

EMC²

EMC 白皮书

《EMC 统一存储系统性能和可用性基础知识》

易安信电脑系统（中国）有限公司
中国北京朝阳区霄云路 38 号现代汽车大厦 15 层
邮编：100027
电话：(8610) 8438 6000
传真：(8610) 8453 8174
售前服务热线：400 650 6006
网站：<http://china.EMC.com>

版权所有 © 2011 EMC Corporation。保留所有权利。

出版日期：2011 年 10 月

EMC 确信本出版物在发布之日内容准确无误。本文档中的信息可随时更改而不另行通知。

本出版物的内容按“原样”提供。EMC CORPORATION 对本出版物的内容不提供任何形式的陈述或担保，明确拒绝对有特定目的的适销性或适用性进行默示担保。

使用、复制或分发本出版物所描述的任何 EMC 软件都要有相应的软件许可证。

有关 EMC 产品名称的最新清单，请参见 <http://china.emc.com> 上的 EMC Corporation 商标。

本文提到的所有其他商标均为其各自所有者的资产。

《EMC 统一存储系统性能和可用性基础知识》

P/N h1094.5

如何使用本手册	xiii
第 1 章	
存储环境	1
存储系统	3
一般体系结构	4
主机	5
存储网络	5
直连存储 (DAS)	6
存储区域网络 (SAN) 和网络连接存储 (NAS)	6
聚合网络	6
第 2 章	
常见协议	7
协议堆栈	8
物理和数据链路协议	9
光纤通道协议	9
以太网协议	9
SAS 协议	10
网络协议	10
光纤通道协议	10
TCP/IP 协议	10
iSCSI 协议	11
文件协议 (CIFS/NFS)	12
数据块协议 (SCSI)	13
第 3 章	
性能和可用性衡量指标	15
带宽	16
吞吐量	17
存储设备上的带宽和吞吐量	17
响应时间	18
用户响应时间	18
I/O 响应时间	19
可用性	19
RPO 衡量指标	19

	RTO衡量指标	19
	RPO/RTO示例	19
第4章	工作负载配置文件	21
	描述工作负载	22
	I/O类型	22
	应用程序缓冲和并发	25
	I/O特征和工作负载	26
	了解工作负载	27
第5章	软件	29
	操作环境	30
	Unisphere	30
	Unisphere Manager	30
	Unisphere CLI	30
	Unisphere Analyzer	30
	复制分层应用程序	30
	MirrorView	31
	RecoverPoint	31
	SAN Copy	31
	SnapView	31
	SnapSure	31
第6章	VNX物理体系结构	33
	存储模块度量	34
	存储处理器存储模块	34
	基于存储处理器存储模块(SPE)的系统	34
	基于磁盘处理器存储模块(DPE)的系统	35
	备用电源	35
	SPS电池电源	35
	SPS就绪测试	36
	VNX交流电源故障行为	36
	磁盘阵列存储模块	37
	DAE O/S	37
	DAE和存储	37
	X-Blade存储模块	37
	控制台存储模块	37
	物理配置示例	38
	硬件文档	38
第7章	VNX逻辑体系结构	39
	前端端口	40
	VNX前端端口	40
	端口位置	41
	自动协商	41
	光纤通道端口	41
	iSCSI端口	42

FCoE 端口数	42
前端端口性能	42
存储处理器 (SP)	44
存储处理器 CPU.....	44
内存	44
后端端口	46
FAST Cache.....	48
存储	49
存储模块	49
双端口驱动器	50
驱动器和后端端口带宽	50
文件逻辑设备	50
X-Blade	50
控制台	50
第 8 章	
存储对象	51
物理存储对象	52
基本的机械硬盘驱动器术语	52
基本闪存驱动器术语	52
硬盘驱动器分类	54
磁盘省电（驱动器降速）	58
逻辑存储对象	59
RAID 组	59
专用 RAID 组.....	66
逻辑单元 (LUN)	67
LUN 类型.....	71
虚拟池	74
传统 LUN.....	77
MetaLUN	78
存储组	80
物理资源配置示例	80
第 9 章	
存储对象性能	83
驱动器性能	84
硬盘驱动器规格	84
硬盘驱动器队列	85
计算硬盘驱动器性能	85
硬盘驱动器速度和性能	87
硬盘驱动器容量利用率和性能	88
驱动器性能比较	88
RAID 组性能	89
分条性能	89
缓存性能	92
未缓存的性能	92

RAID 级别性能差异	92
RAID 级别性能：奇偶校验与镜像	92
RAID 组性能计算	95
吞吐量估计	95
吞吐量计算	95
LUN 性能	95
短寻道	96
第 10 章 存储系统性能	97
百分比利用率	98
利用率保留	98
存储系统资源利用率	99
共享的 RAID 组	105
后端总线性能测量	106
总线性能影响	106
第 11 章 可用性	107
可靠性	108
冗余	108
主动/被动体系结构	108
测量可靠性和可用性	110
可靠性指标	110
可用性指标	110
第 12 章 存储对象可用性	111
物理存储对象可用性	112
硬盘驱动器故障模式	112
硬盘驱动器可靠性分类	113
自我监视、分析和报告技术 (SMART)	113
RAID 可用性差异	114
镜像 RAID 级别可用性	114
奇偶校验 RAID 级别可用性	115
不受保护的 RAID 级别 (RAID 0)	116
降级模式中的 RAID 组操作	116
RAID 组重建	117
LUN 可用性	118
位置错误	118
LUN 验证	118
虚拟池可用性	119
可靠性	120
第 13 章 VNX 可用性	121
前端	122
存储处理器	122
主动/被动所有权模式	122
后端	123

系统驱动器和写缓存可用性	123
后端端口	123
全局热备份	124
重建日志	124
软件和固件更新	125
第 14 章 结论	127
附录 A 术语表	129

图 1	数据块存储环境中的组件	2
图 2	VNX 系列机型范围.....	3
图 3	VNX 存储系统方块图.....	4
图 4	详细的 iSCSI 堆栈.....	8
图 5	NFS /CIFS 堆栈对比	12
图 6	数据块协议摘要 iSCSI 层	13
图 7	机械硬盘驱动器吞吐量与带宽	17
图 8	用户响应时间概念视图	18
图 9	VNX 驱动器 DAE, 具有到 SAS 后端端口的双端口存储连接	37
图 10	VNX5300 示例配置	38
图 11	存储系统方块图	40
图 12	VNX 内存: 概念视图.....	45
图 13	VNX 后端概念图.....	47
图 14	VNX SAS 后端端口概念图	48
图 15	RAID 组概念图.....	59
图 16	镜像 RAID 概念图.....	60
图 17	奇偶校验 RAID 概念图.....	61
图 18	奇偶校验 RAID 组条带元素	61
图 19	40 个驱动器的 RAID 5 存储池概念图	67
图 20	LUN 概念图	68
图 21	存储池容量分配示例	73
图 22	虚拟池概念图	75
图 23	传统 LUN 概念视图	77
图 24	LUN 与物理驱动器的关系	78
图 25	MetaLUN 概念视图	78
图 26	串联的 metaLUN	79
图 27	分条的 metaLUN	79
图 28	串联组件 (metaLUN)	79
图 29	工作负载的物理资源配置示例 — 驱动器布局	82
图 30	驱动器典型服务时间比较	89
图 31	磁盘交叉, 未对齐的 LUN	90
图 32	对齐的 LUN	91
图 33	两个 15 驱动器 DAE 上绑定的概念 RAID 组	92
图 34	写缓存利用率	104
图 35	共享的 RAID 组驱动器	105
图 36	最佳和非最佳 I/O 路径	109

表 1	工作负载的 I/O 特征	27
表 2	各型号的 VNX 存储处理器存储模块	34
表 3	十进制与二进制容量	55
表 4	RAID 级别概要介绍	62
表 5	常见的 RAID 组可用容量 (TB)	66
表 6	物理资源调配示例工作负载	81
表 7	资源调配示例硬件配置	81
表 8	磁盘轴 rpm 与延迟的关系	87
表 9	驱动器性能因素 (VNX)	88
表 10	制造商报告的机械硬盘驱动器 UER	113

您可以将此手册当作一本教程来阅读，以对 VNX 基于存储的性能和可用性功能有一个基本的了解。本文：

- ◆ 对 VNX 存储系统进行了概述。
- ◆ 解释了在评估 VNX 的性能和可用性选项时需要理解的概念。
- ◆ 回答了 VNX 新用户经常会问到的许多问题。
- ◆ 提供了理解《EMC Unified Best Practices for Performance and Availability:Common Platform and OE Block 31.0—Applied Best Practices》（EMC Unified 性能和可用性最佳做法：通用平台和 OE Block 31.0—适用的最佳做法）白皮书（也称为《VNX 最佳做法》白皮书）所需的背景信息。如果您是 VNX 的新用户，则在阅读《VNX 最佳做法》白皮书之前应先阅读本文。
- ◆ 本文结尾部分提供了一个术语表，定义了许多 EMC 特定的术语。

如果您将本文用作参考，则需要确定您是需要 VNX 性能方面的帮助还是可用性方面的帮助，还是二者都需要。然后根据需要查找相应的章节并阅读整个章节。

受众

本文的受众为初次接触 EMC 存储系统的 VNX 存储系统管理员。它是一篇介绍性文章，适合在实施 VNX 数据块或文件存储系统最佳做法之前需要了解存储系统技术和功能的背景信息的用户。本文假定用户了解有关数据中心主机（服务器）、网络和 IT 概念的基本知识。文章开头部分在本质上是一般性介绍，越到后面越详细。有经验的读者可以跳过这些一般的介绍性章节。

术语说明：在本文中，术语 **驱动器** 既指机械硬盘驱动器，也指 **闪存驱动器**。闪存驱动器为基于非易失性存储的驱动器，在 IT 行业中通常称为固态磁盘 (SSD)。对于这两种不同的驱动器类型，将分别进行介绍。

参考资料

- ◆ 《EMC CLARiiON Storage Solutions:Microsoft Exchange 2007 — Best Practices》（EMC CLARiiON 存储解决方案：Microsoft Exchange 2007 — 最佳做法）
- ◆ 《EMC NetWorker PowerSnap and SnapView for CLARiiON》（EMC 针对 CLARiiON 的 NetWorker PowerSnap 和 SnapView）
- ◆ 《EMC Unified Best Practices for Performance and Availability:Common Platform and OE Block 31.0—Applied Best Practices》（EMC Unified 性能和可用性最佳做法：通用平台和 OE Block 31.0—适用的最佳做法）
- ◆ 《EMC VNX Series — A Detailed Review》（EMC VNX 系列 — 详细介绍）

有关本文档的问题、意见或更正？

如果您对本文档有任何问题、意见或更正，请在 [Powerlink](#) 上使用本文档标题旁边的“Feedback to Author”（向作者发送反馈）。作者或 EMC 代表会很快与您联系。

第 1 章 存储环境

本章包括以下主题：

存储系统	3
主机	5
存储网络	5

EMC[®] VNX[®] 存储系统位于包括以下组件的存储环境中：

- ◆ VNX 系列存储系统
- ◆ 主机
- ◆ 存储区域网络 (SAN)、网络连接存储 (NAS) 或聚合网络

在存储环境中，一个或多个存储系统（例如 VNX）通过 SAN 或 NAS 连接到一台或多台主机。这些主机与客户端交互，客户端可以是终端用户，也可以是其他计算机系统。如果有 NAS，则可以直接将存储系统连接到 Celerra/文件客户端。

可以将 VNX 存储系统配置为数据块级 (FC、FCoE、iSCSI)、文件级 (NAS — NFS 和 CIFS) 或统一存储系统 (文件和数据块)。针对存储的配置能力取决于机型，较高级别的机型可以同时配置文件和数据块，较低级别的机型只能配置一种。

数据块级别的存储使用逻辑数据块地址 (LBA) 写入和读取数据块，这些数据块将转换为驱动器上的磁盘扇区地址。（请参见第 51 页上的[第 8 章](#)了解详细信息。）SAN 存储环境使用数据块级别的存储，可提供高于文件级存储的性能级别。（请参见第 2 页上的图 1。）

备选体系结构为文件级存储系统。使用文件级存储的存储系统在数据块级访问之上添加一个抽象层；主机数据作为文件系统扩展区发送，必须先将这些文件系统扩展区映射到逻辑磁盘数据块，然后才能存储到硬盘驱动器上。术语网络连接存储 (NAS) 指文件级存储。

存储环境通常都很复杂。每个组件都有其自己的体系结构，这些体系结构会影响组件自身和整个存储环境的性能和可用性。

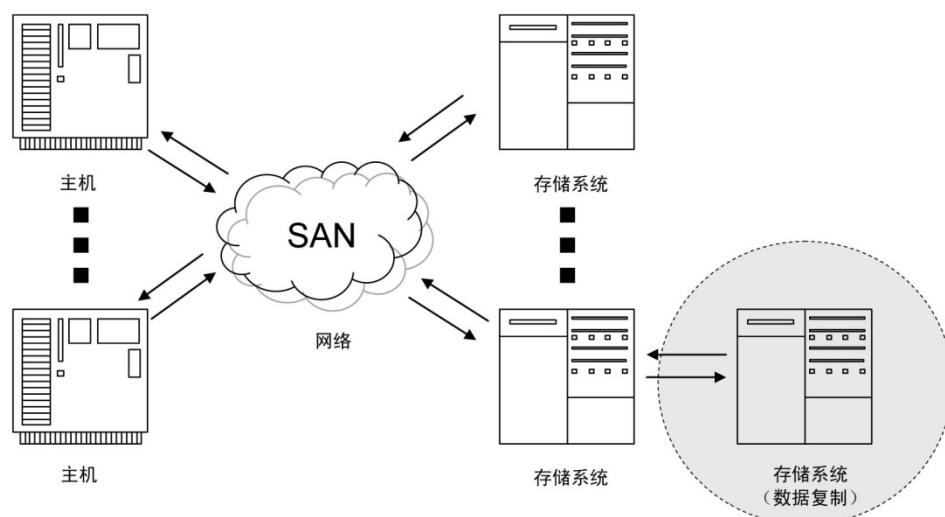


图 1 数据块存储环境中的组件

存储环境的关键衡量指标包括：

- ◆ 性能 — 存储环境为终端用户检索数据所需的时间。
- ◆ 可用性 — 存储环境为用户检索数据（尤其当环境中的某个组件发生故障时）的能力的衡量指标。
- ◆ 容量 — 存储系统中可存储的数据量。
- ◆ 灵活性 — 适应各种不同工作负载的能力。

存储系统

一般来说，存储系统分为：

- ◆ 入门级
- ◆ 中端
- ◆ 企业级

入门级系统通常用于满足小型企业的需求或用于构建大型分布式存储环境的其中一个部分。入门级系统最多可承载约 100 个驱动器。入门级系统的性能和可用性在不同系统之间的差别很大。EMC VNX5100 存储系统就是一个入门级存储示例。EMC 3000 系列 VNXe 产品线全部被视为入门级系统。本文不包括 VNXe 系列内容。

企业级系统用于满足大型企业或公共事业机构的集中业务需求。企业级系统为提供最高性能和可用性而设计和构建。较小的企业级存储系统可以承载大约 500 个驱动器，但更常见的是 1,000 个驱动器。大型企业级存储系统可承载超过 2,000 个驱动器。EMC Symmetrix[®] 产品线和最大的 VNX7500 型存储系统是企业级存储系统示例。本文不包括 Symmetrix 系列内容。

介于入门级和企业级系统之间的是中端存储系统。大部分 VNX 系列机型都是中端存储系统。（第 3 页上的图 2 显示了机型范围。）



图 2 VNX 系列机型范围

中端存储系统在存储系统性能和容量方面选择广泛。最小的中端系统在性能和容量方面与最大的入门级系统相似，但是中端系统具有更多功能和更好的可用性。较小的中端系统可扩展性也更高。可扩展性表示可轻松增加系统的性能和容量以满足不断增长的需求。例如，在 VNX 5500 中，可以很轻松地将系统中的驱动器数量从 150 增加到 225。

中端系统比入门级系统具有更好的可延伸性，也就是说可以很轻松地添加额外的硬件和软件功能来满足新的要求。例如，VNX 可通过以太网和光纤通道以多种协议进行通信。

存储系统所属的类别在某种程度上是弹性的。可以将最大的中端系统配置为与较小的企业级系统具有相似的性能和容量。可以将入门级系统调配为具有低端中型系统的功能。大型中端系统通常具有比企业级系统便宜的优势。

一般体系结构

如第 4 页上的图 3 所示的 VNX 存储系统具有模块化体系结构。VNX 的主要部件包括：

- ◆ 前端端口
- ◆ 存储处理器 — 由管理存储系统功能和内存的 CPU 组成。
- ◆ 镜像的存储系统缓存。
- ◆ 后端 — 存储处理器和驱动器之间的 I/O 接口。
- ◆ 存储 — 用于海量存储的机械硬盘驱动器和闪存驱动器。

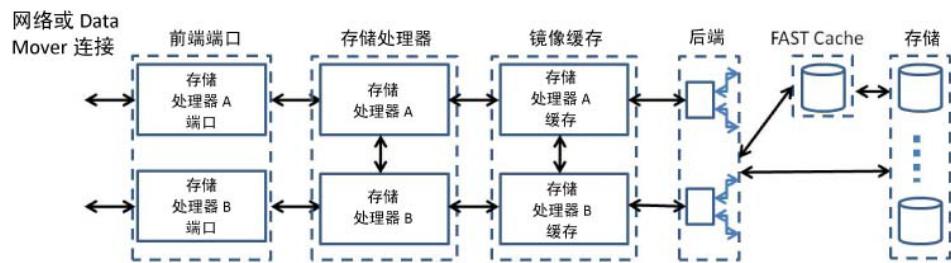


图 3 VNX 存储系统方块图

前端端口包含将存储系统直接连接至主机或通过以太网或光纤通道等网络通信将存储系统连接至主机的 I/O 端口。前端端口还可以直接连接至其他存储系统。在 NAS 配置情形中，前端端口会连接至 NAS 文件服务器（有时称为 *X-Blade* 或 *Data Mover*）。前端端口的数量和接口类型取决于机型。较高端的机型可承载的端口数量比入门级机型可承载的端口数量要多。

存储处理器是自定义设计的计算机，用于控制存储系统。每个 VNX 系统都有两个完全冗余的存储处理器。有时存储处理器称为 *SP*。存储处理器内的微处理器是多核的。多核是将多个 CPU 放在一块微处理器芯片上。它有助于基于高效多处理器（多 CPU）计算机的设备设计。将 CPU 内核群集在单个处理器芯片上可共享本地芯片总线和内存资源并缩短它们之间的物理距离，从而加速单独的 CPU 内核之间的通信速度。单一插槽芯片的电路板占地空间也比较小，允许将电路板上 CPU 的电源和冷却装置整合在一起。

镜像存储处理器缓存通过以下方法优化对存储的读取和写入：将读取和写入缓冲到高速存储处理器内存并提供高可用性。有单独的读取和写入缓存。读缓存使用预测性预读取技术在使用数据之前将其从存储中预取出来。写缓存可以对写入进行缓冲以平衡 I/O，合并写入并避免重写数据的需要。缓存的写入功能是完全冗余的（镜像），可以支持存储处理器冗余。

后端是将存储处理器连接至存储设备的 I/O 接口。

当我们使用术语“存储”时，指的是一种体系结构，该体系结构由大量组合在一起以形成子系统的存储设备组成。传统上，这是由驱动器组成的。这些驱动器曾是机械硬盘驱动器。也有更新的基于半导体的存储技术存储。硬盘驱动器可以包含多种类型的物理连接（SATA、SAS、光纤通道）、很大的容量范围以及多种转速。这些特征对性能和可用性有若干影响。

主机

主机是运行应用程序的计算机。主机可直接或通过网络连接到 VNX 存储系统。主机与存储系统相似，都有 CPU、内存和 I/O 资源需要管理。主机的性能和可用性范围很广，从高性能刀片服务器（在单个存储模块中包含许多单独的服务器，旨在提高效率、节省空间和提供高可用性），到性能适中的个人计算机 (PC) 类型的台式工作站，都包括在内。主机管理其 CPU 和 I/O 资源的方式会对存储系统的性能和可用性产生影响。

主机可以单独运行，也可以集成到执行相关功能的主机网络中。主机组成的网络有时称为群集。群集是一组主机服务器，这些主机服务器以分布式方式执行通用文件系统并以单个软件系统呈现其客户端。Microsoft 群集系统 (MSCS) 就是这样的体系结构。

