径故障转移组中其他路径的状态之前要探查的路径数。 默认值为 5。
dmp_restore_cycles	如果 DMP 还原策略为 check_periodic，则此可调参数的值为在调用 check_all 策略之前经过的周期次数。 默认值为 10。 请参见第 75 页的“配置 DMP 路径还原策略”。

参数	说明
dmp_restore_interval	interval 属性指定路径还原线程检查路径的频率。以秒为单位指定时间。 默认值为 300。 此可调参数的值还可以使用 vxdmpadm start restore 命令进行设置。 请参见第 75 页的“配置 DMP 路径还原策略”。
dmp_restore_policy	DMP 还原策略，可以设置为以下值之一： ■ check_all ■ check_alternate ■ check_disabled ■ check_periodic 默认值为 check_disabled 此可调参数的值还可以使用 vxdmpadm start restore 命令进行设置。 请参见第 75 页的“配置 DMP 路径还原策略”。
dmp_restore_state	如果该参数设置为 enabled，它将使路径还原线程能够启动。 请参见第 75 页的“配置 DMP 路径还原策略”。 如果该参数设置为disabled，它将停止并禁用路径还原线程。 如果该参数设置为 stopped，则在下一个设备发现周期之前，它将停止路径还原线程。 默认值为 enabled。 请参见第 76 页的“停止 DMP 路径还原线程”。
dmp_retry_count	如果某个路径上的 I/O 出现故障，并显示路径繁忙错误，DMP 会将该路径标记为繁忙，并避免在接下来的 15 秒内使用它。如果某个路径连续报告路径繁忙错误达 dmp_retry_count 次，DMP 会将该路径标记为出现故障。dmp_retry_count 的默认值为 5。
dmp_scsi_timeout	确定要为通过 DMP 发送的任何 SCSI 命令设置的超时值。如果 HBA 在此超时值时间段内没有收到发送至设备的 SCSI 命令的响应，该 SCSI 命令将返回并显示失败错误代码。 默认值为 20 秒。

参数	说明
dmp_sfg_threshold	确定在 DMP 开始怀疑故障转移组中的其他路径前在该故障转移组中应失败的最少路径数。值为0将禁用基于子路径故障转移组的故障转移逻辑。 默认值为 1。
dmp_stat_interval	收集 DMP 统计信息之间的时间间隔。 默认值和最小值均为 1 秒。

术语表

Active/Passive disk arrays（**主动/被动磁盘阵列**）	这种多路径磁盘阵列允许指定磁盘的一条路径作为主路径，然后随时通过该路径访问磁盘。在某些磁盘阵列中，使用指定的活动路径以外的路径将导致性能严重下降。
associated plex（**关联的** plex）	与卷关联的 plex。
associated subdisk（**关联的子磁盘**）	与 plex 关联的子磁盘。
associate（**关联**）	在VxVM对象之间建立关系的过程；例如，如果创建一个子磁盘并将其定义为在某个 plex 中有起始点，则称该子磁盘与那个 plex 关联。
atomic operation（**原子操作**）	一种操作，该操作要么完全成功，要么失败并使一切都恢复到操作开始前的状态。如果操作成功，该操作将立即全面生效，用户看不到中间的更改状态。该操作有任何一方面失败都会导致中止，并且不保留局部更改。 在集群中，原子操作要么在所有节点上发生，要么根本不发生。
attached（**已挂接**）	一种状态，处于此状态的 VxVM 对象与另一对象关联并且可用。
block（**块**）	传入或传出磁盘或阵列的最小数据单元。
bootdg	保留的磁盘组名称，它是引导磁盘组的别名。
boot disk group（**引导磁盘组**）	包含可用于引导系统的磁盘的专用磁盘组。
boot disk（**引导磁盘**）	用于引导系统的磁盘。
clean node shutdown（**干净节点关闭**）	在对共享卷的所有访问都已停止时，节点顺利脱离集群的能力。
cluster-shareable disk group（**集群共享磁盘组**）	所含磁盘可由多个主机共享并进行访问的磁盘组，也称为共享磁盘组。
cluster manager（**集群管理器**）	由外部提供并运行于集群内所有节点上的后台驻留程序。每个节点上的集群管理器相互通信并将集群成员集的更改通知 VxVM。
cluster（**集群**）	共享一组磁盘的一组主机，其中的每个主机均称作节点。
column（**列**）	条带 plex 内的一个或多个子磁盘的集合。通过将数据以交替方式均匀地分配在 plex 内的列中来完成条带化。