主机也可以驻留在虚拟机 (VM) 上。虚拟机是模仿硬件环境的软件。虚拟机允许在主机计算需求发生变化时相应增加、减少、分区、分配或取消分配计算资源。虚拟机由虚拟机管理程序层提供支持，虚拟机管理程序负责管理每个虚拟机对基础硬件平台的访问。VMware ESX™ Server 和 Microsoft Hyper-V™ 是虚拟机类型的虚拟机管理程序的两个示例。虚拟机可以提高主机硬件的使用率，但当多个虚拟机争用同一基础平台和存储系统资源时，也可能会带来性能方面的难题。

主机执行系统和应用程序软件，两者都会影响连接的存储系统的性能和可用性。

系统软件包括操作系统 (OS)，例如 Microsoft Windows Server 2008、HP-UX 和 Linux。

应用程序包括 Microsoft Exchange Server、Oracle 11i 和 IBM WebSphere 等程序。应用程序生成大部分存储系统工作负载，但系统软件在启动和维护期间也会生成部分工作负载。可以通过以下多种方式将应用程序部署到存储系统上：

- ◆ 专用：单个应用程序的数据对存储系统具有独占使用权。这会创建单个存储系统工作负载。
- ◆ 共享：存储系统上会存储多个应用程序的数据。这会创建多个工作负载。
- ◆ 优化：应用程序数据跨多个存储系统分布。这会创建单个或多个工作负载，具体取决于应用程序的规模。

存储网络

网络的复杂性及其协议会对存储系统的性能和可用性产生影响。存储网络通常是 DAS、光纤通道 iSCSI SAN 或 NAS（请参见第 [7](#) 页上的[第 2 章](#)）。还可以一次将它们连接到多种网络类型。

直连存储 (DAS)

VNX 还可以直接连接到主机，存储系统和主机之间无需网络设备。这称为**直连存储 (DAS)**。DAS 的优势在于其简单的拓扑以及主机和存储之间的专用通信通道；不存在网络争用的情况。缺点是不能共享存储资源。DAS 连接主要用于本地存储。许多容量较小的入门级存储系统都是以这种方式配置的。DAS 连接也可用于 SAN 或 NAS。物理连接可以是光纤通道，也可以是以太网。

存储区域网络 (SAN) 和网络连接存储 (NAS)

对于数据块级的存储 SAN，光纤通道一直是最常用的网络技术。它具有高吞吐量和低延迟的特点，而且具有高度的可靠性。但通用的以太网网络被广泛用于存储系统部署站点。以太网的庞大客户群、低成本和相对简单等特征使其成为 SAN 用途的极具吸引力的选择。

连接到 VNX 的 FC SAN 使用**光纤通道(FC)** 协议，而以太网 SAN 使用 iSCSI 协议。文件级存储 NAS 在 TCP/IP 和以太网上使用 NFS 和 CIFS 协议。

SAN 或 NAS 的性能和可用性在很大程度上取决于**网络拓扑**。网络拓扑是网络元素的物理和逻辑排列。网络元素是组成网络的组件，例如高容量/高可用性路由器、光纤或铜缆连接以及光纤通道或以太网交换机。

聚合网络

聚合网络是用于满足多个数据通信需求的单一网络。只需维护单个网络，而无需维护单独的存储、LAN 和进程通信网络。需要选择网络的连接结构，以满足其性能和带宽要求，例如低延迟或高带宽。聚合网络在连接结构的用户之间共享适配器、缆线连接和交换机。

如果无需考虑延迟问题，则可以使用以太网来聚合存储相关的通信，而无需维护单独的光纤通道数据块存储 (SAN) 和以太网文件存储 (NAS) 网络。以太网光纤通道 (FCoE) 通过以太网运行 FC 协议。随着 FCoE 协议的引入，可以更容易地将传统的光纤通道 SAN 迁移到以太网连接结构。SAN 的 iSCSI 通信和 NAS 通信可以共享一个以太网网络。这成为将 SAN 和 NAS 通信聚合到单个网络连接结构的一个示例。

第 2 章 常见协议

本章包括以下主题：

协议堆栈	8
物理和数据链路协议	9
网络协议	10
文件协议 (CIFS/NFS)	12
数据块协议 (SCSI)	13

协议是一套规则，用于控制计算机与网络上其他设备之间的通信。很多协议都描述了如何实际传输数据，以及如何控制、管理和解析主机和存储系统之间传输的数据。

协议堆栈

协议设置标准。协议也具有分层结构。它们互相层叠。有时这也称为 **堆栈**。简短的协议探讨只需涉及以下三个堆栈级别：

- ◆ 物理和数据链路层协议 — 描述网络铜质或光纤缆线的规格以及网络设备的操作。
- ◆ 网络层协议 — 通过物理层传递的数据是使用一种或多种 **网络协议** 进行发送的，这些网络协议规定数据的发送和接收方式。
- ◆ 应用程序层协议 — 描述数据的解析方法。例如，它可以是应用程序数据，也可以是控制通信的命令。在 VNX 上，这些数据为读和写 I/O 请求。

例如，读和写 I/O 请求通过光纤通道或以太网网络到达存储系统。光纤通道网络连接至光纤通道前端端口。以太网网络连接至 iSCSI 和 FCoE 前端端口。读写 I/O 是 SCSI 协议命令。

iSCSI 协议是用于传递 SCSI 协议命令的若干网络协议之一。第 8 页的 [图 4](#) 显示了 iSCSI 堆栈详细信息。

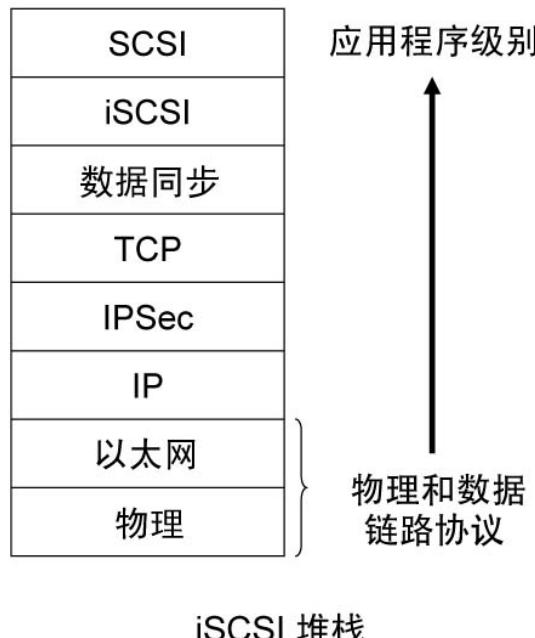


图 4 详细的 iSCSI 堆栈

物理和数据链路协议

有三个主要的物理和数据链路协议用于在存储系统上发送和接收数据，以及在存储系统内部发送数据。它们是：

- ◆ 光纤通道
- ◆ 以太网
- ◆ SAS

这些协议是网络的物理层协议，用于将主机连接至存储系统，将 VNX 的存储处理器连接至其驱动器。

如前面所述，VNX 的前端端口是光纤通道或以太网协议。VNX 系列存储处理器还通过后端的多个 SAS 接口连接至存储系统驱动器。

连接至存储系统处理器总线的驱动器使用 SAS 连接。SAS 协议是存储处理器和 SAS 存储设备之间的点到点连接。

光纤通道协议

光纤通道协议被标准化为[美国国家标准协会 \(ANSI\)](#) 标准 INCITS 387。

如上所述，光纤通道是一个分层协议，它是一个完整堆栈。每一层描述协议的一个不同的方面，从缆线上的位的物理传输到高级别命令和数据的解析。请注意，人们通常将光纤通道堆栈的一个层或所有层简单表述为光纤通道。

需要理解两个重要的光纤通道堆栈层：

- ◆ FC-PH
- ◆ FC-4：协议映射层

光纤通道协议规定设备控制器彼此通信所使用的总线接口

FC-PH

FC-PH 是实际包含四个层（FC-0 到 FC-3）的物理层。这些层组成光纤通道缆线连接，包括数据链路。数据链路将一个位置与另一个位置相连接；数字加密信息通过数据链路进行传输。

在 VNX 上，光纤通道可提供与主机的前端通信。光纤通道协议的运行速度包括：1 Gb/s、2 Gb/s、4 Gb/s、8 Gb/s 和 10 Gb/s。

FC-4

FC-4 是一个网络层，SCSI 协议消息在该层中以光纤通道协议数据包的形式进行封装以便传送。下面将对该层进行探讨。

以太网协议

以太网协议被标准化为电气与电子工程师学会 (IEEE) 标准 [IEEE 802.3](#)。以太网包含单个物理层。

以太网网络连接至 VNX 的前端以太网端口。根据工作负载通信中使用的中间网络协议，这些以太网端口称为 *iSCSI* 端口。以太网协议的运行速度包括：10 Mb/s、100 Mb/s、1 Gb/s 和 10 Gb/s。1 Gb/s 速度称为 *GigE*。10 Gb/s 速度为 *10 GigE*。

SAS协议

串行连接SCSI (SAS) 是一种用于设备连接的 ANSI 标准 (ANSI INCITS 376)。

SAS 是数据块级协议。它提供点到点的串行接口，设备控制器可通过此接口直接相互链接。在 VNX 中，SAS 是存储处理器到海量存储驱动器控制器之间的连接。SAS 整合了 SCSI 和 SATA 协议。它综合了 SCSI 的设备效用和可靠性与 SATA 的串行体系结构的优点。

SAS 协议的运行速度包括 3 Gb/s 和 6 Gb/s。

网络协议

可使用多种网络协议来进行与 VNX 的数据通信。它们是：

- ◆ 光纤通道
- ◆ TCP/IP
- ◆ FCoE
- ◆ iSCSI

这些协议描述了存储系统如何传输和接收数据。它们还控制和管理发送方和接收方之间的连接。

请注意，网络协议可以通过不同的物理协议发送。例如，经过光纤通道协议格式化的数据可以通过 TCP/IP（如下所示）网络发送。

光纤通道协议

这是 FC-4 层光纤通道协议。光纤通道协议传送嵌入式 SCSI 命令，以通过光纤通道网络进行读取和写入。它还包括控制单元。控制单元是与承载数据的消息分隔开的离散消息。这些消息管理传递数据所用的连接。

TCP/IP 协议

TCP/IP 也称为 *Internet* 协议。它实际上是两个标准化协议“叠加”在一起。TCP（传输控制协议）为较高层协议；用于为需要通过 Internet 传输的数据和消息创建数据包并确保它们按顺序到达。IP 为较低层协议，负责在数据包中添加地址以确保数据包到达正确的目的地。IP 还可以与其他高层协议配合使用。

Internet 工程任务组 (IETF) 对 TCP/IP 协议进行了标准化，并将它们记录在一套名为 RFC (请求注解) 的文档中：

- ◆ RFC 791 中介绍了 IPv4。
- ◆ RFC 2460 中定义了 IPv6。
- ◆ RFC 793 中介绍了 TCP。

与光纤通道类似，TCP/IP 是一个载波协议，用于对 iSCSI 协议数据包进行封装。

FCoE 协议

以太网还可用于传送 SAN 上的光纤通道数据包。以太网光纤通道 (FCoE) 对光纤通道数据包进行封装，以通过基于以太网的 SAN 进行传送。这使光纤通道能够使用高带宽和长距离以太网网络，同时保留光纤通道协议。

iSCSI协议

iSCSI 是 Internet 小型计算机系统接口协议，用于通过基于 IP 的以太网网络传送数据块级数据。iSCSI 层位于 TCP/IP 协议之上，用于传送 SCSI 命令。

iSCSI 在 RFC 3720 中进行了标准化。

请注意，TCP/IP 和 iSCSI 协议需要同时在以太网网络上使用。iSCSI 用于 SAN 工作负载通信，TCP/IP 用于其他网络通信。但是，建议将 SAN 通信从物理上或以虚拟方式 (VLAN) 与所有网络通信分隔开来。

虚拟 LAN (VLAN)

存储网络 VLAN 会对以太网网络进行分区以提供更多资源。

VLAN 操作是 TCP/IP 协议套件的一部分，并在 IEEE 802.1q 中进行了标准化。

VLAN 会创建逻辑冲突和广播域。采用 VLAN 广播域，并非所有的消息都会发送到目的地。通信仍会实际并存于同一缆线连接、集线器、交换机和路由器上。但是，创建 VLAN 域会对广播和多播通信带来限制。VLAN 的桥接软件定义要包括在广播中的节点。

如上面所述，建议针对 iSCSI（存储）使用专用网络。从物理上隔离存储环境的网络通信会带来更高的网络性能和更大的可靠性。但有时候，物理分隔存储网络是不实际的，可用的以太网网络可能需要同时满足存储通信和一般用户的网络服务需求。如果不使用专用以太网 LAN，则通过未用于一般 LAN 通信的路由器将 iSCSI 通信分隔到自己的物理 LAN 网段，或使用 VLAN。

将存储通信限制在其自己的 VLAN 是一种不正确的安全方法。数据的保密性无法仅通过 VLAN 得到保护。只有通过实施 CHAP 或 iSCSI 身份认证才能确保 VLAN 上的数据得到正确的安全保护。

CHAP 协议

通过使用挑战握手认证协议 (CHAP) 可为 iSCSI 网络上的数据提供安全保护。RFC 1994 对 CHAP 进行了定义。

CHAP 是用来对连接的另一方进行身份认证的协议。它基于对称加密，对称加密是在对等方之间共享密钥。在 SAN 上，对等方为主机和存储系统。明文是未加密的文本。使用 CHAP，主机和存储系统都知道密钥的明文。但密钥不通过网络发送。

SNMP 协议

简单网络管理协议 (SNMP) 用于管理 SAN 或 NAS 的网络设备。其主要任务是允许主机从任意网络节点（例如主机、交换机和路由器）获取统计数据。管理信息库 (MIB) 用作 SNMP 中的主要资源。托管网络上的管理数据在协议中当作变量使用。协议还可以通过远程修改这些变量来修改和重新配置网络。这些变量以分层结构方式组织：这些分层结构和变量元数据由 MIB 进行描述。SNMP 使用 UDP 作为其通信中的传输协议，因为 UDP 开销低、轻量并且简单。

网络端口聚合

网络端口聚合允许将多个活动以太网连接到同一交换机，以显示为具有单个 MAC 地址的单一链路。这会增加连接的总体带宽。

有两种类型的端口聚合，链路聚合控制协议 (LACP) 和以太网通道。LACP 是一种开放式标准，可在所有支持 IEEE 802.3ad 标准的交换机上使用。以太网通道是 CISCO 系统专用的。它在 CISCO 交换机上与 CISCO 的 EtherChannel™ 标准结合使用。

请注意，端口聚合是一种逻辑连接。到主机的物理连接受限于物理以太网端口的带宽。

在聚合端口上，通过使用 MAC 地址在物理链路之间分发数据。存储系统端口和网络交换机都需要进行配置更改才能实施端口聚合。

文件协议 (CIFS/NFS)

通用接口文件系统 (CIFS) 和网络文件系统 (NFS) 是与 NAS 配合使用的主要网络协议（参见第 12 页的图 5）。这两种协议都可以在受支持的客户端上使用，但 NFS 通常用于 Linux 客户端，CIFS 通常用于 MS Windows 客户端。

RFC 1094 中定义了 NFS。CIFS 是旧的 SAMBA (SMB) 协议的更新，比 SMB 更受欢迎。CIFS 是一种 Microsoft Windows 协议。

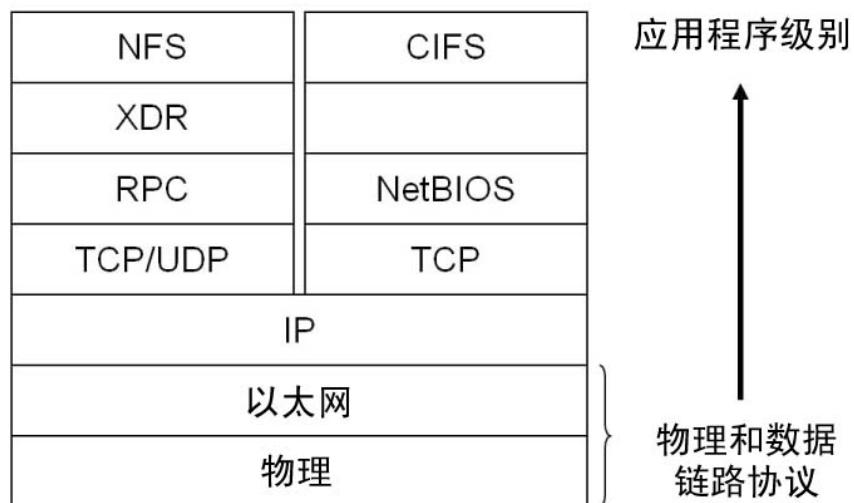


图 5 NFS /CIFS 堆栈对比

请注意，NFS 依赖于远程过程调用 (RPC) 和外部数据表示 (XDR)。RPC 是客户端程序用来调用远程服务器程序上的函数的一套函数调用。XDR 是一个例程库，用于转换进程之间的数据格式。此外，NFS 既可使用 TCP，也可使用用户数据报协议 (UDP)。UDP 是一个不可靠的无连接协议，由 RFC 768 和 1122 定义。它是一种数据报服务。使用 UDP 不能保证数据会到达其目的地。UDP 旨在以很少的传输开销提供服务。

CIFS 依赖于 NetBIOS 协议。NetBIOS 是一种 Microsoft 协议，用于控制计算机之间的会话并维护连接。

数据块协议 (SCSI)

小型计算机系统接口 (SCSI) 是用于向驱动器发送命令（包括读写 I/O）的应用程序层协议。I/O 有时称为 *SCSI* 命令。SCSI 是在主机与 VNX 上由其驱动器组成的 LUN 之间连接和传输数据的标准。

SCSI 命令可以包含数据或控制信息。它们由主机应用程序发送，并通过 iSCSI 协议或光纤通道协议消息单元以嵌入式方式到达。第 [13](#) 页上的图 6 显示了主存储系统FC 和 iSCSI 协议之间的关系。



图 6 数据块协议摘要 iSCSI 层

第3章 性能和可用性衡量指标

本章包括以下主题：

带宽	16
吞吐量	17
响应时间	18
可用性	19
RPO 衡量指标	19
RTO 衡量指标	19
RPO/RTO 示例	19
描述工作负载	22
I/O 特征和工作负载	26
了解工作负载	27

性能是指系统完成的工作量与所用的时间和资源的比值。时间和资源需可衡量，才能判断性能。高性能的定义取决于生产目标。VNX 系列存储系统通过多种操作配置文件实现高性能，从高带宽、高延迟到高吞吐量、低延迟。

可用性指存储系统即使在发生硬件故障时也能为用户提供访问其应用程序和数据的能力，发生硬件故障有时称为降级状态或模式。像 VNX 系列这样的中端系统被归为高可用性系统，因为它们能在发生单个故障时继续提供对数据的访问，但通常，降级模式下的操作性能会有某种程度的下降。

标识适用于资源的衡量指标是构建体系结构或衡量存储系统的性能和可用性的一个重要部分。用来衡量系统的性能和可用性的最常用的术语包括：

- ◆ 带宽
- ◆ 吞吐量
- ◆ 响应时间
- ◆ 可用性
- ◆ 恢复时间目标 (RTO)

不能将可用性衡量指标与一般可用性相混淆，后者是避免非工作系统状态所需的可靠性和冗余的组合。

带宽

带宽是每秒钟可以沿通道传输的数据量的衡量方法。存储系统带宽以每秒钟多少兆字节 (MB/s) 或每秒钟多少千兆字节 (GB/s) 进行衡量。注意，使用大写字母 *B* 表示字节。通信网络带宽通常以每秒钟多少位 (b/s) 进行衡量。

存储系统带宽通常描述顺序或大型数据块 I/O 性能，但它可以描述任何工作负载。

存储系统的可提供带宽直接受连接层的带宽（也称为连接带宽）的影响。连接带宽通常以每秒钟多少兆位 (Mb/s) 或每秒钟多少千兆位 (Gb/s) 进行衡量。注意，使用小写字母 *b* 表示位。

请务必记住，存储系统带宽是以每秒钟多少字节衡量的，而连接带宽是以每秒钟多少位衡量的。连接带宽和存储系统驱动器的带宽之间经常会发生转换。此外，最大指定带宽（称为有线速率或有线速度）与实现的实际带宽之间存在区别。

由于数据包开销和网络协议的影响，数据始终以低于额定带宽的速率传输，因为存在协议开销。

例如，考虑到一个字节是八个位，再加上协议开销，1000 Mb/s 的最大千兆位以太网 (GigE) 速度约等于 119 MB/s。

吞吐量

吞吐量是每秒钟处理的 I/O 操作量 (IOPS)。此衡量方法取决于请求的大小。从单个组件（例如硬盘驱动器）到整个存储系统的吞吐量性能都可以测定。吞吐量衡量方法通常用来描述随机和小型数据库 I/O 性能。

吞吐量可以进一步分为：

- ◆ 读取 IOPS：每秒的读 I/O 数
- ◆ 写入 IOPS：每秒的写 I/O 数
- ◆ 总 IOPS：每秒的读写 I/O 总数

进行这种区分的原因是为读取操作和写入操作提供服务时消耗的存储系统资源量不同。对于存储系统 I/O，经常看到总 IOPS 与读写 IOPS 比率一起标示。例如“10,000 IOPS, 70% 读取”。

存储设备上的带宽和吞吐量

请务必理解存储设备上带宽与吞吐量之间的关系。这种关系与 I/O 大小相关。也就是说，随着 I/O 操作（以每秒钟多少字节进行衡量）以数据块大小增长，每个单位时间传输的数据量也相应增长。但是，对于恒定带宽，随着 I/O 大小的增加，I/O 率（吞吐量）会下降。还可以另一种方式来看这种关系：随着 I/O 操作按数据块大小增加，带宽也增加。

第 17 页的图 7 显示了典型的机械驱动器的吞吐量与带宽的关系。左边的轴是吞吐量。右边的轴是带宽。

红线是带宽，右侧是其 MB/s 测量轴。绿线是吞吐量，左侧是其 IOPS 测量轴。

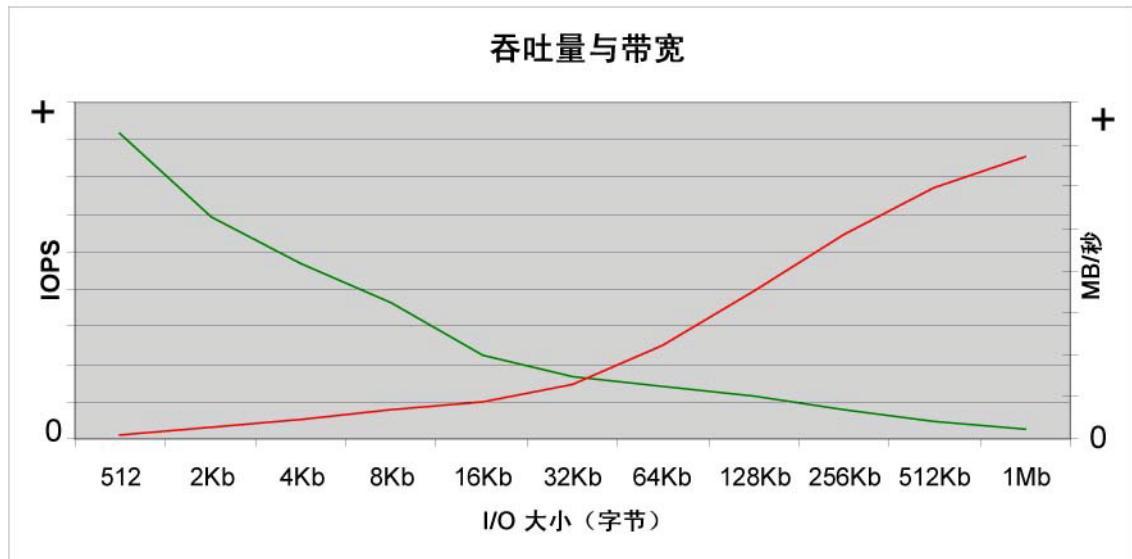


图 7 机械硬盘驱动器吞吐量与带宽

从图中，您可以看到如果 I/O 大小是一个小的 512 字节，则带宽（红线）比较低，但吞吐量（绿线）比较高。相反，随着 I/O 大小增加，带宽（红线）上升，吞吐量（绿线）下降。图表最右边显示得最清楚。

请注意，如果不知道 I/O 大小，任何关于某个设备或接口能够支持“多少多少”GB/s 的带宽和“多少多少”IOPS 的吞吐量的讨论都是没有意义的。知道您的 I/O 大小是进行带宽/吞吐量分析的第一步。

响应时间

响应时间衡量从查询或服务请求结束到请求开始完成之间所用的时间。本文将讨论两种类型的响应时间：用户和 I/O。

用户响应时间是用户对系统（包括存储系统）的“体验”。I/O 响应时间是仅限存储系统的衡量指标，用于衡量满足主机读取或写入请求所花的时间。

用户响应时间

用户期望在各种不同的操作条件下都能获得一致的高级别性能。用户性能的主要衡量指标为用户响应时间。有时，这称为客户端响应时间或主机响应时间。用户响应时间是系统响应请求所需的时间（通常为毫秒）。这可以是查询请求、报告生成，也可以是简单的终端屏幕刷新。通常，有一个服务质量 (QoS) 协议来量化系统架构师和管理员承诺给用户的性能。

在第 18 页的图 8 中所示的系统中，每个组件都有其自己的响应时间（“T”）。

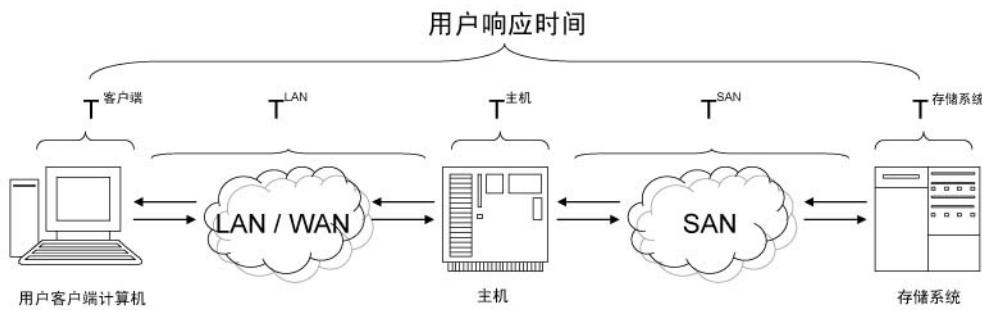


图 8 用户响应时间概念视图

用户响应时间是一种非常高级别的衡量方法。用户响应时间是传输和接收系统内所有子系统的响应时间之和。这些子系统可能包括：

- ◆ 用户的工作站（客户端）
- ◆ 网络（局域网 (LAN) 或广域网 (WAN)）— 将客户端计算机连接至其主机服务器
- ◆ 一个或多个主机服务器
- ◆ 存储区域网络 (SAN) — 将服务器连接至一个或多个存储系统
- ◆ 作为 I/O 的源或目标的存储系统

具有长响应时间的单个子系统会给总体用户响应时间带来负面影响。

例如，存储系统可能会有较短的 I/O 响应时间，而中间的 LAN 因为网络拥堵具有较长的响应时间。这会导致总体的用户响应时间变长。

用户响应时间长是指示存储环境中存在瓶颈的几个首要指示之一。但这并不指示瓶颈所在的位置或其根本原因。

I/O 响应时间

I/O 响应时间用来衡量完成单个 I/O 所需的时间。可将其应用于来自单个驱动器（例如硬盘驱动器）的读取或写入，或应用于到 LUN（由多个驱动器组成）的读取或写入。在陈述响应时间时，务必弄清楚在哪个对象上进行测量。

I/O 响应时间包括在队列中等待的时间和服务请求所花的时间。队列是临时存储区域，I/O 在其中等待执行。如果服务时间短，队列也短，则 I/O 响应时间也会短。长队列会增加响应时间。

此衡量方法通常以毫秒 (ms) 为单位，但有些设备会以微秒 (μs) 作为响应时间。

I/O 响应时间和带宽

为基于小型数据块吞吐量的工作负载提供服务时，很容易实现低响应时间。为带宽工作负载实现低响应时间可能更为困难。基于大型数据块大小带宽的工作负载的响应时间可能会是不同的数量级。这主要是因为每个 I/O 服务需要较长时间。

可用性

可用性衡量当客户端提出相关请求时系统能够返回数据的时间百分比。降级性能不包括在此衡量指标中。

例如，五个 9 的可用性表示 99.999% 的可用性。将该百分比转换为数据不可用的总时间为每年 5 分 15 秒。

在开发数据恢复计划时，使用以下衡量指标。它们确定数据需要备份的频率，以及必须准备好什么类型的保留策略，才能确保满足推动这些需求的 SLA。

RPO 衡量指标

恢复点目标 (RPO) 指有风险的数据量。RPO 由数据保护事件之间的时间量确定，并反映在灾难恢复期间可能有丢失风险的数据量。此衡量指标指示存在丢失风险的数据量。此衡量指标表述为时间间隔，可以短至几秒钟，也可以长达数小时。

RTO 衡量指标

恢复时间目标 (RTO) 与可用性相关。RTO 是将出故障的应用程序恢复到正常工作状态所需的时间。RTO 衡量指标确定存储系统的高可用性要求的数量级。此衡量指标可以表述为时间间隔，可以短至几毫秒，也可以长达数天。

RPO/RTO 示例

假定关系数据库有一个两小时的 RPO 和一个 5 分钟的 RTO。

要确保不丢失超过两小时的用户数据，每两小时进行一次 SnapView 快照可能就足够了，因为快照支持短 RTO。

假定数据库的 RPO 为 15 分钟，RTO 为 30 分钟。

此案例需要采取不同的策略，因为 RTO 比 RPO 要长很多。此情形中的恢复计划可能要求每 6 个或 8 个小时进行完整数据库备份，每 15 分钟记录一次事务日志。持续时间较长的 RTO 由恢复通过“重做”日志记录提供支持。

第 4 章 工作负载配置文件

本章包括以下主题：

描述工作负载	22
I/O 特征和工作负载	26
了解工作负载	27

工作负载是在一段时间内分配给资源的工作单元数量。

不熟悉存储对象的读者可能需要先阅读[存储对象](#)章节，然后再阅读本章，以熟悉本部分中的某些术语。

VNX 的工作负载是它每秒钟接收和传输的读和写请求的数量。整个系统的应用程序（如何使用、如何配置以及何时使用这些应用程序）会对存储系统的工作负载产生很大影响。

存储系统可以包含一个或多个应用程序的数据。承载单个应用程序的存储系统为专用存储系统，因此具有单个工作负载。许多存储系统都存储多个应用程序或多类应用程序的数据。例如，存储系统可以承载 NAS，也可以为执行数据块级 I/O 的应用程序提供服务。这会带来两个或更多的工作负载。此外，应用程序可能会随着时间推移而更改其操作配置文件，这也是应用程序所执行的任务的一个结果。

总工作负载是单个应用程序工作负载之和。多个工作负载可能会导致互补或冲突的存储系统资源使用情况。如果工作负载以相同方式使用存储系统的资源并且会注意“资源预算”，则这些工作负载为互补工作负载。如果工作负载以完全不同的方式使用存储系统资源，或者相互排斥，则这些工作负载为冲突的工作负载。需先了解每个应用程序的工作负载，然后才能了解总的系统工作负载。

例如，NAS 系统通常执行小型数据块随机 I/O。备份等应用程序执行大型数据块顺序 I/O。共享相同物理存储对象的小型数据块随机和大型数据块顺序工作负载就是一个冲突工作负载示例。

最后，有些应用程序的数据分布在多个存储系统上。这种情况按处理多个应用程序的方式来处理。

描述工作负载

I/O 类型

主机应用程序的运行设计（如何以及何时使用这些应用程序）会对存储系统负载产生影响。能够描述工作负载的 I/O 对于理解适用哪种最佳做法非常重要。

应用程序工作负载产生的 I/O 具有以下广泛特征：

- ◆ 写入与读取
- ◆ 顺序与随机
- ◆ 大型数据块大小与小型数据块大小
- ◆ 高局部性与低局部性
- ◆ 稳定与猝发
- ◆ 多线程与单线程

写入与读取 I/O

需要知道并量化由应用程序执行的写操作与读操作的比率。这是因为它们以不同的方式消耗存储系统资源。知道应用程序执行的读操作与写操作的比率可直接影响适用于缓存、RAID 组和 LUN 资源调配的最佳做法。

写操作消耗的存储系统资源比读操作消耗的要多。针对存储系统写缓存的写入将同时镜像到两个存储处理器 (SP)，最后通过后端发送到存储设备。写入到 RAID 组时，镜像或奇偶校验数据保护技术会消耗额外的时间和资源。此外，包含闪存驱动器的存储设备的写入速度比读取速度要慢。

读操作通常会消耗较少的存储系统资源。例如，读操作不需要创建冗余数据或元数据以提供数据保护。此外，大多数存储设备执行读操作的速度都比写操作要快。在存储系统的主缓存或次缓存中找到数据（*缓存命中*）的读操作，其消耗的资源比未命中的读操作消耗的资源要少。这类读操作的主机响应时间最低。但是，未在缓存中找到数据的读操作（*缓存未命中*）的响应时间要远高于命中的读操作的响应时间。这是因为必须从驱动器中检索数据。

顺序与随机 I/O

需要知道并量化应用程序所执行的 I/O 类型。知道 I/O 类型可确定适用于您的工作负载的缓存配置、RAID 级别保护和 LUN 资源调配的最佳做法。

一个应用程序可以有三种类型的 I/O：

- ◆ 顺序
- ◆ 随机
- ◆ 混合

存储系统处理读写操作的水平取决于工作负载主要是顺序 I/O 还是随机 I/O。

随机 I/O 中 I/O 操作的源或目标“散落”在存储设备的整个地址空间。这会导致机械硬盘驱动器寻道。（请参见第 52 页上的“物理”。）顺序 I/O 中 I/O 的源或目标按顺序排列。I/O 请求要么彼此相邻，要么相互离得很近。

小型随机 I/O 使用的存储系统资源比大型顺序 I/O 使用的存储系统资源要多。（请参见下一节。）随机 I/O 吞吐量受存储系统内许多其他因素的影响。仅执行顺序 I/O 的应用程序的带宽比执行随机或混合 I/O 的应用程序的带宽要高。处理同时包含这两种 I/O 类型的工作负载需要进行分析和平衡，以确保带宽和吞吐量都能得到优化。

注意，使用闪存驱动器是一个例外。闪存驱动器是本机随机访问设备。它们在处理随机 I/O，尤其是小型数据块随机 I/O 时非常有效。有关更多详细信息，请参见下一节。

大型数据块大小与小型数据块大小 I/O

有必要知道工作负载的应用程序所使用的主要 I/O 大小和 I/O 大小的分布。这可确定适用于您的工作负载的缓存配置、RAID 组和 LUN 资源调配的最佳做法。

每个 I/O 都有固定的和可变的资源成本，主要取决于 I/O 大小。请注意，此定义已随时间推移发生了变化，较大型的数据块大小 I/O 变得更常见。出于本文目的，达到并包括 16 KB 的 I/O 被视为小型 I/O，32 KB 及更大的 I/O 被视为大型 I/O。在 VNX 上执行大型 I/O 可提供比执行小型 I/O 更好的带宽。

低主机响应时间需要低访问时间。小型数据块随机访问应用程序（例如在线事务处理 (OLTP)）比使用顺序 I/O 的应用程序的访问时间要短很多。此类 I/O 可能会受到物理存储设备的最大 IOPS 的限制。

使用较小还是较大的 I/O 数据块大小通常取决于应用程序。使用大的请求还是将其分为较小的顺序请求的决策可能需要在应用程序级别、在主机总线适配器 (HBA) 及其存储系统 LUN 进行配置。

高局部性与低局部性 I/O

在规划使用次缓存和存储分层时，了解工作负载应用程序的局部性非常重要。这决定了次缓存容量和分层资源配置的最佳做法。

由于虚拟池中包括 FAST Cache 的次缓存和 FAST Virtual Pools (FAST VP) 功能的分层存储，数据局部性变得重要起来。

局部性基于数据集的 **访问局部性**。访问局部性指频繁访问的存储位置。局部性适用于随机访问工作负载。顺序工作负载没有局部性。

有两类局部性：“写入时间”和“写入地点”。

相比三年前创建和写入的数据，应用程序更有可能访问今天的数据。数据写入时间或时间局部性指在短时间内对存储位置进行再次访问。短持续时间指在数秒或最多数小时之内。

最近创建的数据很有可能位于驱动器扇区或磁道上相互邻近的机械硬盘驱动器的 LBA 上。数据位置或空间局部性指使用中数据在存储对象上其地址空间中的分布。这可能会导致数据存储在机械硬盘驱动器上附近的扇区或附近磁道的扇区上。但对于闪存驱动器这样的随机访问存储设备，它指的是地址空间。在大多数情况下，很有可能出现邻近最近访问的位置的地址也会被访问。

次缓存和自动分层利用 I/O 请求的局部性来实现更高的吞吐量，方法是确保具有高局部性的活动访问数据位于具有最低响应时间的存储设备上。活动访问数据有时称为**工作数据集**。这些功能由时间和空间局部性触发。

具有高访问局部性的工作集的工作负载可通过次缓存和存储分层获得最佳性能。理解数据集内的局部性程度也很重要。应用程序不同，局部性程度也不同。局部性程度是指使用率最高的数据集的百分比。请注意，局部性是一种“统计”分布。绝大部分的用户数据可能都位于一个相对受限的地址空间内。但可能有一些离群者——到此空间之外的地址的 I/O。3% 到 5% 的局部性程度很常见，但也很有可能达到 20%。某些数据类型（元数据是一个典型），例如面向对象的数据库索引表，可能有很高的局部性。

例如，一个包含 20% 的工作数据集的 1.2 TB 数据库大约有 250 GB 的频繁访问的容量。数据库内的索引表可能具有高局部性。它们相对紧凑并且经常被访问。同样，数据库中也会有非常大但不经常被访问的表；它们具有较低的局部性。FAST 缓存和 FAST VP 层可从容量角度适当调整，以满足组成工作集的高局部性用户数据的需求。

请注意，有些应用程序具有非常低的局部性。例如，具有完全随机 I/O 的工作负载会具有可能的最低局部性。基准应用程序（例如公共域 IOMeter™ I/O 子系统衡量和特征化工具能够生成完全随机的工作负载。

VNX 存储系统管理员应当与他们的存储和应用程序架构师协商其应用程序的局部性。

稳定与猝发 I/O

需要知道 I/O 模式以及 I/O 模式何时、如何以及多久发生变化，以确定适用于您的工作负载的资源利用率的最佳做法。

到存储系统的 I/O 通信可以是稳定的，也可以是不定时的。不定时 I/O 有时称为猝发 I/O。通信模式也可能随着时间的变化而变化，例如长时间不定时，然后又变为稳定。下面的做法很常见：在业务时间段内将存储系统配置为供随机访问应用程序使用，然后在非工作时间进行备份和批处理时，将存储系统重新配置为要求良好的顺序性能。

猝发行为会导致通信高峰。高峰是突然、非完全可预测的大量活动增加。要应对高峰，需要保留一定余量的存储系统性能资源。这包括未提交的 SP 利用率容量、I/O 带宽和存储容量。需要进行这种保留，以应对最严峻形势下的高峰需求。否则，如果在繁忙期间出现高峰，用户响应时间可能会受到严重影响。

多线程与单线程

有必要了解您的工作负载 LUN 使用的是什么 I/O 线程模式。这可确定适用于您的工作负载的最佳做法，尤其是对于闪存驱动器的使用。

工作负载的并发程度是任何时候对存储系统发出的尚未完成的 I/O 请求的平均数。并发是一种通过使用存储系统上的多个驱动器来实现高性能的方法。当出现更多的 I/O 请求时，驱动器会变得忙碌起来，I/O 开始排队，从而增加响应时间。但是，如果应用程序的 I/O 队列提供一个稳定的 I/O 流，应用程序可以达到其最高吞吐量。

这些 I/O 请求分配给存储系统的方式取决于线程模式。

线程是软件程序中执行某项功能的一系列命令。基于主机的应用程序会创建进程，进程包含线程。线程可以是同步的，也可以是异步的。同步线程会等待其 I/O 完成后才继续执行。这种等待有时称为挂起。异步线程不会挂起。它们会继续执行，并且可能发送其他 I/O 请求，完成后处理每个请求，可能不会按发出时的顺序处理 I/O 请求。