concatenation（**连续**） 一种将子磁盘按顺序紧密排列的布局方式。

configuration copy（**配置副本**） 配置数据库的单个副本。

configuration database（**配置数据库**） 包含关于现有 VxVM 对象（如磁盘和卷属性）的详细信息的记录集。

data stripe（**数据条带**） 指条带中可以使用的数据部分，大小等于条带减去奇偶校验区域。

DCO (data change object)（**数据更改对象**） 一个 VxVM 对象，用于管理有关 DCO 卷中重新同步映射的信息。必须同时将 DCO 对象和 DCO 卷与某个卷关联才能在此卷上实现持久性重新同步。

DCO volume（DCO **卷**） 用于存放持久性重新同步更改映射和脏区日志的特殊卷。另请参见“脏区日志”。

detached（**已分离**） 一种状态，处于此状态的 VxVM 对象与另一对象关联，但不能使用。

device name（**设备名称**） 用来访问物理磁盘的设备名称或地址，如 `sda` 或 `sda3`，其中 `sda` 表示整个设备，`sda3` 表示 `sda` 上的第三个分区。

在SAN环境中，使用基于磁盘阵列的命名机制会更方便些，该机制通过将磁盘阵列的名称（如 `enc0`）与磁盘阵列内的磁盘编号用下划线连接来组成设备名称（如 `enc0_2`）。也可使用术语“磁盘访问名”指代设备名称。

dirty region logging（**脏区日志**） VxVM 监视对 plex 的修改并将其作为已更改区域的位映射进行记录时所用的一种方法。对于具有新式 DCO 卷的卷，在 DCO 卷中维护脏区日志 (DRL)。否则，DRL 将被分配给称为日志子磁盘的关联子磁盘。

disabled path（**已禁用路径**） 不能用于 I/O 的磁盘路径。由于实际的硬件故障或用户在控制器上使用了 `vxdmpadm disable` 命令，会导致路径被禁用。

disk access name（**磁盘访问名称**） 设备名称的另一种表述。

disk access records（**磁盘访问记录**） 用于指定特定磁盘的访问路径的配置记录。每个磁盘的访问记录均包含名称和类型，还可能包含某些类型特有的信息，这些信息由 VxVM 用于决定如何访问和操作由该磁盘访问记录定义的磁盘。

disk array serial number（**磁盘阵列序列号**） 这是磁盘阵列的序列号。通常印在磁盘阵列机柜上，也可以通过对磁盘阵列上的磁盘发出特定于供应商的 SCSI 命令获得。DMP 子系统使用此数字唯一标识磁盘阵列。

disk array（**磁盘阵列**） 以逻辑方式排列为一个对象的磁盘集合。阵列有助于提供诸如冗余或提高的性能等优点。

disk controller（**磁盘控制器**） 在 VxVM 的多径处理子系统中连接至主机的控制器（主机总线适配器或 HBA）或磁盘阵列，操作系统将其表示为磁盘的父节点。

disk enclosure（**磁盘阵列**） 一种智能磁盘阵列，通常带一个内置有光纤通道环路的底板，该底板允许进行磁盘热交换。

disk group ID（**磁盘组 ID**）　用于标识磁盘组的唯一标识符。

disk group（**磁盘组**）　共享一个公共配置的一个磁盘集合。磁盘组配置是一组记录，其中包含有关现有的 VxVM 对象（如磁盘和卷属性）及其关系的详细信息。每个磁盘组都有管理员分配的名称和内部定义的唯一 ID。系统保留了磁盘组名称 `bootdg`（引导磁盘组的别名）、`defaultdg`（默认磁盘组的别名）和 `nodg`（表示无磁盘组）。