单线程访问表示一次只有一个线程可以执行到存储（例如 LUN）的 I/O。以前，许多大型数据块顺序工作负载都是单线程并且是同步的。异步单线程仍然可以实现高聚合性能，因为其队列中的多个 I/O 可实现并发。多线程访问表示有两个或多个线程同时执行到存储的 I/O。来自应用程序的 I/O 变为并行 I/O。这会带来更高级别的吞吐量。过去，小型数据块随机工作负载是多线程的。但现在您会发现，许多大型数据块顺序工作负载也是多线程的。

VNX 存储系统管理员应当与他们的存储和应用程序架构师协商其应用程序的线程模式。

应用程序缓冲和并发

许多企业应用程序执行它们自己的 I/O 缓冲来合并文件更新。Microsoft Exchange、Microsoft SQL Server 和 Oracle 等应用程序使用应用程序缓冲来智能地管理 I/O 并提供低响应时间。

例如，某些数据库会定期对其自身重新进行索引，以确保低响应时间。可从 [Powerlink](#) 上找到有关许多特定应用程序的缓冲配置（也称为缓存配置）的详细信息。白皮书《EMC CLARiiON Storage Solutions:Microsoft Exchange 2007 — Best Practices Planning》（EMC CLARiiON 存储解决方案：Microsoft Exchange 2007 — 最佳做法规划）对应用程序的缓存配置提供了具体建议。

应用程序并发解决了应用程序内到单个对象（例如表格行）的同步读取和写入的冲突需求。它尝试避免覆盖、非重复性读取（读取之前发生了变化的值）和阻止。I/O 并发性越高，存储系统的性能就越好。

许多应用程序都可以配置为在内部调整并发。请查阅工作负载应用程序的配置文档，以了解它们在并发配置方面的最佳做法。

I/O 特征和工作负载

下面介绍了六种常见的工作负载，用以展示 I/O 的众多特征。

- ◆ 在线事务处理 (OLTP)
- ◆ 邮件系统
- ◆ 文件服务器
- ◆ 决策支持系统 (DSS)
- ◆ 磁盘备份
- ◆ 富媒体

OLTP 工作负载出现在用于以事务处理为导向的应用程序的数据库系统中。这些系统的示例包括销售订购、库存控制和电子银行。这种工作负载属于随机访问、多数读取、小型数据块并且对吞吐量很敏感。

邮件系统用于为所有规模的组织发送消息和整合服务应用程序。邮件系统是一种常用的存储系统应用程序。Microsoft Exchange Server 是一个邮件系统示例。这种工作负载属于随机访问、混合读写并且对吞吐量很敏感。

文件服务工作负载出现在用作网络连接存储 (NAS) 设备的存储系统中。这种工作负载要求存储系统维持大量的 IOPS。这种工作负载属于随机访问、多数读取并且对吞吐量很敏感。

DSS 工作负载是指存储系统承载一个事务数据库，该数据库接受针对不同类型的数据类型的大量大表格的复杂查询。以前，这种工作负载是顺序访问，较新的应用程序现在包含随机访问。在这两种情况下，I/O 都是多数读取并且对带宽很敏感。

磁盘备份工作负载是使用基于磁盘的存储（而不是磁带）进行归档存储的备份过程。存储系统会拷贝数据、数据库或服务器来创建源材料的归档副本。将主存储备份转储到基于辅助磁盘的存储比转储到基于磁带的存储花的时间要少。如果需要恢复数据，基于磁盘的备份也会更快。磁盘上的备份数据已经联机，可供快速恢复数据使用。这种工作负载属于大型数据块、顺序访问、多数写入并且对带宽很敏感。

富媒体工作负载是存储系统在多媒体流中的使用。多媒体流为众多终端用户提供视频和声音。这种工作负载属于按流的顺序访问、多数读取并且对带宽很敏感。为大量的流提供服务时，这种工作负载可能多为随机的。

第 27 页的表 1 汇总了这些工作负载通常观察到 I/O 模式。

表 1 工作负载的 I/O 特征

工作负载	I/O 类型		访问类型		I/O 大小		I/O 流		描述性衡量指标	
	随机	顺序	读取	写入	小	大	稳定	猝发	吞吐量	带宽
OLTP	X		X	X	X			X	X	
邮件系统	X		X	X	X	X		X	X	
文件服务器	X		X	X	X	X		X	X	
DSS	X	X	X			X	X			X
磁盘备份		X		X		X	X			X
富媒体		X	X			X	X			X

请注意，有些工作负载可能会同时选择一个类别中的两个类型。这表示混合使用不同的访问类型，具体取决于基础架构。

例如，在邮件系统工作负载中，I/O 大小不会明显表示为小型或大型数据块。通常还会包括以下详细信息：准确描述工作负载的说明；例如 OLTP 工作负载可能简明地描述为“生成随机 I/O 的应用程序。I/O 总共创建 20,000 个 IOPS，其中 70% 为读取。I/O 大小为 16 KB”。

了解工作负载

要实施最佳做法，应先了解您的存储系统的工作负载。这要求具有主机应用程序知识。请记住，当工作负载的需求超出存储系统的性能能力时，应用性能最佳做法没有什么效果。

性能是一种评估行为。维护系统性能的历史记录很重要。在应用任何最佳做法之前使用性能衡量指标判断结果可节省可观的时间和劳力。

新工作负载在进入生产之前应在非生产环境中进行基准测试。请与您的 EMC 销售代表联系，让 EMC USPEED 专业人员为您讲解在进入生产之前评估和模拟新系统性能方面的相应技术。

最后，请注意工作负载或总体系统配置是否有任何变化，以便了解变化对整体性能的影响。定期使用 Unisphere Analyzer 监视和分析性能是一种谨慎的做法。使用 Analyzer 定期监视可提供基线性能指标，用于进行历史对比。这些信息可以提供有关计划外性能变化的早期警报。

第 5 章 软件

本章包括以下主题：

操作环境	30
Unisphere	30
Unisphere Analyzer.....	30
复制分层应用程序	30

本章提供对 VNX 软件的简短概述以及新用户需要熟悉的一些重要功能。

操作环境

操作环境 (OE) 控制 VNX 系列存储系统的运行。VNX OE for Block 管理存储系统的所有 I/O 功能。OE Block 是可扩展的，因此它可根据配置提供各种不同的复制功能。

VNX OE for File 是 VNX NAS 机头的操作环境。NAS 机头又称为 X-Blade 或 Data Mover。OE File 也是 VNX NAS 控制台的一部分。

Unisphere

Unisphere 是一套应用程序，用于配置、监视和管理 VNX 存储系统。不同的 Unisphere 程序适用于不同的存储环境。

Unisphere Manager

Unisphere Manager 提供基于浏览器的界面，用于系统管理。通过 Unisphere Manager，可以管理多个 EMC 存储系统和主机。常见的管理任务包括：

- ◆ 对存储对象进行资源调配（配置），这些对象包括驱动器、RAID 组、LUN、文件系统、检查点等等。
- ◆ 管理缓存
- ◆ 监视错误
- ◆ 调配虚拟 LUN

Unisphere CLI

Unisphere 命令行界面 (CLI) 提供到 VNX 的命令行界面。有少量的高级调整任务只能通过 CLI 执行。安全 CLI (naviseccli) 提供基于安全凭据的界面，以强制实施用户角色和添加审核功能。

CLI 还可用于脚本编写。CLI 脚本是包含 Navisphere 命令语句的可执行文件。脚本可用于自动执行存储系统管理任务。

Unisphere Analyzer

Unisphere Analyzer 是 VNX 性能分析工具。它提供的衡量指标是性能和可用性调整的重要组成部分。Unisphere Analyzer 允许查看（实时或离线）存储系统的性能以识别性能瓶颈。Unisphere Analyzer 也包含一个 CLI 界面。

记录有关存储系统工作负载的历史性能信息是性能和可用性调整的第一步。

复制分层应用程序

复制分层应用程序（有时称为 分层应用程序）是存储系统应用程序。它们提供独立于主机的服务，这些服务对于整个系统维护和可用性很重要。这些应用程序为可许可软件，必须随存储系统单独购买。

数据复制是独立于主机运行的 可用性功能。数据复制分层应用程序包括：

- ◆ MirrorView™
- ◆ RecoverPoint

- ◆ SAN Copy™
- ◆ SnapView™
- ◆ SnapSure™

这些数据复制应用程序的重要区别在于本地复制与远程复制。本地复制在本地存储系统创建拷贝。远程复制在其他存储系统（有时是不同地理位置的存储系统）上创建拷贝。

[Powerlink](#) 上提供了针对各个复制分层应用程序的最佳做法。

MirrorView

MirrorView/S 和 MirrorView/A 是 VNX 系列存储系统支持的可选软件应用程序。这两款应用程序都提供远程数据复制。

MirrorView/S 将一个或多个 LUN 从一个存储系统同步（并且一致）镜像到另一个存储系统。在正常操作期间，次数据映像与主数据保持同步。

MirrorView/A 将一个或多个 LUN 从一个存储系统异步（并且一致）镜像到另一个存储系统。次映像定期与主映像保持同步。可根据部署的体系结构调整同步时间间隔。这通常是距离方面的考虑。

RecoverPoint

RecoverPoint 是一个数据保护应用程序，提供 LUN 的本地和远程数据复制。它可与任意存储系统配合工作。它可在称为 RecoverPoint 应用装置的 SAN 网络元素上运行，也可在主机上运行。还可以安装其重定向功能（称为拆分器），以直接在存储系统上运行。

SAN Copy

SAN Copy 是一个数据迁移应用程序。它支持在存储系统之间或存储系统内部进行批量数据传输。这些存储系统不必是 EMC 生产的系统。使用 Open Replicator，可以在无需主机参与的情况下将数据从一个存储系统传输到另一个存储系统。

SnapView

SnapView 是一个应用程序，允许创建存储系统数据的时间点拷贝（称为快照）或完整的本地镜像（称为克隆）。此功能有时称为本地复制。这些拷贝可用于在线备份或数据复制。其他应用程序（例如开发测试）可直接访问拷贝。还可将拷贝用于分载生产主机的备份活动。

SnapSure

EMC Celerra SnapSure 支持创建生产文件系统 (PFS) 的时间点逻辑映像。PFS 的时间点视图称为检查点。使用检查点，您可以从在线备份恢复单个文件或整个文件系统。

第6章 VNX 物理体系结构

本章包括以下主题：

存储模块度量	34
存储处理器存储模块	34
备用电源	35
磁盘阵列存储模块	37
X-Blade 存储模块	37
控制台存储模块	37
物理配置示例	38
硬件文档	38

VNX 系列具有架装式模块化物理体系结构，每个机架由两个或多个存储模块组成。每个存储模块包含执行一个或多个系统功能的硬件。VNX 使用下列存储模块类型：

- ◆ 存储处理器存储模块 (SPE) 或磁盘处理器存储模块 (DPE)
- ◆ 备用电源 (SPS)
- ◆ 磁盘阵列存储模块 (DAE)
- ◆ Data Mover 存储模块 (X-Blade 存储模块)
- ◆ 控制台

通过将存储模块合并到配置中来配置不同的机型、存储容量以及存储系统的使用。这通常包括：针对存储处理器性能选择 SPE 或 DPE、针对存储容量选择 DAE 的数量和型号，以及添加或省略文件相关的 X-Blade 和控制台。

存储模块度量

机架单元（或 U）是一个 EIA 标准度量单元，用于描述要安装到设备机架中的 IT 设备的高度，具体值为 1.45 英寸或 482.6 毫米。VNX 机架最高可达 40U。VNX 系列存储系统可以是一个或多个机架，具体取决于安装的驱动器的数量。

存储模块的宽度符合 IT 19 英寸机架标准。IT 计算设备应用最广泛的是 19 英寸（482.6 毫米）标准。

所有 VNX 存储模块在宽度上都采用 19 英寸架装标准。不同的存储模块类型有不同的类型，具体取决于其功能和型号。存储模块的厚度也各不相同。

存储处理器存储模块

VNX 存储处理器位于存储处理器存储模块中。有两种可供存储处理器使用的存储模块：存储处理器存储模块 (SPE) 和磁盘处理器存储模块 (DPE) 请参见第 34 页的表 2。请注意，DPE 是一种存储处理器存储模块，它也可以装载驱动器。

请注意，SPE 的高度为 DPE 的 2/3。但是，在每种配置中，SPE 都需要一个 DAE (O/S DAE)。SPE 占用的机架空间比 DPE 要多。

每个存储系统只有一个 SPE 或 DPE。

表 2 各型号的 VNX 存储处理器存储模块

VNX 型号	存储处理器存储模块类型	高度 (U)
VNX5100	DPE	3
VNX5300		
VNX5500		
VNX5700	SPE	2
VNX7500		

基于存储处理器存储模块 (SPE) 的系统

SPE 存储模块提供连接、缓存和冷却，但不提供驱动器。这种设计可提供最灵活的部署选项。

基于磁盘处理器存储模块 (DPE) 的系统

DPE 存储模块在一个存储模块中提供连接、缓存和数个驱动器。这些系统可提供最高的存储密度，因为没有单独的 SPE 占用架装空间。

备用电源

VNX 系列存储系统具有冗余电源、冗余配电和电池备份，以确保不受内部和外部电源故障的影响。

以下内容仅适用于交流电 (AC) 供电的存储系统。也有直流电 (DC) 供电的系统，但本文不作讨论。请与您的 EMC 代表联系，了解有关配置了直流电源的存储系统的详细信息。

VNX 采用双共享电源。VNX 5100、5300 和 5500 的 DPE 中只有两个电源。一个电源可使写缓存保持启用状态以维持正常操作。有两个冷却模块，一个 SP 一个。一个冷却模块可使写缓存保持启用和正常操作状态。

在 VNX 5700 和 7500 中，每个 SPE 有四个电源/冷却模块。也就是说，对于机柜中的每个存储模块，有两个电源共同为它配电。如果一个电源出现故障，另一个正常工作的电源可以为存储模块提供支持。

VNX 机柜配备有配电装置 (PDU)。机柜中的 DAE 类型的存储模块直接插入 PDU 获得电能。SPE/DPE 不直接插入 PDU。每个存储模块都连接至左边的一个 PDU 和右边的一个 PDU。这确保在发生电源故障时有冗余。如果任意一侧的电源发生故障，另一侧可以维持整个系统的电源需求。

备用电源 (SPS) 是一个电源监视和电池备份单元 (BBU)。电源监视确定数据中心电源质量变化或干扰是否会对存储系统的运行带来负面影响并触发电池备份功能。SPS 可在出现短暂的外部电源波动和完全断电时提供写缓存保护。

有些 VNX 机型只能配置一个 SPS。所有机型都可以配置两个 SPS。双 SPS 能够提供高于单 SPS 安装的存储系统可用性级别。存储系统只需要一个运行正常的 SPS 即可启用写缓存。维护和启用写缓存的同时仍能保证数据完整性是一项重要的性能特性。

SPS 的工作方式与不间断电源 (UPS) 的工作方式不同。其设计目的不是也不能长时间保持存储系统正常运行，而是期望电源能很快恢复。断电时，SPS 会发挥作用，以保护缓存中的数据并将其转储到存储区。存储区位于存储系统的前四个驱动器上，如果出现长时间中断，存储区可以无限期存放缓存内容而不会丢失数据。缓存保存时间各不相同，从几秒到几分钟都有，具体取决于机型、缓存容量和系统驱动器类型。

SPS 通常为 1U 单元。在基于 SPE 并且配置了包含超过 25 个驱动器的 DAE/OS 的 VNX 机型中，SPS 为一个 2U 单元。

SPS 电池电源

SPS 使用具有不同额定安时的电池类型。（安时是电池容量指标。）这些电池具有适当大小，可以支持连接的 SP 和存储系统组件，以维持写缓存，直到将其存储到存储区的操作完成。此过程有时称为到存储区的转储。

SPS 就绪测试

通过两种方式测试 VNX 的 SPS 系统的就绪性：在存储系统通电时，以及在运行期间定期测试（每周）。两种测试都会验证电源布线的配置和正确性，并确保 SPS 电池有足够的电量来支持连接的 SP 和存储系统组件，以维持写缓存，直到转储到存储区的操作完成。如果每周 SPS 测试会导致写缓存被禁用，则不会执行每周 SPS 测试。可以通过 Unisphere 配置每周测试的时间。不能配置测试时间间隔。

VNX 交流电源故障行为

虽然会有各种各样的故障，但 VNX 系列及其永久缓存仍可继续运行。（请参见第 123 页上的“系统驱动器和写缓存可用性”。）发生交流电源故障时存储系统的行为取决于故障原因和安装的 SPS 数量。交流电源故障情形包括以下组件的掉电：

- ◆ 两个 SPS
- ◆ 双 SPS 系统中的一个 SPS
- ◆ 单 SPS 系统中的唯一 SPS
- ◆ 单 SPS 系统中不受 SPS 保护的 SP

SPS 的交流电断电：这是最严重的情况。其中一个示例是整个数据中心的交流电发生故障。发生交流电断电情况时，所有 SPS 都会向操作环境 (OE) 报告“电池联机”。OE 会定期短时间轮询 SPS 的状态，以确定故障是临时掉电还是完全中断。如果所有 SPS 在轮询时段继续报告“电池联机”，则会启动到存储系统存储区的写缓存转储。将写缓存写入到存储区后，OE 会继续检查 SPS 状态。如果“电池联机”状态持续，DPE 或 SP 以及 DAE O/S 将关闭。如果交流电源恢复，SP 和 Bus 0 Enclosure 0 将自动启动并恢复操作。

双 SPS 系统中的一个 SPS 交流电掉电：如果向存储系统供电的两个数据中心交流电装置其中一个出现故障，就会出现这种情况。在交流电完全断电的情况下，SPS 会向连接的 SP 和对等 SP（仍在通过其 SPS 提供的交流电运行）报告“电池联机”。在这种情况下，OE 将关闭没有交流电输入的 SPS 以阻止不必要的电池电量消耗。两个 SP 都会通过连接到仍在接收交流电的 SPS 的电源继续运行。存储系统的写缓存仍处于启用状态。

单 SPS 系统中的唯一 SPS 交流电掉电：SPS 向其连接的 SP 和对等存储处理器（仍在通过交流电运行）报告“电池联机”。请注意，对等 SP 没有连接 SPS。如果 SPS 在轮询时段持续报告“电池联机”，将启动到存储区的写缓存转储。写缓存转储完成后，存储系统将在没有写缓存的情况下继续运行，这可能导致性能降级。这种情况将持续到 SPS 的交流电恢复并且 SPS 电池完全充满电为止。

单 SPS 系统中不受 SPS 保护的 SP 交流电掉电：SP 向 OE 报告交流电故障，但存储系统会通过对等 SP 的电源继续运行。写缓存仍保持启用状态。请注意，安装单个 SPS 的 VNX 5100 和 VNX 5300 例外。每个 SP 的电源供其自己的 SP 使用。它们不与对等 SP 的装置相互冗余。它们通过第二个交流主电源为自己提供冗余。

磁盘阵列存储模块

磁盘阵列存储模块 (DAE) 提供到 SPE 或 DPE 的磁盘驱动器扩展。DAE 的类型有多种：VNX 系列 DAE 可承载多达 60 个驱动器。此外，还支持两种驱动器外型尺寸：2.5 英寸和 3.5 英寸。这两种外型尺寸需要不同的 DAE。这两种外型尺寸的驱动器都不能支持所有驱动器计数。

存储系统中的 DAE 数量取决于机型。基于 DPE 的存储系统不需要任何 DAE。基于 SPE 的存储系统至少需要一个 DAE。

DAE O/S

在非 DPE VNX 存储系统（参见第 27 页上的表 1）上，承载存储系统的系统驱动器的 DAE 称为 *DAE O/S*。系统驱动器是包含操作环境的文件系统的驱动器。请注意，Bus 0 Enclosure 0 指承载存储系统的系统驱动器的 DAE，它包括系统驱动器。

DAE 和存储

DAE 提供到驱动器和存储处理器的冗余连接。驱动器（机械硬盘或闪存）通过端口接收其 I/O 请求。所有驱动器都是双端口配置。DAE 中的每个驱动器都连接至 SP 的 SAS 后端端口。双端口存储可随时通过一个或两个 SP 接收 I/O 请求。

到驱动器和存储系统的后端端口的 DAE 内部连接基于 SAS 协议。任何类型的驱动器：SAS、闪存或 NL-SAS 可以与任何其他类型的驱动器一起承载在 DAE（或 DPE）中。