disk ID（**磁盘 ID**）　为每个磁盘提供的通用唯一标识符，可用于标识磁盘，即使磁盘发生移动也可标识。

disk media name（**磁盘介质名**）　磁盘名的另一种表述。

disk media record（**磁盘介质记录**）　用磁盘 ID 标识某个磁盘并为该磁盘提供逻辑（或管理）名称的配置记录。

disk name（**磁盘名**）　为在 VxVM 控制之下的磁盘选择的逻辑或管理名称，如 `disk03`。术语“磁盘介质名”也用来代指磁盘名。

disk（**磁盘**）　已建立了索引并可以快速访问的读取/写入数据块的集合。每个磁盘均有一个通用唯一标识符。

dissociated plex（**已分离的 plex**）　已经从卷分离的 plex。

dissociated subdisk（**已分离的子磁盘**）　已经从 plex 分离的子磁盘。

dissociate（**分离**）　删除存在于两个 VxVM 对象间的任何链接的过程。例如，从 plex 分离子磁盘意味着从 plex 中删除子磁盘并将其添加到空闲空间池中。

distributed lock manager（**分布式锁管理器**）　运行在某个集群中的不同系统上并确保对分布式资源的访问保持一致的锁管理器。

enabled path（**已启用路径**）　可用于 I/O 的磁盘路径。

encapsulation（**封装**）　将指定磁盘上的现有分区转换为卷的过程。如果存在包含文件系统的任何分区，则将修改 `/etc/fstab` 项，以便改为将文件系统装入卷中。

enclosure-based naming（**基于磁盘阵列的命名**）　请参见“设备名称”。

enclosure（**磁盘阵列**）　请参见“磁盘阵列”。

fabric mode disk（**光纤模式磁盘**）　可通过光纤通道交换机在存储区域网络 (SAN) 上访问的磁盘设备。

FastResync（**重新同步**）	快速重新同步功能，用于对失效镜像执行快速、有效的重新同步和提高快照机制的效率。
Fibre Channel（**光纤通道**）	通常用于设置存储区域网络 (SAN) 的光纤技术的总称。
file system（**文件系统**）	组织成某种结构的文件集合。 UNIX 文件系统是由目录和文件组成的分层结构。
free space（**空闲空间**）	受 VxVM 控制的磁盘区域，该区域未分配给任何子磁盘或保留供任何其他 VxVM 对象使用。
free subdisk（**空闲子磁盘**）	未与任何 plex 关联的子磁盘，该子磁盘具有空的 `putil[0]` 字段。
hostid（**主机** ID）	向 VxVM 标识主机的字符串。主机的主机 ID 存储在它的 `volboot` 文件中，用于定义磁盘和磁盘组的所有权。
hot-relocation（**热重定位**）	一种在磁盘发生故障时自动还原冗余及对镜像卷和 RAID-5 卷的访问的技术。通过将受影响的子磁盘重新定位到同一磁盘组中指定为备用和/或空闲空间的磁盘，可以做到这一点。
hot-swap（**热交换**）	指不需要先关闭系统电源即可在系统中拔下或插上的设备。
initiating node（**启动节点**）	一类节点，系统管理员在这种节点上运行请求对VxVM对象进行更改的实用程序。该节点启动重新配置卷的操作。
JBOD (just a bunch of disks)（**简单磁盘捆绑**）	可能支持也可能不支持磁盘热交换的非智能磁盘阵列的通称。
log plex（**日志** plex）	用于存储 RAID-5 日志的 plex。术语“日志 plex”还可用于代指脏区日志 plex。
log subdisk（**日志子磁盘**）	用于存储脏区日志的子磁盘。
mastering node（**主控节点**）	磁盘挂接到的节点，又称磁盘所有者。
master node（**主节点**）	被软件指派在集群中协调某些 VxVM 操作的节点。任何节点都能够用作主节点。
mirroring（**镜像**）	将卷内容镜像到多个 plex 上的布局技术。每个 plex 都复制存储在卷上的数据，但 plex 本身可能有不同的布局。
mirror（**镜像**）	卷及卷数据（以有序子磁盘集合的形式存在）的重复副本。每个镜像由与该镜像关联的卷的一个 plex 组成。
multipathing（**多径处理**）	如果与系统相连的磁盘有多条物理访问路径，则称该磁盘被进行多径处理。同时称驻留在主机上并使用户无法察觉此事实的任何软件（如 DMP 驱动程序）提供多径处理功能。
node abort（**节点中止**）	节点在紧急情况下不停止正在进行的操作便离开集群的情形。
node join（**节点加入**）	节点加入集群并获得对共享磁盘的访问权限的过程。

node（**节点**）
集群中的主机之一。

Non-Persistent FastResync（**非持久性重新同步**）
重新同步的一种形式，这种重新同步不能在系统重引导之间保留映射，原因是它将更改映射存储到内存中。

object（**对象**）
定义到VxVM并由其在内部识别的实体。VxVM对象包括：卷、plex、子磁盘、磁盘和磁盘组。实际上，有两种类型的磁盘对象 - 一种属于磁盘物理方面的，另一种属于磁盘逻辑方面的。

parity stripe unit（**奇偶校验条带单元**）
包含奇偶校验信息的 RAID-5 卷存储区域。包含在奇偶校验条带单元中的数据可用于帮助重建 RAID-5 卷中因 I/O 故障或磁盘故障而丢失的区域。

parity（**奇偶校验**）
一个计算出的值，可用于在发生故障后重建数据。将数据写入 RAID-5 卷时，还可以通过对数据执行异或 (XOR) 过程来计算奇偶校验。计算出的奇偶校验值随后被写入卷。如果 RAID-5 卷的一部分失败，可以通过其余的数据和奇偶校验重新创建失败卷上的那部分数据。

partition（**分区**）
物理磁盘设备的标准划分，由操作系统和磁盘驱动器直接支持。

pathgroup（**路径组**）
对于 `vxdmp` 不进行多径处理的磁盘，VxVM将每条路径都视为是一个磁盘。在这种情况下，可以将磁盘的所有路径分组。这样，路径组中只有一条路径对于VxVM是可见的。

path（**路径**）
磁盘连接到主机后，磁盘的路径由主机上的 HBA（主机总线适配器）、SCSI 或光纤电缆连接器以及磁盘或磁盘阵列上的控制器组成。这几个部分共同构成了磁盘路径。任何部分出现故障都会导致 DMP 尝试将磁盘的所有 I/O 切换到剩余（备用）的路径上。

Persistent FastResync（**持久性重新同步**）
重新同步的一种形式，这种重新同步通过将更改映射存储在磁盘上的DCO卷中，可以在系统重引导之间保留其映射。

persistent state logging（**持久性状态日志记录**）
一种日志类型，可确保在恢复时只使用活动镜像并防止选择失败镜像。也称为内核日志。

physical disk（**物理磁盘**）
基础存储设备，可能受也可能不受 VxVM 的控制。

plex
Plex是子磁盘的逻辑分组，用于创建不受物理磁盘大小或其他限制约束的磁盘空间区域。可通过为单个卷创建多个数据 plex 来设置镜像。镜像卷中的每个数据 plex 都包含卷数据的相同副本。还可以创建 Plex 来表示连续、条带式和 RAID-5 卷布局以及存储卷日志。

primary path（**主路径**）
在主动/被动磁盘阵列中，可以将磁盘绑定到磁盘阵列上的某个特定控制器或使其为某一控制器所有。这样就可以使用通过此特定控制器的路径访问该磁盘。

private disk group（**专用磁盘组**）
其磁盘仅由集群中的某个特定主机访问的磁盘组。

private region（**专用区域**）
物理磁盘中用于存储专用结构化VxVM信息的区域。专用区域包含磁盘头文件、目录和配置数据库。目录表映磁盘的内容。磁盘头文件包含一个磁盘ID。专用区域中的所有数据都被复制以提高可靠性。

public region（**公共区域**）
物理磁盘中由 VxVM 管理的区域，包含可用空间并用于分配子磁盘。

RAID (redundant array of independent disks)（**独立磁盘冗余阵列**）
一种磁盘阵列，设置为使用组合存储容量的一部分来存储该阵列中所存储数据的复制信息。如果发生磁盘故障，这种方法有助于重新生成数据。

read-writeback mode（**读取-写回模式**）
一种恢复模式，在此模式下，每个读操作都恢复所读区域的plex一致性。恢复plex一致性的方法是从一个plex的块中读取数据，然后将这些数据写入所有其他可写的plex。

rootability（**根目录可置性**）
可将根文件系统和交换设备置于VxVM控制之下。可以镜像产生的卷，以提供冗余性并使用户能够在磁盘发生故障后进行恢复。

root configuration（**根配置**）
根磁盘组的配置数据库。根配置的特别之处在于，它总是包含其他磁盘组的记录，而这些记录仅用于备份目的。它还包含定义系统上所有磁盘设备的磁盘记录。