例如，在第 37 页的图 9 中，Bus 0 将 SP A 和 SP B 连接到 DAE0 中的硬盘驱动器 0。

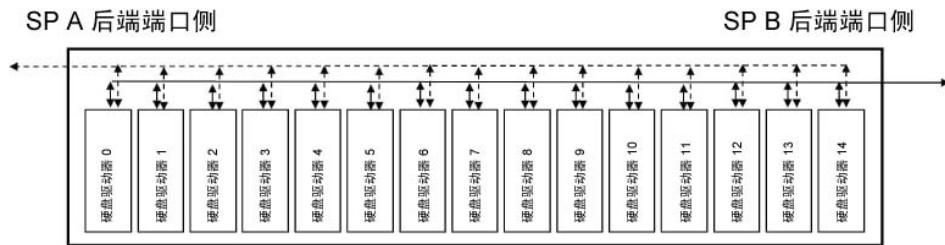


图 9 VNX 驱动器 DAE，具有到 SAS 后端端口的双端口存储连接

VNX 系列 DAE 由链路控制卡 (LCC) 提供支持。每个 DAE 都有两个 LCC。LCC 是后端总线、DAE 与驱动器之间的接口。（请参见[驱动器和 DAE](#) 部分了解更多信息。）

X-Blade 存储模块

X-Blade 是文件存储的网络和存储接口设备，有时称为 *Data Mover* 或 *NAS 机头*。

控制台存储模块

控制台是文件存储的管理接口设备。

物理配置示例

第 38 页的图 10 显示了 VNX5300 的正面（左侧）和背面视图，这是一个针对 NAS 工作负载而配置的入门级存储系统。

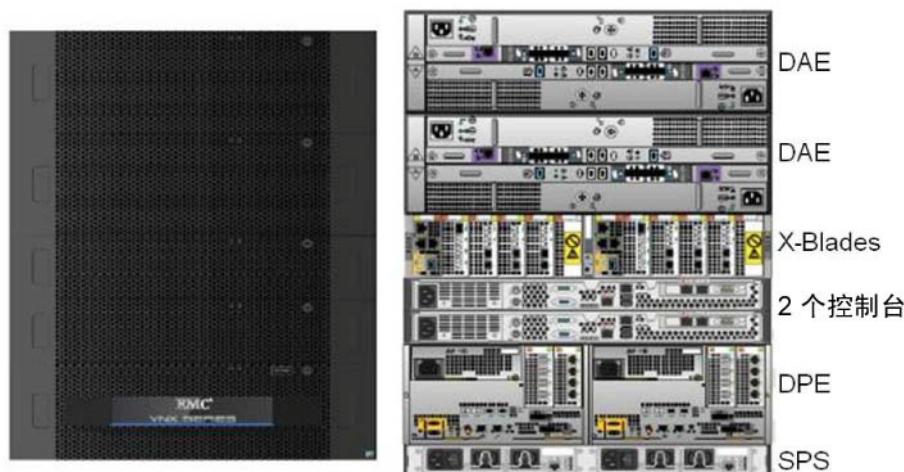


图 10 VNX5300 示例配置

该图显示的 VNX5300 包含 7 个存储模块。该配置总共包含 14U，高度大约为 20 英寸 (52 cm)。这些存储模块从下往上依次为：

- ◆ 双备用电源 (SPS)
- ◆ 包含双存储处理器的 DPE
- ◆ 2 个控制台
- ◆ 双 X-Blade
- ◆ 2 个 DAE (包含 15 个外型尺寸为 3.5" 的驱动器)

请注意其模块化体系结构。可以通过添加额外的 DAE 增加显示的系统存储容量。VNX5300 可包含的驱动器最大数量为 125。可以向该配置添加另外五个包含 15 个驱动器的 DAE。如果需要更大的存储容量或额外的存储处理器，可以将 DPE 升级到 VNX5500 型号的 DPE。

硬件文档

为了更好地熟悉 VNX 硬件，我们建议您阅读《EMC VNX 系列简介 — 详细介绍》。可以从 [Powerlink](#) 上下载此文档。

第 7 章 VNX 逻辑体系结构

本章包括以下主题：

物理存储对象	52
逻辑存储对象	59
物理资源配置示例	80

以下各节介绍了 VNX 存储系统的逻辑体结构。

VNX 从逻辑上分为以下部分：

- ◆ 前端端口
- ◆ 存储处理器
- ◆ 读/写缓存
- ◆ 后端
- ◆ FAST Cache (可选的辅助缓存)
- ◆ 存储

第 40 页的图 11 显示了这些主要逻辑组件之间的关系。显示顺序一般如本图所示，顺序为从左到右。

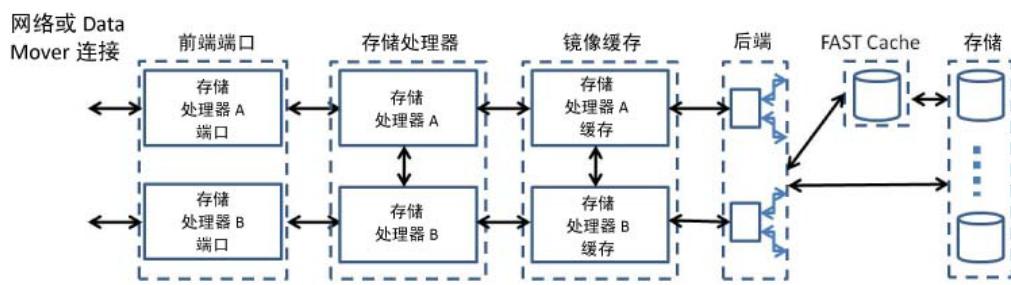


图 11 存储系统方块图

前端端口

每个存储系统都有其自己的前端端口。端口指与网络的物理缆线连接。前端端口可以连接到 Data Mover，直接连接到主机，或在 SAN 环境中通过网络连接到主机。

VNX 前端端口

可以为 VNX 系列配置：可采用以下物理协议的 UltraFlex™ I/O 模块：

- ◆ 光纤通道
- ◆ iSCSI
- ◆ 以太网光纤通道 (FCoE)

iSCSI 端口仅支持 iSCSI 协议，即使使用标准以太网端口也如此。硬件模型和用户界面也会将它们当作 iSCSI 端口，而不是以太网端口。

存储系统的端口数量和类型标准取决于机型。所有 VNX 型号都设计为可以添加额外端口，或添加不属于基本配置的端口。与低端型号相比，高端 VNX 型号能够承载更大数量的端口。通常，应当使用最少数量的前端端口来满足工作负载的 I/O 性能和可用性要求，而将尽可能多的扩展插槽留待将来扩展时使用。

需要的端口数量和类型通常取决于应用程序的工作负载和主机连接。在某些情况下，前端端口要求可能由主导工作负载的吞吐量或带宽决定。有些前端端口会自动预留给 MirrorView。

始终成对安装 I/O 模块 — 一个模块安装在 SP A 中，另一个模块安装在 SP B 中。两个 SP 均必须在相同插槽中安装相同类型的 I/O 模块。额外的 I/O 模块可占用任何可用插槽。

请注意，光纤通道、iSCSI 和 FCoE 标准在继续演变，尤其在带宽方面。UltraFlex 端口的模块化允许将较高的带宽连接翻新为标准开发。

端口位置

根据 VNX 的型号，I/O 端口可能位于 DPE 或 SPE 中。产品文档中也会提供这些信息。

管理和服务端口

VNX 系列存储系统的每个 SP 有两个额外的以太网端口，它们不属于前端端口，而是管理端口和服务端口。这些端口专门用于通过 Unisphere 访问和管理 SP。它们不支持 iSCSI 协议，也不能用于工作负载数据通信。

自动协商

如果支持，端口的速度可以自动调整为较低的带宽交换机或 DAS 连接，但会导致带宽损失。这种自动调整称为 *自动协商*。在自动协商中，网络对等双方可交换其链路级别协议功能，以提供兼容服务。默认情况下，自动协商由众多前端端口执行。并非所有 VNX 前端端口都支持多种速度，因而不支持自动协商。

例如，10 Gb/s iSCSI 和 10 Gb/s FCoE 端口不会自动协商为较低的速度。

光纤通道端口

光纤通道端口通过光纤通道 SAN 与主机进行通信，或直接与主机通信。主机必须具有光纤通道 HBA。这些端口还可以直接连接到其他存储系统或 X-Blade 的光纤通道端口。

VNX 系列上的光纤通道前端端口为 8 Gb/s 光学接口标准。

VNX 系列上的标准光纤通道前端端口使用光学 OM3 缆线作为其标准接口。这些缆线有时称为 M5E。

如果使用光缆布线，缆线类型中的“OM”指的是光纤的质量。这称为带宽距离积 (BDP)。其测量单位为 MHz/km。数值越高，可以通过更远距离传输的数据越多。

例如，OM4 可在 550 米的范围内支持 10 Gb/s，而 OM3 只能在 300 米的范围内支持该速率。

光纤通道端口会自动调整到所有组件都支持的最高速度。支持较低的带宽交换机或直接连接，但会导致带宽损失。并非每种协议速度都可以协商。

例如，8 Gb/s 光纤通道端口不会自动向下协商到 1 Gb/s 的速度。

光纤通道端口的文件和 MirrorView 支持

将保留机载端口，以用于存储处理器与 File X-Blade 之间的光纤通道连接。此外，还会默认指定用于数据复制的 MirrorView 端口。MirrorView 和 X-Blade 端口之间不会共用端口。这些端口是在存储系统首次通电时就自动分配的。但可以针对全数据块或无 MirrorView 工作负载的存储系统手动覆盖这些分配。

iSCSI 端口

iSCSI 端口通过以太网 SAN 与主机通信，也可以直接与主机的 iSCSI HBA 或网络接口卡 (NIC) 通信。还可以将这些端口直接连接到其他存储系统的 iSCSI 端口。

iSCSI 端口通过千兆以太网 (GigE) 或 10 GigE LAN 与主机通信。1 Gb/s iSCSI 和 10 Gb/s iSCSI 端口位于不同的模块上。1 Gb/s 和 10 Gb/s iSCSI 端口都是可选购项目。

VNX 系列上的标准 1 Gb/s iSCSI 前端端口使用标准的 CAT 6 GigE 铜缆连接与 LAN 或主机进行接口连接。10 GigE iSCSI 接口为铜缆或光缆连接。铜质接口是活动双轴缆线连接，通常称为 *Twinax*。包含 LC 连接器的光缆 OM3 是一种典型的接口。

GigE 端口会自动调整到较低带宽的交换机或直接连接，这会导致带宽损失。关于 iSCSI 端口可以自动向下协商到多低，存在一些限制。

FCoE 端口数

FCoE 端口通过以太网 SAN 与主机通信。不能将 FCoE 端口直接连接到主机或其他存储系统的端口。

FCoE 端口只能用于 10 Gb/s 通信。需要使用 10 Gb/s 聚合增强以太网 (CEE) 全双工网络，以支持在以太网光纤通道 (FCoE) 环境中进行无损耗通信。

主机必须具有能进行 FCoE 协议传输的聚合网络适配器 (CNA)。EMC 支持列表标识了当前受支持的适配器列表以及（所有）软件要求。有一种替代方法是主机具有光纤通道 HBA 并使用网络交换机将 FCoE 传输转换为光纤通道传输。

VNX 系列中的 FCoE 前端端口使用相同的 10 GigE 铜缆或光缆作为 10 Gb/s iSCSI 端口。FCoE 端口不会从 10 GigE 自动向下协商。

前端端口性能

对于较小的 I/O 大小，光纤通道和 iSCSI 端口的 IOPS 性能在其大部分总体性能上都相似。但光纤通道端口可提供高于 iSCSI 端口的带宽。此外，使用 iSCSI，存储处理器的 CPU 利用率会稍微高一些。

FCoE 端口具有与 4 Gb/s 光纤通道端口相似的性能。

端口类型之间的 IOPS 性能区别取决于协议以及端口取消 I/O 排队方面的效率。在这方面，光纤通道端口比 iSCSI 更高效。

但是，除非工作负载是带宽密集型或 I/O 很大 (> 64 KB)，否则，光纤通道与 iSCSI 端口之间的性能差异很小。带宽密集型工作负载的示例包括：富媒体、磁盘备份和 DSS。

扇入

扇入是指使用单个端口的主机的数量。可以连接至 VNX 存储系统的主机 HBA（称为启动器）的最大数量取决于机型。虽然某些机型支持每个 SP 超过 256 个启动器，但单个端口不能超过 256 个启动器。不过，知道有多少主机能够连接到端口是一个重要的考虑因素。

每个端口的启动器估计

在接端口的级别不实施启动器限制。有可能出现超额预订端口的情况，虽然这种可能性不大。通常，与端口能力相比，单个主机的 I/O 要求会比较适中。但可以将多个主机扇入单个端口。

可通过以下方式快速估计扇入到某个端口的主机：

主机端口 IOPS = 端口 IOPS / 主机 IOPS

主机端口带宽 = 端口带宽 / 主机带宽

例如，假定有一类生产主机需要 250 IOPS 的吞吐量和 10 MB/s 带宽。假定此示例中，每个存储处理器只有一条路径。

单个 4 Gb/s 光纤通道端口可以处理 360 MB/s 的带宽和 50k IOPS。单个 GigE iSCSI 端口可以处理 80 MB/s 的带宽和 10k IOPS。

可以将多少个描述的主机扇入单个 4 Gb/s 光纤通道端口？可以将多少台服务器扇入 GigE iSCSI 端口？

- ◆ 服务器 IOPS: 250
- ◆ 服务器带宽 (MB/s): 10
- ◆ 1 GigE 端口 IOPS: 10,000
- ◆ 1 GigE 端口带宽 (MB/s): 80
- ◆ 4 Gb/s 光纤通道 IOPS: 50,000
- ◆ 4 Gb/s 光纤通道带宽 (MB/s): 360
- ◆ 按端口 IOPS 的光纤通道主机 = $200 = 50,000 \text{ IOPS} / 250 \text{ IOPS}$
- ◆ 按端口带宽的光纤通道主机 = $36 = 360 \text{ MB/s} / 10 \text{ MB/s}$
- ◆ 按端口 IOPS 的 iSCSI 主机 = $40 = 10,000 \text{ IOPS} / 250 \text{ IOPS}$
- ◆ 按端口带宽的 iSCSI 主机 = $8 = 80 \text{ MB/s} / 10 \text{ MB/s}$

在这两种情况下，带宽都是限制因素。在此示例中，可以将 36 个主机连接至光纤通道端口，或者将 8 个主机连接至 iSCSI 端口。

前端端口队列

前端端口队列保留正在处理或等待分派到存储处理器的 I/O。前端端口队列由拥有 I/O 端口的存储处理器管理。

每个端口上未处理的 I/O 数会直接影响其性能。源自多个主机 HBA 的前端端口 I/O 可能会很繁重，并具有很大的猝发性。为了消化这些 I/O，前端端口会变成排队设备。

每个到达的 I/O 都会占据一个队列位置。存储处理器从队列中移走 I/O 并进行处理。已经就绪可供使用的队列条目数称为 **端口队列深度**。光纤通道、FCoE 和 iSCSI 端口具有相同的队列深度。

如果 I/O 到达端口的速度超过存储处理器能够处理的速度，则会出现端口队列堵塞。如果超过队列深度，存储系统会向主机返回一个 *queue full* (QFULL) (队列已满) 的状态以响应 I/O。主机上 QFULL 的实际影响取决于其 O/S。但队列已满状况始终会对吞吐量产生负面影响。

存储处理器 (SP)

VNX 的 SP 是针对 I/O 吞吐量进行了优化的特殊用途处理器。它们需要管理 CPU、内存和 I/O 资源。

每个 VNX 有两个 SP。SP 为双冗余。这意味着，在其对等方发生故障时，其中一个 SP 可以接替其对等工作负载。

Operating Environment (O.E.) Block 是专门用于存储处理器的操作系统。

每个 VNX SP 都有一个 CPU。每个 CPU 有两个或更多 CPU 内核。与较低机型相比，较高机型的 VNX 有更多的更高速的内核。内核越多，存储系统能够同时执行的功能就越多。更高的内核速度决定了存储系统执行这些功能的速度有多快。

每个 SP 都有其自己的内存和 I/O 资源。与较低机型相比，较高机型的 VNX 具有更大的安装内存容量。此外，SP 与其对等 SP 共享内存资源。存储系统上最重要的内存使用是读/写缓存。缓存是存储系统用来缩短来自主机的读写请求响应时间的内存。

存储处理器 CPU

VNX 的 CPU 资源取决于机型。

所有 VNX 系列的 SP 都基于多处理器，因为 SP 具有多核 CPU。内核是 CPU 芯片上的处理器单元。对于基于多处理器（多 CPU）计算机的设备，这是一种有效的设计。将 CPU 内核群集在单个处理器芯片上可共享本地芯片总线和内存资源并缩短它们之间的物理距离，从而加速单独的 CPU 内核之间的通信速度。单一芯片还会减少电路板空间占用，并允许在电路板上整合 CPU 电源和冷却装置。

VNX 入门级和中端存储系统的内核要少一些，但始终至少有两个。企业级存储系统有更多的内核，至少有四个或更多。不同机型上的 CPU 有不同的时钟速度。CPU 上的所有内核都以相同的速度运行，但不在使用状态中的内核可以降低速度，以减少电源消耗和热量生成。CPU 的时钟速度越高，每单位时间内可执行的指令就越多。

需要在内核的数量和速度之间进行平衡。如果需要执行的并发功能比较少，则适合使用较少的但速度更快的内核。如果需要执行大量的并发功能，则使用更多但速度较慢的内核可以提高执行速度。与较高机型相比，入门级 VNX 系统的每个 CPU 的内核要少一些，因此它们的可扩展性比中端机型要小一些。

CPU 性能

SP 利用率是一项重要的 SP 性能指标。这是 SP 的 CPU 利用率衡量方法。

理想情况下，在对高可用性和一致性能有严格要求的生产环境中，SP CPU 的利用率应在 50% 左右。这样，如果 SP 故障触发 SP 故障切换，对等 SP 可轻松调整自己的处理负载和对等方的故障切换负载。

请注意，通过 Unisphere 报告的 SP 利用率是所有 CPU 内核利用率的平均值。它不反映多核 CPU 的单个内核利用率。

内存

每个 SP 都有自己的内存。所有 SP 内存都有错误修正码 (ECC)，ECC 执行错误检测和更正，以确保其内容的准确性。

不同 VNX 型号的 SP 具有不同的安装内存数量和速度。入门级存储系统拥有的内存比较高端机型的内存要少，速度也更慢。一般而言，内存容量越大，SP 的内存就越快，能够同时以较高水平的性能提供的功能也越多。

系统内存分为 SP 内存和缓存内存。写缓存内存从一个 SP 镜像到对等 SP。读缓存不会镜像到对等 SP。第 45 页的图 12 显示了存储系统内存的概念视图。

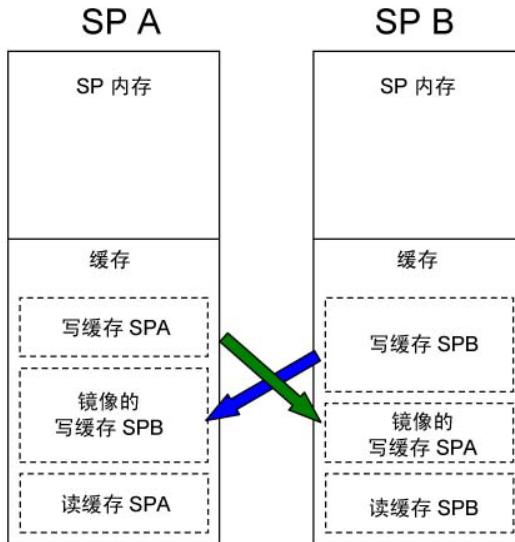


图 12 VNX 内存：概念视图

SP 内存

部分存储处理器内存专用于操作环境 (Block OE)、Unisphere GUI、功能和安装的应用程序。功能包括 FAST Cache、FAST VP、精简资源调配和 LUN 压缩。安装的应用程序包括复制应用程序，例如 SnapView、MirrorView 和 SAN Copy。

专用于 SP 内存的内存量取决于机型、功能的数量以及安装的应用程序。功能和安装的应用程序数量越多，分配给 SP 内存的可用系统内存就越大。

SP 缓存

存储系统的读/写缓存是一个连续的 SP 内存部分，它独立于 OE 用于系统软件和应用程序的内存部分。

VNX 上的可用缓存内存量取决于机型。缓存容量大小取决于系列机型，入门级机型的缓存内存量少于较高机型的缓存内存量。

缓存自身从逻辑上分为两个区域：读缓存和写缓存。读缓存用于内存中保留的、预计在以后的读取 I/O 中会被请求的数据。写缓存则存储正等待写入某驱动器的写请求数据。

写缓存分区可应用于两个 SP。可以单独为每个 SP 划分读缓存分区。所有缓存内存的大部分都分配给写缓存，剩余的小部分分配给读缓存。这是因为除了执行顺序读取 I/O 的工作负载以外，读取操作发现缓存中已存在其 I/O 的可能性比较小。避免发生写缓存容量不足的情况，这一点很重要。