root disk（**根磁盘**）
包含根文件系统的磁盘。该磁盘可能受 VxVM 控制。

root file system（root **文件系统**）
作为 UNIX 内核启动序列的一部分安装的初始文件系统。

root partition（**根分区**）
根文件系统所驻留的磁盘区域。

root volume（**根卷**）
包含根文件系统的 VxVM 卷（如果系统配置指定了这样的卷）。

SAN（**存储区域网络**）
一种组网范例，在这种网络中，可以轻松地重新配置任何计算机、磁盘存储和互连硬件（如交换机、集线器和网桥）子集之间的连接。

secondary path（**次级路径**）
在主动/被动磁盘阵列中，主路径以外的磁盘路径被称为次级路径。在磁盘发生故障前，应只通过主路径访问磁盘；磁盘发生故障后，磁盘的所有权被移交给其中一个次级路径。

sector（**扇区**）
一种度量单位，可能与系统有关。扇区大小是按设备（硬盘驱动器、CD-ROM等）设置的。虽然为了实现互操作而通常将同一系统内的所有设备设置为相同的扇区大小，但并非总是如此。

一个扇区通常有 512 字节。

shared disk group（**共享磁盘组**）
所含磁盘可由多个主机共享并进行访问的磁盘组，也称为集群共享磁盘组。

shared VM disk（**共享 VM 磁盘**）
属于集群中的共享磁盘组的 VM 磁盘。

shared volume（**共享卷**）
属于共享磁盘组并且同时在集群的多个节点上打开的卷。

slave node（**从属节点**）	没有被指定为集群的主节点的节点。
slice（**片**）	逻辑磁盘设备的标准划分。 术语分区有时等同于磁盘片。
snapshot（**快照**）	卷的即时点副本（卷快照）或文件系统的即时点副本（文件系统快照）。
spanning（**跨接**）	一种布局技术，用于在多个物理磁盘上配置由于太大而无法放入单个磁盘的卷（及其文件系统或数据库）。
sparse plex（**稀疏** plex）	一种 plex，它或者长度小于卷或者有孔洞（没有支持子磁盘的 plex 区域）。
stripe size（**条带大小**）	条带单元大小的总和，这些条带单元构成跨越所有条带列的单个条带。
stripe unit size（**条带单元大小**）	每个条带单元的大小。 默认条带单元大小为 64KB。 条带单元大小有时也称为条带宽度。
stripe unit（**条带单元**）	交替分布在每个条带 plex 的子磁盘上（在列内）的大小相等的区域。 在阵列内，这是在从阵列中的下一磁盘进行重定位之前存在于每个磁盘上的一组逻辑上连续的块。 条带单元也可以称为条带元素。
stripe（**条带**）	在一系列列中占据同一位置的条带单元集。
striping（**条带化**）	使用条带将数据分布在多个物理磁盘上的布局技术。数据交替分布到每个 plex 的子磁盘内的条带上。
subdisk（**子磁盘**）	形成逻辑磁盘区段的一组连续的磁盘块。 子磁盘可与 plex 关联以形成卷。
swap area（**交换区**）	一种磁盘区域，用于容纳由系统分页进程换出的内存页副本。
swap volume（**交换卷**）	配置以用作交换区域的 VxVM 卷。
transaction（**事务**）	一组配置更改，作为一个整体而非个体成功或失败。事务在内部使用以保持配置的一致性。
VMdisk（VM **磁盘**）	受 VxVM 控制并分配给磁盘组的磁盘。VM 磁盘有时又称 VxVM 磁盘。
volboot file（volboot **文件**）	一个小文件，用于定位引导磁盘组配置的副本。此文件可能列出包含标准位置中的配置副本的磁盘，而这些磁盘还可能包含直接指向配置副本位置的指针。`volboot` 文件的存储位置与系统有关。
volume configuration device（**卷配置设备**）	卷配置设备 (`/dev/vx/config`) 是一种接口，通过它可以执行对卷设备驱动程序进行的所有配置更改。
volume device driver（**卷设备驱动程序**）	在应用程序层面和物理设备驱动程序层面之间形成虚拟磁盘驱动器的驱动程序。卷设备驱动程序通过虚拟磁盘设备节点来访问，该节点的字符设备节点出现在 `/dev/vx/rdsk` 中，其块设备节点出现在 `/dev/vx/dsk` 中。
volume（**卷**）	即虚拟磁盘，表示由文件系统或数据库等应用程序使用的可寻址磁盘块范围。卷是由 1-32 个 plex 组成的集合。
vxconfigd	VxVM 配置后台驻留程序，负责更改 VxVM 配置。必须运行该后台驻留程序才能执行 VxVM 操作。

主动/主动磁盘阵列 使用这种经过多径处理磁盘阵列可以同时通过磁盘阵列中某个磁盘的所有路径来访问该磁盘，并且不会降低性能。

索引

符号

/dev/vx/dmp 目录 13
/dev/vx/rdmp 目录 13
/etc/vx/dmppolicy.info 文件 63

A

A/A 磁盘阵列 12
A/A-A 磁盘阵列 12
A/P 磁盘阵列 12
A/P-C 磁盘阵列 12–13
A/P-F 磁盘阵列 13
A/P-G 磁盘阵列 13
active 路径属性 60
adaptive 负载平衡 64
adaptiveminq 策略 64
APM
    配置 77
ASL
    阵列支持库 81–82

B

部分设备发现 80

C

check_all 策略 75
check_alternate 策略 75
check_disabled 策略 75
check_periodic 策略 75
持久性
    设备命名选项 96
持久性设备名称数据库 97
持久性设备命名机制 97
磁盘 82
    OTHER_DISKS 类别 82
    磁盘阵列 19
    从 DISKS 类别中删除 94
    调用发现 83
    辅助路径 41
    更改命名机制 95
    禁用的路径 41
    类别 82
    列出 JBOD 中受支持的磁盘 91
    命名机制 19
    配置持久性名称 97
    配置最新添加的 79
    启用的路径 41
    扫描 79
    设备发现层 84
    添加到 DISKS 类别 92
    显示命名机制 96
    由 DMP 发现 79
    由 VxVM 发现 81
    元设备 18
    阵列支持库 82
    主路径 41
磁盘名称
    配置持久性 97
磁盘阵列 19
    A/A 12
    A/A-A 12
    A/P 12
    A/P-F 13
    A/P-G 13
    JBOD 设备 81
    从 DISKS 类别中删除磁盘 94
    发现磁盘访问名称 99
    非对称主动/主动 12
    经过多径处理 18
    拒绝支持 90
    列出 DISKS 类别中受支持的磁盘 91
    列出被拒绝 90
    列出支持 90
    路径冗余 61–62
    设置路径属性 60, 62
    显示相关信息 50
    向 DISKS 类别添加磁盘 92
    支持 DMP 89
    重新支持 90
    主动/被动 12
    主动/主动 12
次级路径 12
存储处理器 12

错误
处理瞬态错误 122
错误的 I/O
显示统计数据 57

D

DDL 18
设备发现层 84
DISKS 类别 82
列出受支持磁盘 91
删除磁盘 94
添加磁盘 92
DMP
check_all 还原策略 75
check_alternate 还原策略 75
check_disabled 还原策略 75
check_periodic 还原策略 75
path-switch 可调参数 124
vxdmpadm 43
安排辅助路径上的 I/O 66
磁盘发现 79
动态多径处理 12
负载平衡 16
还原策略 75
基于磁盘阵列的命名 14
集群环境中 17
节点 13
禁用控制器 68
禁用路径 68
禁用阵列端口 68
路径故障转移机制 15
路径老化 121
配置 DMP 路径还原策略 75
配置 I/O 控制 72
配置磁盘设备 79
配置对 I/O 错误的响应 71, 74
启用控制器 69
启用路径 69
启用阵列端口 69
日志记录级别 122
设置 DMP 还原轮询间隔 75
使用模板调整 113
收集 I/O 统计信息 55
停止 DMP 还原后台驻留程序 76
显示 DMP 节点 45–46
显示 DMP 路径还原线程的状态 76
显示 DMP 数据库信息 40
显示 recoveryoption 值 74
显示 TPD 信息 51
显示磁盘阵列的 DMP 节点 45–46
显示磁盘阵列的相关信息 50
显示磁盘阵列端口的路径 48
显示节点的 LUN 组 47
显示控制器的路径 48
显示路径的 DMP 节点 44
显示由 DMP 节点控制的路径 47
显示有关控制器的信息 49
显示有关路径的信息 40
显示有关阵列端口的信息 51
元节点 13
重命名磁盘阵列 70
DMP 还原轮询间隔 75
DMP 节点
设置名称 42
显示综合信息 45
DMP 支持
JBOD 设备 81
DMP 中的路径故障转移 15
dmp_cache_open 可调参数 121
dmp_daemon_count 可调参数 121
dmp_delayq_interval 可调参数 121
dmp_fast_recovery 可调参数 121
dmp_health_time 可调参数 121
dmp_log_level 可调参数 122
dmp_low_impact_probe 122
dmp_lun_retry_timeout 可调参数 122
dmp_monitor_fabric 可调参数 123
dmp_monitor_osevent 可调参数 123
dmp_monitor_ownership 可调参数 123
dmp_native_support 可调参数 123
dmp_path_age 可调参数 124
dmp_pathswitch_blks_shift 可调参数 124
dmp_probe_idle_lun 可调参数 124
dmp_probe_threshold 可调参数 124
dmp_restore_cycles 可调参数 124
dmp_restore_interval 可调参数 125
dmp_restore_state 可调参数 125
dmp_scsi_timeout 可调参数 125
dmp_sfg_threshold 可调参数 126
dmp_stat_interval 可调参数 126
单个活动路径策略 65
第三方驱动程序 (TPD) 83
调整 DMP
使用模板 113
端口
列出 86
多径处理
显示相关信息 40