可以对读缓存和写缓存的容量进行调节，以更好地满足特定工作负载或高峰需求。但是，读缓存和写缓存的总容量不能超过机型的最大总 SP 缓存容量。缓存的容量还取决于 SP 内存的容量以及所安装的可选软件应用程序。如果用户要重新分配读或写缓存内存，则必须禁用缓存功能。这会导致存储系统性能出现暂时下降，直到重新启用缓存。

除了 SP 自身的读和写缓存外，缓存内存还包含对等 SP 的写缓存镜像拷贝。这是一个重要的可用性功能。请注意，SP 上任何时候都可用作写缓存的内存量取决于其对等 SP 上的写缓存使用情况，分配给写缓存的缓存量是按系统来配置的，对两个 SP 都相同。

缓存页面

所有缓存都划分为页面。传入 I/O 放在缓存页面中。页面是缓存分配的最小单位。如果 I/O 小于缓存页面，则当 I/O 具有连续的 LBA 地址时，可能会让多个 I/O 共享同一个缓存页面。这是一种有效利用缓存页面的方式。随着 I/O 的执行（发送到驱动器并得到确认），其页面会释放给下一个 I/O 使用。其 I/O 尚未写入存储的写缓存页面称为 **脏页**。

读缓存操作

收到主机读请求时，存储系统会先检查其读和写缓存。如果数据不在任一缓存中，则必须从存储读取该数据。读取数据后，数据会保留在读缓存中供后续使用。

在缓存中找到其数据的读请求（**缓存命中**）消耗的存储系统资源最少；具有最高的吞吐量和最短的响应时间。但是，未在缓存中找到其数据的读取（**缓存未命中**）具有较长的响应时间，因为需要从驱动器中检索数据。

写缓存操作

存储系统的写缓存是镜像的回写缓存。这意味着：对于每一个写操作，其数据都会存储在缓存中并拷贝到对等 SP，然后向主机确认请求。在此过程中，写缓存会在存储系统的 SP 之间进行复制（**镜像**），以通过冗余确保数据保护。此外，还会等到请求获得确认后才将其写入磁盘。从写缓存确认请求可使写入响应时间不受磁盘速度和 RAID 作用影响，这样可以提高性能。

镜像写缓存的可用性功能会对性能产生轻微的影响。镜像操作完成前，会对主机 I/O 请求的确认带来延迟。SP 用于写缓存的可用内存也会减少，因为必须维护一份其对等方的缓存拷贝。在带宽密集型工作负载中，镜像过程可能会限制总的存储系统写入带宽。

后端端口

VNX 系列采用 6 Gb/s 串行连接的 SCSI (SAS) 后端。SAS 协议是对用于传统 EMC CLARiiON 中端存储系统后端的光纤通道协议的演变。

VNX 的 SAS 扩展器和链路构成了后端。从技术上而言，后端是一个包含两类 SAS 扩展器的分层结构：扇出和边缘。VNX 将 SAS 扩展器硬件用作交换机，以简化存储系统的配置，使其能够在最小延迟的情况下根据驱动器进行扩展，同时仍为不断增加的工作负载提供相同的带宽。

第 47 页的图 13 显示了 VNX 后端的高级方块图。此图是一个概念图，而不是全面的内容介绍，也不是为了展示任何特定的 VNX 存储系统型号。

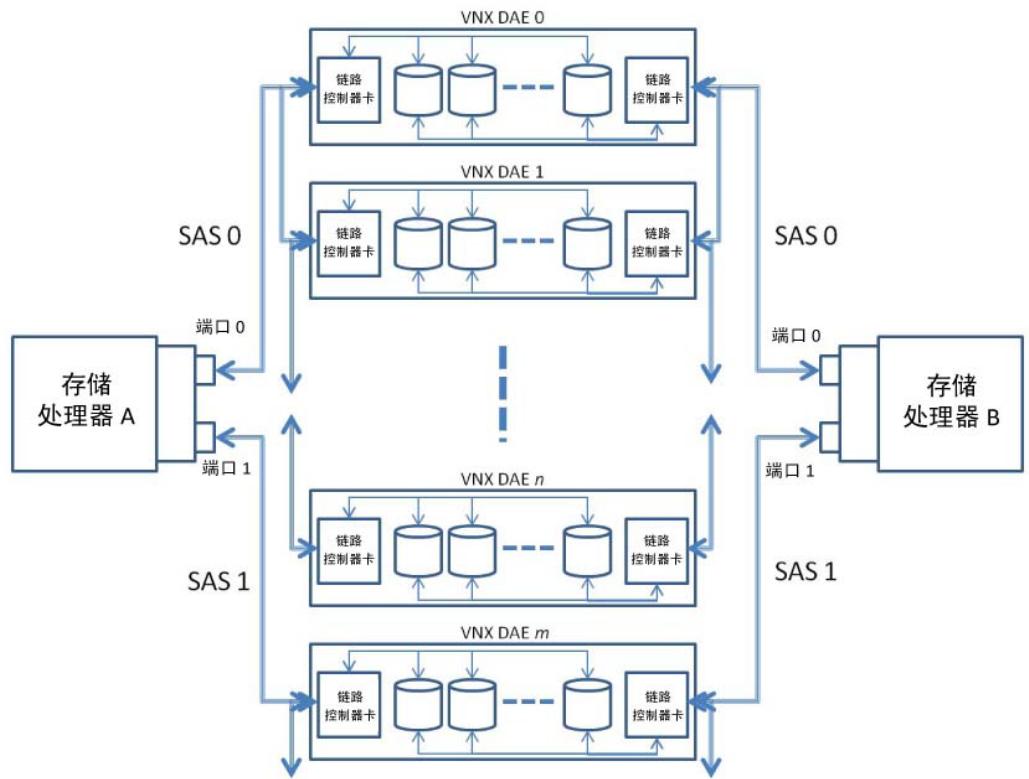


图 13 VNX 后端概念图

每个端口的 VNX 存储模块（DPE、SPE 和 DAE）的最大数量（图 13 中显示为 n ）为 10。存储模块可以承载的每个端口的最大驱动器数量为 250。

不支持添加所包含的驱动器插槽超过总线或系统型号最大值的存储模块。

例如，如果在总线上已经有 240 个驱动器的情况下再添加一个存储模块，则不会将该存储模块联机。

也称为“总线”和“环路”

LCC 的菊花链有时称为总线。LCC 与存储设备之间的连接有时称为环路。这是传统的 EMC CLARiiON 的术语。请注意，SAS 协议规范中不使用总线和环路术语。Unisphere 维持了传统的总线表示法。

后端 SAS 端口

VNX 系列的每个存储处理器有 1 到 4 个扇出扩展器，具体取决于机型。一个扇出扩展器有两个宽端口（0 和 10），如第 48 页的图 14 所示。端口支持到设备的链路。设备通常是一个 SAS 驱动器，但也可以是另一个扩展器。宽端口支持多条 SAS 链路（有时称为通道）。

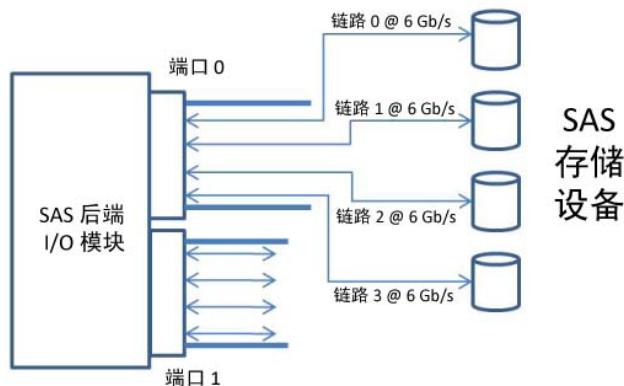


图 14 VNX SAS 后端端口概念图

每个 VNX 端口包含四条链路。每条链路的能力为 6 Gb/s，换算成每个端口的额定值即为 2400 MB/s。但是，考虑到仲裁延迟和组帧，每个聚合端口大约有 2200 MB/s。请注意，实际的链路带宽取决于链路末端的存储设备。扩展器可聚合带宽并平衡端口内链路的流量。提供的性能取决于工作负载和存储处理器型号。这种体系结构可在扩展器之间产生 2.4 GB/s 的流量。在具有理想工作负载的链路上，可以达到大约 750 MB/s。

存储系统内可同时进行寻址的驱动器数量受整合到扇出扩展器中的物理端口内的链路数量的限制。

例如，如果有两个宽端口，每个端口有四条通道，一个存储处理器最多可同时向八个驱动器发布命令。

请注意，VNX 后端端口的带宽比传统的 EMC CLARIION 总线的带宽高两倍以上。此外，后端的并行度也有所增加。较旧技术的 CLARIION 总线一次只能对一个驱动器进行寻址；VNX 后端可同时对四个驱动器进行寻址。

宽端口还可提供冗余。损失任何一个端口通道都会使性能降级，但它不会导致整个宽端口发生故障。四个通道中，只需有一个保持连续运行即可。

在 VNX 中，每个扇出扩展器的端口都连接至单独的边缘扩展器。每个磁盘阵列存储模块 (DAE) 中都有一个边缘扩展器。实际上，DAE 中有两个边缘扩展器，每个 SP 连接一个。请参见第 47 页的图 13。这些边缘扩展器称为链路连接器控制器 (LCC)。边缘扩展器直接连接至 DAE 中的 SAS 存储设备和一个其他扩展器。与另一个边缘扩展器的连接会创建一个菊花链式边缘扩展器/DAE。

其带宽利用率会对端口性能产生影响。随着后端端口接近其全部传输容量，我们称之为饱和。高带宽工作负载更容易使后端端口达到饱和，但高吞吐量工作负载却不容易使后端端口达到饱和。添加更多负载不会进行线性扩展。结果会增加驱动器访问端口的响应时间。

FAST Cache

VNX 系列支持可选的性能增强功能 FAST Cache。FAST Cache 是一个闪存驱动器池，配置用作次 I/O 缓存。

FAST Cache 概述

FAST Cache 可减少读写 I/O 的主机响应时间。

FAST Cache 是硬件和软件的组合，可作为一项可选功能安装。该功能的软件用于监视地址块的存储处理器 I/O 活动，这些地址块在指定的时间间隔内从存储中读取或写入了多次。当软件检测到此类 I/O 活动时，会将这些地址块提升到 FAST Cache。提升地址块后，FAST Cache 会处理去往或来自该地址块的 I/O。

FAST Cache 调配了闪存驱动器。闪存驱动器的响应时间非常短。（请参见第 52 页上的“物理”。）FAST Cache 的闪存驱动器位于 DAE 或 DPE 中。但是，从理论上而言，由于该功能的软件作用，它们位于通常被视为海量存储的驱动器的“前方”。调配 FAST Cache 时，需全部使用闪存驱动器，并成对分配。

当地址块的活动减少时，可能会逐出该地址块，以在缓存内腾出空间供高使用率的地址块使用。在这种情况下，FAST Cache 会将其缓存的写入发送到主存储（通常是机械硬盘驱动器）。

FAST Cache 会减少到后端的读和写活动。FAST Cache 可提高虚拟池中精简 LUN 的性能。它还允许存储处理器的写缓存更快地进行刷新，因为它刷新到的是高速闪存驱动器。这使得主读/写缓存可以吸收更大量的非 FAST Cache 写入 I/O。系统整体性能得到了提高。

FAST Cache 可展示整个存储系统的优点，并加速那些具有高访问局部性特征的工作负载。但是，FAST Cache 资源必须进行管理。FAST Cache 的使用应限制在能从其使用中获取最大好处的 LUN。

FAST Cache 的理想 I/O 是随机 I/O，尤其是具有高读取百分比的 I/O。避免针对具有小型数据块顺序 I/O 的 LUN 使用 FAST Cache。日志是接收小型数据块顺序 I/O 的 LUN 的典型示例。

FAST Cache 的容量取决于机型，可从 100 GB 到 2 TB。但应用程序数据集可能会有非常大的容量并且有非常广的局部性。请注意不要配置太小的 FAST Cache。

例如，一个非常大的数据库可能包含一个 2 PB 的数据集。仅 5% 的低局部性可能会超过最大的 FAST Cache 的容量。

FAST Cache 减少了机械驱动器上的负载，因此可提高特定存储系统的总体性能，这些存储系统的工作负载有很好的访问局部性和兼容的 I/O 特征，并且已针对 FAST Cache 适当调整了大小。

存储

存储是机械硬盘驱动器和基于内存的闪存驱动器，以及这些驱动器的宿主存储模块。

存储模块

UltraPoint™ DAE 是硬件存储模块，为存储系统驱动器提供物理保护、组织和连接。类似地，DPE 提供相同的驱动器承载功能，但它在存储模块中包括存储处理器硬件。（请参见第 37 页的“磁盘”。）

VNX 系列中使用的 UltraPoint DAE 具有多种不同的驱动器插槽容量和驱动器外型。插槽是 DAE 中的开口，驱动器通过插槽插入 DAE。

双端口驱动器

所有驱动器都具有双端口。每个驱动器有两个独立的 SAS 端口，用于连接到不同的后端端口。每个驱动器端口连接到其 DAE 中不同的 LCC，LCC 又反过来通过单独的后端端口连接到 SP。（请参见第 48 页的图 14。）

驱动器和后端端口带宽

与机械硬盘驱动器的最大外部传输速度相比，后端总线具有更多可用带宽。

例如，6 Gb/s SAS 后端总线的带宽超过 2100 MB/s。典型的 15K rpm 机械 SAS 硬盘驱动器具有 70 到 100 MB/s 之间的持续传输速率（猝发率要高很多，但仍不能使链路达到饱和。）

不过，可同时对多个驱动器进行寻址。总线和驱动器之间的可用带宽差异可通过 SP 的排队机制（针对发送到驱动器的 I/O 命令和数据）进行调解。

每个驱动器都有两个 I/O 队列，每个存储处理器一个，也就是每个驱动器端口一个。队列由 SP 通过软件进行维护。SP 将命令和数据发送到驱动器，命令被添加到队列中。驱动器执行命令并进行确认。当命令得到确认后，即从队列中将其移除。对每个 SP 的每个驱动器只能发送限定数量的命令。该数量由可能的队列条目数量设置。系统会对可能的队列条目的数量进行限制，以确保最优性能，即使驱动器能够处理额外的命令也一样，但会有响应时间罚分。会一直发送命令，直到 SP 的驱动器队列填满。如果驱动器队列已满，存储处理器会保留命令，直到排队的命令完成。存储处理器会实施序列化规则，通过发布命令来防止驱动器的内部 I/O 执行出现排序问题。

文件逻辑设备

除了前面讨论的数据块存储，文件存储的逻辑体系结构包括 X-Blade 和控制台。

X-Blade

X-Blade 是独立的文件服务器，用于将请求的文件传输到客户端计算机。

Operating Environment File 是专用于 X-Blade 的操作系统。该操作系统进行了优化，可在 NAS 和基本 VNX 存储系统之间执行文件操作。它支持 NFS 和 CIFS 协议。

X-Blade 以冗余对的形式安装，其中一个为活动主 X-Blade，另一个为备用 X-Blade。当主 X-Blade 发生故障时，备用 X-Blade 变为活动 X-Blade。

控制台

控制台是管理 NAS 环境的专用计算机。管理包括：安装、资源配置和监视 X-Blade 和文件环境。

第 8 章 存储对象

本章包括以下主题：

物理存储对象	52
逻辑存储对象	59
物理资源配置示例	80

VNX 系列包含两类存储对象：

- ◆ **物理存储对象** — 机械硬盘驱动器和闪存驱动器等驱动器。
- ◆ **逻辑存储对象** — 存储池、RAID 组、LUN 和 metaLUN。

物理存储对象

机械驱动器和基于半导体的驱动器都用于永久性存储数据。主机读取物理存储对象以进行检索，写入物理存储对象以进行创建、修改或删除。

基本的机械硬盘驱动器术语

硬盘驱动器是一种计算机化设备。基于微处理器的**磁盘控制器**控制驱动器的操作。除此之外，驱动器通过定位读/写磁头来响应 I/O 请求，执行性能优化、数据保护功能和诊断。

硬盘驱动器包含一个或多个有磁性涂层的、旋转的**盘片**。盘片通过**磁盘轴**连接在一起，驱动器轴由驱动器马达转动。有时硬盘驱动器也称为**磁盘轴**。通过读/写磁头对盘片进行数据读写操作。驱动器有多少盘片，就有多少读/写磁头。读/写磁头安装在一个**传动臂**上，传动臂将读/写磁头放在要读取或写入的盘片位置上。传动臂的移动称为**寻道**。**全程寻道**是从最外侧的磁道移动到靠近磁盘轴的最内侧磁道。

磁道是环绕盘片轴的一个类似同心环的区域，数据存储在磁道上。磁道分为**扇区**。扇区是可单独寻址的最小数据单元。有时扇区也称为**数据块**。一个驱动器扇区通常包含 512 字节的数据。许多 VNX 配置和调整选项都以**数据块大小**为单位，即 512 字节。VNX 驱动器扇区总共包含 512 字节的数据和 8 个额外的元数据字节，总共 520 个字节。这些元数据是存储系统的专有数据完整性功能。**逻辑块寻址 (LBA)** 是一种映射技术，用于独立于扇区在驱动器盘片上的物理磁道/扇区位置对扇区进行寻址。最低编号的 LBA 位于驱动器的最外侧磁道。LBA 随着与最内侧磁道的靠近而增加。

硬盘驱动器包含使用特定协议的物理连接。连接通过前面提到的端口进行。所有 VNX 驱动器都具有双端口。双端口可提供冗余连接。VNX 仅支持使用 SAS 协议的硬盘驱动器。

基本闪存驱动器术语

与机械硬盘驱动器相比，闪存驱动器的结构更简单。闪存驱动器旨在模拟机械硬盘驱动器，以与基于计算机的设备兼容。

从内部而言，闪存驱动器的硬件体系结构可以分为以下主要区域：

- ◆ **连接**：向闪存驱动器提供电源和数据
- ◆ **控制器**：管理驱动器功能的微型计算机
- ◆ **海量存储**：存储数据的非易失性内存

连接

与机械硬盘驱动器一样，连接为闪存驱动器提供电源和数据。使用 6 Gb/s SAS 连接，可实现 768 MB/s。

控制器

控制器由若干组件组成：包括微控制器、RAM 和非易失性内存 (NVRAM)。微控制器管理所有驱动器功能。闪存驱动器是基于计算机的设备，理解这一点很重要。控制器通常是一个专用设备，其设计和结构符合规范，通过半导体制造商进行许可。控制器可以包含一个或多个嵌入式处理器内核。在双内核配置中，一个内核通常负责处理 I/O，而另一个负责管理存储。闪存驱动器的控制器通过软件实施许多驱动器功能。这是与机械硬盘驱动器的一个重要区别。使用闪存驱动器，只要控制器的 CPU 足够快，则对闪存驱动器性能影响最大的是其软件的体系结构。此外，可以轻松地更改此软件，以添加功能（例如机载加密）或提高性能。

RAM 用于管理从作为缓存的闪存驱动器读取和写入的数据，并用于闪存驱动器运行程序和数据。闪存驱动器上的典型 RAM 大小从 32 MB 到 256 MB 不等。控制器缓存可增强闪存驱动器的总体性能。闪存的写入速度比读取速度慢。缓存使用快速动态 RAM (DRAM) 来最大程度减少写入数量并改善较慢的驱动器闪存的响应时间。写合并是缓存中采用的技术之一。此过程将写 I/O 组合在一起并以单个内部操作的形式写入它们，这与拆分成多个较小的写入操作正好相反。除缓存之外，RAM 还包含映射表和访问历史记录。映射表将闪存驱动器的内部数据结构与主机文件系统的数据结构相关联。访问历史数据用于通过负载平衡延长驱动器的寿命。负载平衡是一种内部日常管理过程，它在驱动器的所有内存位置平均分布驱动器的写入数量。此过程可避免过度使用某些位置，从而延长内存寿命。