E

EMC PowerPath
    与 DMP 共存 83
EMC Symmetrix
    自动发现 83
errord 后台驻留程序 15

F

FAILFAST 标志 15
访问端口 12
非对称主动/主动磁盘阵列 12
非自动侵入模式 13
分区大小
    显示值 63
    指定 64
辅助路径显示 41
负载平衡 12
    显示策略 63
    指定策略 63

G

故障转移模式 12
光纤设备 80

H

HBA
    列出端口 86
    列出目标 86
    列出支持的 86
HBA 信息
    显示 50
hdx 18
还原策略
    check_all 75
    check_alternate 75
    check_disabled 75
    check_periodic 75
恢复选项值
    配置 74
活动路径
    设备 61–62
基于 hdx 的命名机制 19
基于 sdx 的命名机制 19
基于磁盘阵列的命名 21, 95
    DMP 14
基于磁盘阵列的命名机制 19

I

I/O
    安排辅助路径 66
    控制 15
    收集 DMP 的统计信息 55
I/O 策略
    示例 66
    显示 63
    指定 63
I/O 控制 72
iSCSI 参数
    使用 DDL 管理 88
    使用 vxddladm 设置 88

J

JBOD
    DMP 支持 81
    从 DISKS 类别中删除磁盘 94
    列出受支持磁盘 91
    向 DISKS 类别添加磁盘 92
基于磁盘阵列的命名机制
    由 vxprint 显示 99
集群
    DMP 的使用 17
节点
    DMP 13
禁用的路径 41

K

可调参数
    dmp_cache_open 121
    dmp_daemon_count 121
    dmp_delayq_interval 121
    dmp_fast_recovery 121
    dmp_health_time 121
    dmp_log_level 122
    dmp_low_impact_probe 122
    dmp_lun_retry_timeout 122
    dmp_monitor_fabric 123
    dmp_monitor_osevent 123
    dmp_monitor_ownership 123
    dmp_native_support 123
    dmp_path_age 124
    dmp_pathswitch_blks_shift 124
    dmp_probe_idle_lun 124
    dmp_probe_threshold 124
    dmp_restore_cycles 124
    dmp_restore_interval 125

dmp_restore_state 125
dmp_scsi_timeout 125
dmp_sfg_threshold 126
dmp_stat_interval 126
空闲 LUN 124
控制 15
控制 I/O 选项
配置 74
控制器
对 DMP 禁用 68
对 DMP 启用 69
显示相关信息 49
在 DMP 中禁用 39
控制器 ID
显示 50

L

LUN 12
空闲 124
LUN 组
显示详细信息 47
LUN 组故障转移 13
类别
磁盘 82
列出
DMP 节点 45
支持的磁盘阵列 89
路径
对 DMP 禁用 68
对 DMP 启用 69
设置属性 60, 62
路径老化 121
逻辑单元 12

M

mrl
关键字 62
名称
设备 18
命名
DMP 节点 42
命名机制
磁盘 19
为磁盘更改 95
为磁盘显示 96
目标
列出 86

N

naming scheme
TPD 磁盘阵列的更改 97
nomanual 路径属性 60
nopreferred 路径属性 61

O

OTHER_DISKS 类别 82

P

PowerPath
与 DMP 共存 83
preferred priority 路径属性 61
primary 路径属性 61
排队 I/O
显示统计数据 57
配置 DMP
使用模板 113
乒乓效应 17
平衡路径策略 64

Q

启用的路径
显示 41
全球名称标识符 19

R

restored 后台驻留程序 15
round-robin
负载平衡 65
冗余级别
为设备指定 62
显示设备的 61
冗余循环访问 20

S

scandisks
vxdisk 命令 79
secondary 路径属性 61
standby 路径属性 61
设备
JBOD 81
光纤 80
列出所有 85
路径名称 18
路径冗余 61–62
添加外部 94

元设备 18
设备发现
部分 80
引入 18
设备发现层 84
设备发现层 (DDL) 18, 84
设备名称 18
配置持久性 97
用户指定 42
设置
路径冗余级别 62
属性
active 60
nomanual 60
nopreferred 61
preferred priority 61
primary 61
secondary 61
standby 61
路径的设置 62
为路径设置 60

T

TPD
显示路径信息 51
支持共存 83
tpdmode 属性 97
统计信息收集 15

U

use_all_paths 属性 66
use_avid
vxddladm 选项 96

V

vxdctl enable
调用设备发现 82
配置新磁盘 79
vxddladm
从 DISKS 类别中删除磁盘 84, 94
更改命名机制 96
列出 DISKS 类别中受支持的磁盘 91
列出被拒绝的磁盘阵列 90
列出排除的磁盘阵列 92
列出配置的目标 87
列出配置的设备 88
列出所有设备 85
列出支持的 HBA 86
列出支持的磁盘阵列 90
列出主机总线适配器上的端口 86
设置 iSCSI 参数 88
添加外部设备 94
显示磁盘命名机制 96
向 DISKS 类别添加磁盘 92
用于拒绝支持磁盘阵列 90
用于重新支持磁盘阵列 90
vxdisk
发现磁盘访问名称 99
扫描磁盘设备 79
显示多径处理信息 41
vxdisk scandisks
扫描设备 80
重新扫描设备 80
vxdiskadm
更改磁盘命名机制 95
vxdmpadm
发现磁盘访问名称 99
更改 TPD 命名机制 97
列出有关阵列端口的信息 51
配置 APM 77
配置 I/O 控制 72
配置对 I/O 错误的响应 71, 74
删除 APM 77
设置 I/O 策略 65
设置还原轮询间隔 75
设置路径属性 61
收集 I/O 统计信息 55
停止 DMP 还原后台驻留程序 76
显示 APM 信息 77
显示 DMP 还原线程的状态 76
显示 DMP 数据库信息 40
显示 I/O 策略 63
显示 I/O 错误恢复设置 74
显示 TPD 信息 51
显示磁盘阵列的 DMP 节点 45–46
显示磁盘阵列的相关信息 50
显示分区大小 63
显示路径的 DMP 节点 45, 47
显示由 DMP 节点控制的路径 47
显示有关控制器的信息 49
在 DMP 中禁用 I/O 69
在 DMP 中禁用控制器 39
在 DMP 中启用 I/O 70
指定 DMP 路径还原策略 75
重命名磁盘阵列 70
vxdmpadm list
显示 DMP 节点 45

vxprint
基于磁盘阵列的磁盘名称 99
用于基于磁盘阵列的磁盘名称 99
VxVM
磁盘发现 81

W

WWN 标识符 19
外部设备
添加 94

X

显示
DMP 节点 45
HBA 信息 50
冗余级别 61
支持的磁盘阵列 89
显示统计数据
错误的 I/O 57
排队 I/O 57
显式故障转移模式 13
性能
DMP 中的负载平衡 16

Y

隐式故障转移模式 12
用户指定的设备名称 42
优先负载平衡 65
元节点
DMP 13
元设备 18

Z

阵列
DMP 支持 81
阵列策略模块 (APM)
配置 77
阵列端口
对 DMP 禁用 68
对 DMP 启用 69
显示相关信息 51
阵列卷 ID
设备命名 96
阵列支持库 (ASL) 82
指定
冗余级别 62
主动/被动磁盘阵列 12
主动/主动磁盘阵列 12
主路径 12, 41
自定义命名
DMP 节点 42
自动侵入模式 12
最小队列负载平衡策略 65
最小冗余级别
为设备指定 62
显示设备的 61