闪存驱动器缓存内存为非易失性内存。如果驱动器掉电，驱动器缓存内容将会保留并且驱动器内容不会被损坏。目前，用来保护闪存驱动器缓存的非易失性 RAM 解决方案为电池备份和超级电容器。在出现电源故障时，这两种技术均可提供足够的电力，以便有足够的时间将缓存内容保存到驱动器的非易失性存储。

海量存储

NVRAM 用于在断电时存储闪存驱动器的操作软件和数据。并非所有闪存驱动器都有针对其操作软件的单独 NVRAM。有些型号将其程序和数据存储到驱动器的海量存储中。NVRAM 由内部通道寻址。通道将可寻址存储分为可独立寻址的存储数据块。闪存驱动器的通道越多，其并发性越高。通道数量和可由通道寻址的容量取决于驱动器型号。

管理闪存存储

从外部而言，闪存驱动器的最小可寻址数据单位为扇区。闪存驱动器会模拟 LBA 以确保兼容性。但是，从内部而言，最小的可写入数据单位为页面。一个页面包含多个扇区：8 个扇区 (4 KB)、16 个扇区 (8 KB) 或 32 个扇区 (16 KB)，具体取决于驱动器型号。闪存驱动器的理想 I/O 数据块大小为其原始页面大小。可以直接向空的、未使用的页面进行写入。最后一点，在写入到闪存驱动器内存之前，必须先对其进行擦除。一次擦除的内存单位为数据块，数据块包含多个页面。

首选操作是一次写入整个数据块。写入数据块之前，必须对数据块内的所有页面进行寻址。此外，只能向已经擦除的数据块执行写入操作。具有大量空数据块的驱动器的性能非常高。驱动器的总容量中有一部分被保留用于此目的，用户无法访问这一部分。（该容量不是驱动器额定容量的一部分。）虽然存在空数据块，每次写入时还是会使用新的数据块页面。当驱动器所有的空数据块都用尽时，其性能会下降。这称为“重写性能损失”或“驱动器老化”。即使某个驱动器表面上显示有可用的存储容量，也有可能已经没有空的数据块 — 所有数据块都具有有效数据和无效页面（由于删除或覆盖）的组合。

一旦驱动器没有空的数据块，便会重新使用标记为无效的页面。无效页面与使用中页面分隔开，以便仅使用使用中页面创建数据块，使用包含可用于写入的页面创建新且空的数据块。包含无效页面的数据块将被擦除，并成为所有新写入的目标。这种分隔包含读取-修改-写入过程。

要创建驱动器内部的新的空数据块，将读取一个或多个数据块的内容，使用中页面将被打包到完全包含使用中页面的数据块。将使用中页面打包在一起允许从标记为无效的剩余页面创建空数据块。将对空数据块进行擦除并用于所有新写入。这些额外步骤由驱动器控制器执行，需要一定的时间。由于需要执行数据块重写，因此到驱动器的每个写入请求都变慢了。读取可能会因为下列原因而推迟：先前排队的、要求重写的写入；等待整合时被锁定的数据块。创建新数据块的操作是连续的。在需要时发生并在驱动器闲置时进行。

硬盘驱动器分类

硬盘驱动器根据下列项有广泛分类：

- ◆ 容量
- ◆ 速度
- ◆ 缓冲区大小
- ◆ 外型
- ◆ 类型

例如，驱动器可以称为 600 GB、15,000 rpm、64 MB 缓冲、3.5 英寸或 SAS 硬盘驱动器。

闪存类型的驱动器的很多分类都与机械硬盘驱动器相同，但也存在例外。例如，闪存驱动器没有与寻道相关联的延迟。

容量

人们经常用容量来区分硬盘驱动器。容量通常指存储的额定 GB 或 TB 数。例如 600 GB 或 2 TB。

机械硬盘驱动器的原始容量由以下因素决定：盘片的直径、驱动器内的盘片数量以及盘片的面密度。

闪存驱动器的原始容量由驱动器中安装的用户非易失性内存量决定。闪存驱动器的实际容量比报告的要大。多出的这一部分容量供内部日常管理使用，例如前面介绍的数据块管理。保留容量越大，闪存驱动器在持续写入工作负载下的性能越好。保留容量因闪存驱动器而异，具体取决于制造商和型号。这种非易失性机载 RAM 通常是报告的原始容量的 20% 到 10%。

面密度（仅限机械驱动器）

面密度是可以存储在机械硬盘驱动器盘片面积上的位数。如果面密度高，可以存储在单个盘片上的容量就大。面密度以 Gb/in² 或 Gb/cm² 为测量单位。较高的面密度大约为每平方英寸 625 Gb。采用最新的技术，可以在单个 3.5 英寸的盘片上存储高达 1 TB 的数据。

有必要增加面密度，以增加驱动器容量和性能，同时保持外型（3.5 或 2.5 英寸）和降低驱动器复杂程度、节省电能消耗以及每 GB 的成本。面密度越高，读/写磁头越灵敏，达到指定容量所需的盘片就越少。盘片减少又可以反过来减少传动臂和驱动器磁头的数量。这使驱动器从机械上变得更加简单。同时，它还允许驱动器磁头在更短的盘片磁道物理距离内访问更多的数据，从而有助于提高性能。

集成扩展

集成扩展是增加可在闪存驱动器所使用的 NVRAM 芯片上存储的位数。随着半导体处理技术的进步，生产的芯片在相同的面积内具有更大的内存容量。这使得可以相同或更小的外型提供容量更大的闪存驱动器。此外，随着整合度提高，通常 NVRAM 芯片的性能也会提高。这是因为访问单个页面所需的时间减少了。

为什么容量报价值会有不同

遗憾的是，存在多种方式的存储容量报价。它可能是：

- ◆ 十进制或二进制，即十进制（基数为 10）或二进制（基数为 2）。本节将对此进行阐述。
- ◆ 磁盘格式化之后的容量；这是磁盘的可用容量或用户数据容量。
- ◆ 磁盘格式化之前的容量；这是磁盘的额定容量或原始容量。对磁盘进行格式化会使用部分容量。这是购买驱动器时的报价容量。

新用户常常想知道为什么他们购买了 1 TB 的驱动器，但主机 O/S 却报告其格式化后的容量为 931 GB。原因是硬盘驱动器制造商以十进制 GB 报告容量，而主机 O/S 以二进制 GB 报告容量。

二进制与十进制容量

一个字节始终包含八个位。但是，千字节可以是十进制（1000 字节）或二进制（1024 字节），具体取决于报告来源。其容量测量方式的差异会随着测量单位的增加而加大。第 55 页的表 3 显示了常见容量测量方式之间的差异。

表 3 十进制与二进制容量

测量单位	二进制字节	十进制字节	百分比差异
千字节	1024	1000	2
兆字节	1,048,576	1,000,000	5
十亿字节	1,073,741,824	1,000,000,000	8
万亿	1,099,511,627,776	1,000,000,000,000	10

请注意，在 TB 容量中，有 10% 的值差异。使用二进制 GB 来测量其可用容量的计算机 O/S，其显示的硬盘驱动器的容量远低于制造商标注的容量（以十进制 GB 标注）。

Unisphere 存储管理应用程序会同时以二进制 GB 和十进制字节来报告磁盘容量，以避免造成困惑。

驱动器格式化和 OE 专用空间

对驱动器进行格式化时，将永久设定其扇区大小。传统标准为每个扇区 512 字节。VNX 存储系统上的驱动器不使用传统扇区划分。每个扇区多了八个字节，用于存储扇区级别的数据保护信息。这些额外的字节用于验证和维持数据的完整性。这种 520 字节的驱动器扇区划分方法使硬盘驱动器的可用容量减少了不到两个百分点。但这一容量上的轻微减少对可用性带来了很大的正面影响。此外，VNX 上的每个驱动器都保留了固定的空间量用于存储 VNX 系统信息。

主机 LUN 格式化容量

分区和格式化会减少 LUN 的可用容量。（请参见第 67 页的“逻辑单元 (LUN)”。）这种容量上的差异是元数据带来的。元数据是关于数据的数据。元数据用于支持可用性功能，例如数据完整性、恢复和信息安全（保密）。它还包括文件系统信息。

操作环境 (OE) 会保留少量磁盘空间用于跟踪表和日志，这会占用每个驱动器大约 64 MB 的容量。

不同的文件系统有不同的元数据量开销。例如，Linux 文件系统（例如 ext2 和 ext3）的元数据量比 Microsoft 的 NTFS 元数据量要小。

原始 LUN 存储

分区和格式化 LUN 会在向驱动器写入任何用户数据之前便减少驱动器的可用容量。但有些应用程序不需要格式化驱动器。原始设备是没有文件系统的 LUN，因此它没有文件系统开销带来的容量损失。有些应用程序（例如 Oracle）负责管理原始设备上的数据，这不仅避免了文件系统开销，而且还提高了事务处理吞吐量。

速度

转速将硬盘驱动器分为不同的性能等级。转速按每分钟的转动次数 (rpm) 来衡量。最常见的硬盘驱动器转速包括：

- ◆ 15,000 rpm
- ◆ 10,000 rpm
- ◆ 7,200 rpm
- ◆ 5,400 rpm

磁盘驱动器的转速（例如 15,000 rpm）取决于每个磁盘驱动器的型号。转速是不会改变的。同一存储模块或后端端口的其他驱动器的速度不会影响驱动器的速度。

转速可以使用 K 表示法来缩写。例如，一个 15,000 rpm 的硬盘驱动器可以描述为 15k rpm。

请注意，转速不适用于闪存类型的驱动器。

缓冲区大小

驱动器通常会有一个机载缓冲区，驱动器的微控制器用它来优化驱动器操作。缓冲区是一个托管内存空间，可实现比磁盘访问更快的请求响应时间。有时，缓冲区也称为 **磁盘缓存**。

驱动器的 **缓冲速度** 是驱动器通过其缓冲区的 RAM 实现的传输速度。该速度不受驱动器的机械操作的影响，也不受闪存驱动器的 NVRAM 操作（与缓冲区 RAM 相比较慢）的影响。

典型的缓冲区大小包括：16 MB、32 MB 和 64 MB，容量越大，可以提供的驱动器性能越高。

该缓冲区的优势在于读取缓冲。读取缓冲由驱动器独立执行。驱动器可自行将读取预取到缓冲区。在存储处理器级别的读取未命中在硬盘驱动器的缓冲区中可能是读取命中。这可以改善总体性能，尤其是在驱动器上顺序排列的文件节之间存在缺口的文件系统中。

除了闪存驱动器和特定的 ATA 重建方案，VNX 中的驱动器不使用其独立缓冲区进行写入或写入缓存。这是出于数据完整性考虑。机械驱动器的缓冲区没有电池备份。如果发生彻底的电源故障或移除驱动器，缓冲区的数据内容可能会丢失。

外型

外型描述 IT 行业一组标准化的驱动器尺寸。机械硬盘驱动器的 **外型** 由其盘片大小决定。外型是指硬盘驱动器盘片的直径，以英寸为单位。市面上有 3.5、2.5 和 1.8 英寸外型的驱动器。2.5 英寸外型有时称为 **笔记本电脑格式**。

为了实现物理兼容性，闪存驱动器具有与物理硬盘驱动器相同的外型。

VNX 存储系统既支持 2.5 英寸外型的驱动器，也支持 3.5 英寸外型的驱动器。

类型

VNX 系列支持以下类型的驱动器：

- ◆ SAS
- ◆ NL-SAS
- ◆ 闪存

这些驱动器具有不同的特征，使它们更适用或更不适用于不同的工作负载。

SAS 硬盘驱动器

在所有 VNX 机械驱动器中，串行连接的 SCSI (SAS) 硬盘驱动器具有最高的性能和可用性。目前，SAS 硬盘驱动器使用 6 Gb/s SAS 协议连接。这种连接理论上具有 600 MB/s 的最大带宽。SAS 硬盘驱动器是可靠的企业级硬盘驱动器。

最高性能的 SAS 硬盘驱动器为 15k rpm。此外，VNX 还支持更经济的、性能适中的 10k rpm 驱动器。

近线 SAS (NL-SAS) 硬盘驱动器

NL-SAS 硬盘驱动器凭借其适中性能，成为 VNX 上具有容量优势的最经济实惠的存储解决方案。

NL-SAS 驱动器使用 6 Gb/s SAS 标准连接。这种连接具有大约 600 MB/s 的最大猝发速度。顾名思义，这些硬盘驱动器是近线可靠性级别的驱动器。与企业级存储相比，近线存储的 MTBF 要低一些。

NL-SAS 驱动器为 7.2k rpm。这些驱动器在 VNX 系列上具有最大的可用容量。对于需要大量存储的工作负载，它们是非常经济的驱动器。

闪存驱动器

闪存驱动器是基于半导体的驱动器。它们没有移动部件，具有非常高的性能和可靠性。

与机械硬盘驱动器相比，闪存驱动器的平均服务时间等衡量指标特别低。例如，闪存驱动器可在大约 0.1 ms 的时间内执行一次小型数据块随机读取。相比而言，最快的 SAS 机械硬盘驱动器也需要 4 ms 的时间。闪存驱动器的读取速度要快 40 倍。

出于兼容性考虑，闪存驱动器模拟操作环境的机械硬盘驱动器。闪存驱动器存储使用 SAS 协议直接连接到后端端口。闪存驱动器具有企业级可靠性。

采用目前的闪存技术，读 I/O 要明显快于写 I/O。随机读取的存储系统读缓存未命中数量相当可观。这使得闪存驱动器成为具有高随机读取百分比的工作负载的理想选择。随着 I/O 数据块大小的增加以及顺序 I/O 的使用，闪存驱动器的性能会下降。此外，在基于闪存的 RAID 组（由两个存储处理器分别拥有）中设置多个 LUN 是一种利用闪存驱动器并发优势的有效方法。闪存驱动器还可以用作短寻道机械驱动器的替代品。

除了具有高性能之外，闪存驱动器消耗的电能也比机械硬盘驱动器消耗的电能少。

特殊注意事项

包括 VNX 的闪存驱动器在内的所有 SSD（无论制造商是谁）的性能都会因为状态和负载情况而发生变化。

尽管闪存驱动器模拟机械硬盘驱动器，但其内部操作具有很大的不同。从内部而言，它们是随机访问存储设备。对于机械硬盘驱动器很常见的操作有可能会通过在闪存驱动器上实施线性寻址而降低闪存驱动器的性能。要避免的操作包括：

- ◆ 文件系统碎片整理
- ◆ Linux 介质格式化和检查
- ◆ Linux

磁盘省电（驱动器降速）

VNX 支持一种称为驱动器降速的省电功能。驱动器降速允许将 RAID 组的机械硬盘驱动器配置为在非使用状态时进入省电状态。这种状态称为待机。当 RAID 组在预定义的时间内未接收 I/O 时，即进入待机状态。系统会定期对待机状态的驱动器进行测试，以确保它们的就绪性。向包含处于待机状态的驱动器的 LUN 发出主机 I/O 请求时，存储系统在将 RAID 组的驱动器返回就绪状态时会暂挂 I/O，之后会处理读或写请求。

此外，可以将未配置到 RAID 组中的驱动器（例如热备盘和未绑定的驱动器）配置为在需要它们之前一直保持待机状态。请注意，只有经 EMC 认证可专门用于驱动器降速的驱动器才可以进行驱动器降速。Unisphere 会从存储系统的已调配驱动器中检测可用于驱动器降速的候选驱动器并将其显示给用户。虚拟池中的驱动器和存储系统的系统驱动器不能进行驱动器降速操作。

逻辑存储对象

资源调配术语用来描述以下过程：对存储系统进行逻辑配置，以满足其工作负载对有效资源利用、容量、性能和数据安全的需求。逻辑存储对象包括 RAID 组、LUN、metaLUN 和虚拟池。

RAID 组

将驱动器整合到具有单一 RAID 级别的相关组中，称为 RAID 组（第 59 页的图 15）。RAID 组中的驱动器数量及其 RAID 级别决定了该 RAID 组的可用性、容量和性能。理解 RAID 组的功能是存储系统的一个重要的基础概念。

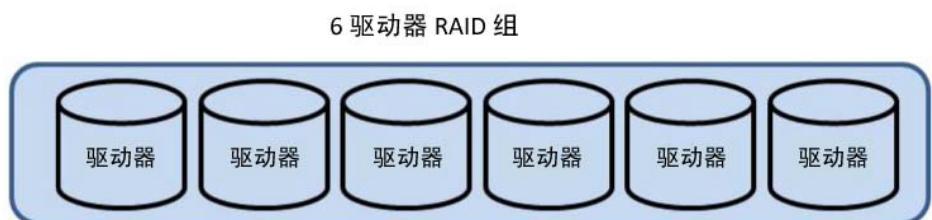


图 15 RAID 组概念图

RAID

独立磁盘冗余阵列 (RAID) 是性能和可用性方面的关键技术。需要了解 RAID 技术，才能确定 RAID 组中需要使用多少个磁盘；了解它们能够提供多大的容量；并评估它们在正常运行和降级状态下的性能。

RAID 级别通过冗余提供数据保护。请不要将冗余与数据完整性相混淆。虽然二者都与可用性相关，但它们是不同的。

存储冗余是存储系统在发生硬件故障后继续提供数据访问的能力。冗余通过以下功能实现：复制驱动器、电源、对驱动器的多个后端端口访问以及 RAID 保护方案。

冗余不是数据备份或时间点数据复制的代替。错误数据、用户损坏的数据和数据删除都会以冗余形式存储在驱动器上。在驱动器发生故障时，这些“坏数据”仍可用（如果进行了数据删除，则不可用）。

数据完整性是维护驱动器上的、写入驱动器的或从驱动器读取的数据的准确性。通过存储系统的自动错误检测和更正机制处理存储系统上的数据完整性（请参见第 107 页的第 11 章）。驱动器上可能会发生未检测到的数据损坏情况；例如驱动器盘片上的介质错误。在 VNX 的驱动器中，数据完整性通过驱动器的 520 位扇区划分方法进行维护。每个扇区的额外容量被分配给元数据，元数据用于错误检测以及对许多扇区级别的数据错误进行更正。错误检测和更正均由 SNiFFER 服务在数据访问时作为连续的后台进程执行。

请务必注意，不同存储系统供应商的 RAID 实施可能存在不同。存储系统管理员或架构师不应根据使用另一家供应商的存储系统的经验来假定 RAID 级别的操作特征。

RAID 级别

VNX 系列支持六种 RAID 级别。每个 RAID 级别具有不同的容量、性能和可用性特征。镜像和奇偶校验是两种广泛使用的 RAID 类型。条带技术适用于这两种 RAID 类型。将分别讨论分条的非冗余 RAID 类型 (RAID 0) 和单个磁盘存储类型。

镜像 RAID

当随机写入性能和可用性比容量成本更重要时，采用镜像类型的 RAID (第 60 页的图 16)。镜像 RAID 在两个或更多的驱动器上创建数据的完整拷贝，以保护数据。主 (驱动器) 具有数据的原始拷贝，而镜像 (驱动器) 具有数据的完整拷贝。如果 RAID 组发生一个或两个驱动器受损的情况，数据可以得到保护。

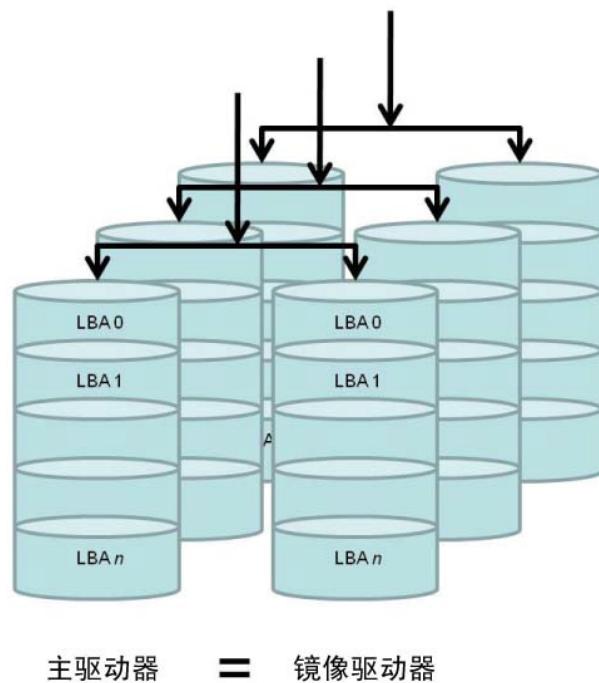


图 16 镜像 RAID 概念图

奇偶校验 RAID

奇偶校验是一种基于算法的数据保护技术，该技术会存储冗余数据以便检测错误并进行可能的修正。奇偶校验元数据可能分布于所有驱动器上，也可能分布于单个驱动器上。（请参见第 61 页的图 17。）所有奇偶校验 RAID 类型都会进行分条。奇偶校验 RAID 在三个或更多磁盘上对数据进行分条。如果 RAID 发生一个或两个驱动器受损的情况，数据可以得到保护，具体取决于 RAID 级别。

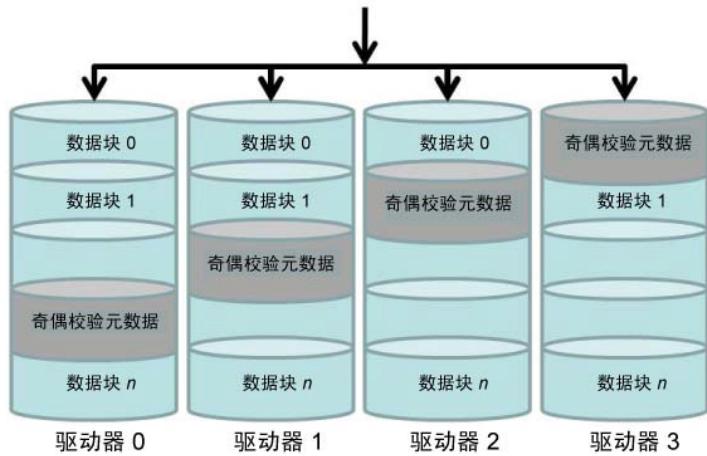


图 17 奇偶校验 RAID 概念图

分条 RAID

分条 RAID 是一种提供多个驱动器的容量和性能优势的 RAID 级别，没有数据保护功能。

分条

分条是将数据分布在多个驱动器上以提高性能并增加可用容量的方法。
(请参见第 61 页的图 18。)

带宽和吞吐量都可以通过分条得到提升。这是因为可以对 RAID 组中的磁盘并行执行单个大型或多个小型 I/O。并行指多个磁盘同时为主机的 I/O 请求提供服务。条带中的驱动器越多，并行度越高，传输速度更快。

同样，通过在多个磁盘上分条，容量也得到增加。添加到条带中的每个额外驱动器都会增加总的可用存储容量。这对于要求的容量比单个驱动器所能提供的容量多的数据集也非常有用。

分条带来的性能和容量优势取决于条带中所有驱动器的功能。驱动器性能越高，条带的性能越高。驱动器容量越高，提供的条带数量越多。

跨条带的数据分布会在驱动器之间产生相互依赖。这种相互依赖性会降低可靠性。如果条带中的一个驱动器发生永久数据丢失，则所有设备上的所有数据都会丢失。奇偶校验和镜像 RAID 级别中的数据冗余减小了这种可能性。但是，奇偶校验会增加 RAID 级别的复杂性，在某些情况下还会降低其性能。一般来说，相对于镜像级别而言，条带化 RAID 级别更容易导致数据丢失。

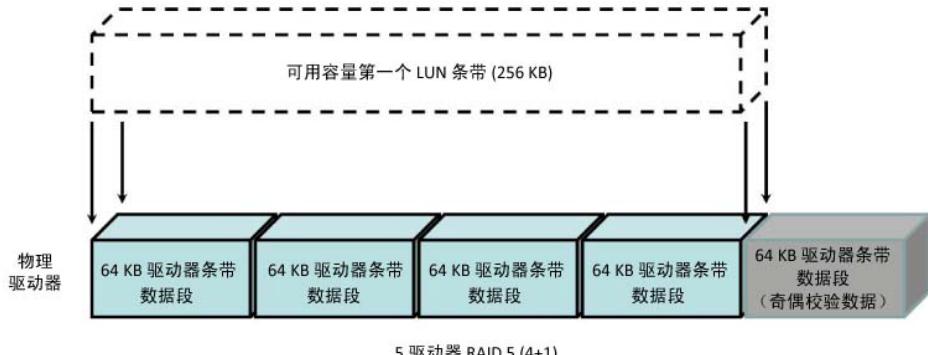


图 18 奇偶校验 RAID 组条带元素

条带元素

条带元素是存储在条带的单个磁盘上的连续数据量。条带元素按 512 字节数据块或千字节 (KB) 测量。默认条带元素大小为 128 个数据块，即 64 KB。条带元素大小不可通过 Unisphere 进行配置。默认元素大小已针对操作环境进行了优化。对默认元素大小进行“调整”可能会降低性能。

条带大小

条带大小是 RAID 组条带中的用户数据量。该大小不包括用于奇偶校验或镜像的驱动器。条带以 KB 为测量单位。其计算方式为条带磁盘的数量乘以条带元素大小。

例如，包含八个磁盘的 RAID 1/0 的条带宽度为 4，当条带元素大小为 64 KB 时，将具有 256 KB ($4 * 64 \text{ KB}$) 的条带大小。包含五个磁盘的 RAID 5 (4+1)，在条带元素大小为 64 KB 时，也具有 256 KB 驱动器的条带宽度。

支持的 RAID 级别

第 62 页的表 4 简要概括了 VNX 系列支持的 RAID 级别。

表 4 RAID 级别概要介绍

RAID 级别	描述	最少磁盘
0	不受保护的分条数据。数据在一组硬盘驱动器上进行分条。提供与 RAID 5 类型相同的单个访问功能，但没有奇偶校验信息。RAID 组中的单个磁盘故障会导致 RAID 组中的数据丢失。	3
1	镜像保护数据。通过将其数据镜像到另一个磁盘来提供数据冗余。这是数据复制。此 RAID 类型以最经济的磁盘空间成本提供高数据可用性。此 RAID 级别的最大大小为两个磁盘。	2
3	分条、奇偶校验保护的数据。使用存储在 RAID 组中一个物理磁盘上的奇偶校验信息提供数据冗余。	3
5	分条、分布式、奇偶校验保护的数据。使用存储在 RAID 组中每个驱动器上的分布式奇偶校验信息提供数据冗余。	3
6	分条、分布式、双奇偶校验保护的数据。在两个驱动器出现故障时提供容错，也就是说，在最多两个驱动器发生故障的情况下，存储系统仍可继续运行。RAID 6 具有最高的可用性级别。	4
1/0	镜像保护的数据，出于性能考虑进行了分条。通常称为 RAID 10，有时称为 RAID 1-0。提供高性能和高数据冗余，但存储容量成本高。	2

RAID 级别不是指用来实施该 RAID 级别的驱动器的数量。这是技术新手经常会有的错误假设。

例如，RAID 5 组的条带中不一定有 5 个驱动器。

RAID 组构成

不能在 RAID 组中将不同的驱动器类型（闪存、SAS 和 NL-SAS）绑定在一起。例如，不能将闪存驱动器绑定到包含机械硬盘驱动器的 RAID 组中。Unisphere 会阻止绑定不同类型的驱动器。

理想情况下，应使用相同速度（15k rpm、10k rpm 和 7.2k rpm）和相同容量的驱动器创建 RAID 组。这称为 RAID 组对称。RAID 组中具有相同驱动器可确保最高的可用容量以及一致的性能和可用性。

如果在 RAID 组中混用不同容量的驱动器，那么在构成 RAID 组的总容量时，组中的每个磁盘都只会用到绑定的最小驱动器的容量部分，而不会用到 RAID 组的较大磁盘的完整容量。

一个具有 5 个 600 GB（原始）SAS 驱动器的 5 驱动器 RAID 5 组的原始用户容量为 2.4 TB ($4 * 600 \text{ GB}$)。如果该 RAID 组由 4 个 600 GB 驱动器和 1 个 300 GB（原始）SAS 驱动器组成，则较高容量的驱动器将被削平到最低驱动器容量。该专用 RAID 组的原始用户容量为 1.2 TB ($4 * 300 \text{ GB}$)。请注意，RAID 组的可用容量因包括单个低容量的驱动器而减少了一半。

同样，将较高 rpm 的驱动器与较低 rpm 的驱动器混合使用，也会降低 RAID 组的总体性能。与到较快的驱动器的 I/O 相比，到较慢驱动器的 I/O 请求需要更多的时间来完成。这会增加 RAID 组的平均响应时间和多驱动器 I/O 的响应时间。

例如，一个包含 5 个驱动器的 RAID 5 组由 4 个 15K rpm SAS 驱动器和 1 个 10K rpm SAS 驱动器组成。到 10K rpm 驱动器的 I/O 响应时间比到其同组驱动器的响应时间要慢。对于随机 I/O（包括单个驱动器），区别可能不明显。但是，对于到 10k rpm 驱动器的 I/O，以及涉及完全条带写入的 I/O，响应时间会慢一些。就是因为添加了这一个较慢的驱动器，该 RAID 组的总体吞吐量低于全部由 15K rpm 驱动器组成的 RAID 组。

RAID 组对称也适用于虚拟池。（请参阅虚拟池一节。）为达到理想的性能和容量利用率，同构池中的所有驱动器都应当相同。在 FAST VP 池中，同一层中的所有驱动器也应当相同。

RAID 组的最小大小

表 4（第 62 页）提供了支持的 RAID 级别的最少驱动器数。

最小的 RAID 组大小为两个驱动器。仅镜像类型 RAID 可以采用这种最少量的驱动器配置。配置包含两个驱动器的 RAID 1/0，随时间推移可向镜像 RAID 组添加额外的驱动器；与之相反，使用相同的初始配置配置包含两个驱动器的 RAID 1，则不允许直接添加额外的驱动器。（可以使用 LUN 迁移或 metaLUN 来增加其容量。）

虚拟资源调配池默认专用 RAID 组具有不同的最小大小值。

驱动器数量

根据 RAID 级别的最小所需磁盘数量（参见第 62 页的表 4），一个组最少可以包含 2 个硬盘驱动器，最多可以包含 16 个硬盘驱动器。

例如，对于 RAID 级别 6 的 RAID 组，最少可包含 4 个硬盘驱动器，最多可包含 16 个磁盘。

条带宽度

简单地说，条带宽度就是指条带中驱动器的数量。它适用于分布式奇偶校验方案（RAID 5 和 RAID 6），也只有在该上下文环境中才真正有用。宽度定义在随机 I/O 中可以并行读取的驱动器的数量。因此，它是条带中驱动器的数量。

RAID 组表示法

RAID 组表示法在括号中使用两个数字，数字之间以加号 (+) 分开。例如：(4+1)。这两个数字的和是 RAID 组中的驱动器总数。第一个数字表示组中可用于数据存储的驱动器数量。第二个数字表示用于数据冗余的驱动器数量或根据容量计算的驱动器当量（参见第 64 页的“RAID 组和 LUN 容量”）。

下面是可能的 RAID 组示例：

- ◆ 包含 2 个磁盘的 RAID 1 组: (1+1)
- ◆ 包含 5 个磁盘的 RAID 3 组: (4+1)
- ◆ 包含 7 个磁盘的 RAID 5 组: (6+1)
- ◆ 包含 10 个磁盘的 RAID 6 组: (8+2)
- ◆ 包含 12 个磁盘的 RAID 1/0 组: (6+6)

请注意，RAID 组表示法不区分 RAID 类型 5 和 3。

创建 RAID 组

可通过以下方式创建 RAID 组：使用 Unisphere 或命令行界面 (CLI)、使用存储系统管理软件，或通过功能自动创建。RAID 组是基本的驱动器组织级别。

每个存储系统可创建的 RAID 组的最大数量取决于机型。入门级存储系统比较高机型的 RAID 组要少。

绑定是使用 CLI 创建 RAID 组的过程的名称。可以将相同类型的任何驱动器绑定到 RAID 组。操作环境会强制实施同一类型的驱动器。RAID 组中驱动器的速度和容量不必完全相同，操作环境也不会强制相同。但是，出于性能、容量利用率和维护原因，最好在 RAID 组中使用相同速度和容量的驱动器（请参见第 84 页的“硬盘驱动器规格”了解详细信息）。

请注意，在某些情况下，要将 RAID 组配置为具有最大性能，可能需要使用 Unisphere CLI。

RAID 组和 LUN 容量

不同的 RAID 级别具有不同的容量使用级别。创建 RAID 组时，宿主于组中的 LUN 的最终可用用户数据容量取决于所选择的 RAID 级别。（请参见第 67 页的“逻辑单元 (LUN)”。）Unisphere 针对 RAID 组报告的“总容量”表示 RAID 组中所有可分配的存储。它是除去磁盘级别格式化、专用空间使用和 RAID 保护外的净值。

在 LUN 属性中，Unisphere 报告的是“原始”容量，这是磁盘级别格式化的净值，但包括 RAID 保护。许多服务级别协议都基于原始容量进行描述。用户可以使用它作为估计存储系统中所使用的总原始容量的一种方法。

请注意，如果将整个 RAID 组分配给单个 LUN，则“原始”容量将少于驱动器供应商列出的容量（如前面的章节所述）。这是因为二进制到十进制的区别、使用 520 字节扇区以及用于元数据的驱动器容量。

RAID 组驱动器用户容量或数据容量

可用于数据存储的容量称为**用户容量或数据容量**。放在 RAID 组中的驱动器的用户容量比原始容量小，因为用于数据保护的冗余和数据完整性机制都会消耗容量。

镜像 RAID

如果采用镜像 RAID 级别，RAID 级别的驱动器的用户存储容量是驱动器容量的一半。例如，放在 RAID 级别 1/0 (4+4) 中的 8 个未格式化的 600 GB 硬盘驱动器（原始总容量为 4.8 TB）中，可用于数据存储的容量总共为未格式化的 2.1 TB。

奇偶校验 RAID

一般而言，当容量成本相对于随机 I/O 性能而言是更重要的考虑因素时，可以使用奇偶校验类型的 RAID。

使用奇偶校验 RAID，可用于用户数据到数据保护的容量比率会因 RAID 级别和 RAID 组中的磁盘数而变化。

对于奇偶校验 RAID 级别 3 和 5，需要相当于一个驱动器的容量用于奇偶校验。对于 RAID 6，需要相当于两个驱动器的容量用于奇偶校验。在 RAID 5 和 RAID 6 中，没有专用于奇偶校验的驱动器。在这些 RAID 级别中，会使用组中每个驱动器的一部分容量用于奇偶校验。这种机制称为循环奇偶校验。在 RAID 3 中，有一个专用的奇偶校验驱动器。该驱动器仅包含奇偶校验数据，不包含用户数据。

由于用于奇偶校验的驱动器容量是固定的，因此，对于任何大小的 RAID 组，专用于奇偶校验的存储容量百分比会随着奇偶校验 RAID 组中驱动器数量的增加而降低。这使得包含大量驱动器的奇偶校验 RAID 组在 GB 成本上非常经济。权衡容量与可用性时请考虑，与较小的组相比，大型 RAID 组从统计角度而言发生多驱动器故障的可能性更大并且重建需要的时间更长（参见下面的章节）。

例如，RAID 5 (2+1) 可用于数据存储的原始容量为 66%；33% 的总原始容量专用于数据保护，不能用于存储用户数据。通过为该 RAID 组增加两个驱动器使其成为 RAID 5 (4+1)，则专用于奇偶校验的组原始容量百分比将下降为 20%。

RAID 组容量示例

第 66 页的表 5 显示了 VNX 最常调配的 RAID 组（使用可用的硬盘驱动器）的用户容量。请注意，用户容量考虑了用于确保数据完整性的 520 字节的扇区划分方法（参见第 54 页的“容量”）、用于冗余和元数据的 RAID 类型的开销。用户容量为二进制 TB，舍入为 10 GB。二进制 GB 是主机报告的容量。使用表中的 RAID 组绑定的 LUN 向主机报告该容量。主机文件系统格式化会进一步减少可用的用户容量。

表 5 常见的 RAID 组可用容量 (TB)

RAID 级别	驱动器数量	100 GB 闪存 (TB)	200 GB 闪存 (TB)	300 GB SAS (TB)	600 GB SAS (TB)	2 TB NL-SAS (TB)
RAID 1 (FAST Cache)	1+1	0.09	0.18			
RAID 1/0	3+3	0.27	0.54	0.79	1.57	5.37
	4+4 (池默认)	0.36	0.72	1.05	2.10	7.17
RAID 5	4+1 (池默认)					
	8+1	0.72	1.43	2.10	4.19	14.13
RAID 6	6+2 (池默认)	0.54	1.07	1.57	3.15	10.75
	8+2	0.72	1.43	2.10	4.19	14.33

RAID 0

RAID 0 较为特殊。RAID 0 是唯一的分条 RAID 级别。RAID 级别 0 没有数据保护，因此不会占用容量或妨碍性能。采用 RAID 级别 0，该级别的磁盘的原始存储容量为原始驱动器的容量之和。此外，到 RAID 0 组的读取和写入比镜像或奇偶校验 RAID 级别要快。但由于没有数据保护，组中发生任何驱动器故障都会导致全部数据丢失。

磁盘

VNX 支持单个磁盘（磁盘）存储类型。该磁盘类型的工作方式与标准的单个磁盘相同。单个磁盘不是 RAID 存储类型。它不具有奇偶校验或数据镜像提供的数据保护。它没有分条所提供的性能或容量优势。这种 RAID 类型适用于不太重要的临时目录。

专用 RAID 组

虚拟资源配置功能在创建池时创建 **专用 RAID 组**。这些 RAID 组由该功能的软件管理，用户无法访问。（请参见第 67 页的图 19 和第 74 页的“虚拟池”。）

RAID 组扩展

RAID 组扩展是向现有 RAID 组添加额外驱动器的过程。与旧的 EMC 中端存储系统不同，VNX 系列不支持 RAID 组扩展。

40 驱动器存储池

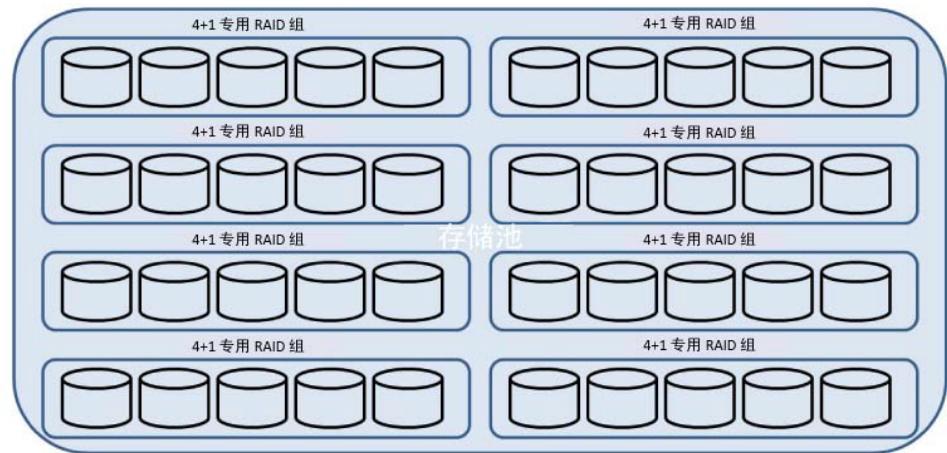


图 19 40 个驱动器的 RAID 5 存储池概念图

逻辑单元 (LUN)

逻辑单元 (LUN) 是叠加在 RAID 组上的逻辑存储结构。主机将 LUN 视为物理磁盘。LUN 常常被称为磁盘、卷或分区，具体取决于上下文。LUN 向主机隐藏了 RAID 组的组织和构成。创建 LUN 的目的是分配容量、确保性能和保证信息安全。请注意，信息安全不是数据保护。信息安全是保密功能。

用户 LUN 和专用 LUN

LUN 还可细分为以下两类：用户 LUN 和专用 LUN。用户 LUN 是主机可见的 LUN，用户存储操作在用户 LUN 上执行。

专用 LUN 支持用户相关或功能创建的 LUN。用户对专用 LUN ID 的分配没有控制权。

专用 LUN 示例包括 metaLUN 组件、FAST Cache 的内容和保留 LUN 池 LUN。

专用 LUN 不能属于存储组，服务器不能针对它们执行 I/O，但 metaLUN 除外。使用 metaLUN 的组件，服务器可以通过 metaLUN 向那些专用组件执行写入操作。

RAID 组和 LUN

RAID 组的数据容量可以分区为一个或多个 LUN。（请参见第 68 页的图 20。）每个 RAID 组的最大 LUN 数取决于操作环境版本。对于 Operating Environment Block 31 版和更高版本，每个 RAID 组的最大 LUN 数为 256。

LUN 的容量可以是一个数据块的大小，也可以是 RAID 组的最大容量。LUN 从基本 RAID 组的所有磁盘中均等获取容量。请注意，虚拟资源调配池和 metaLUN 是使用专用 LUN 来实施的。

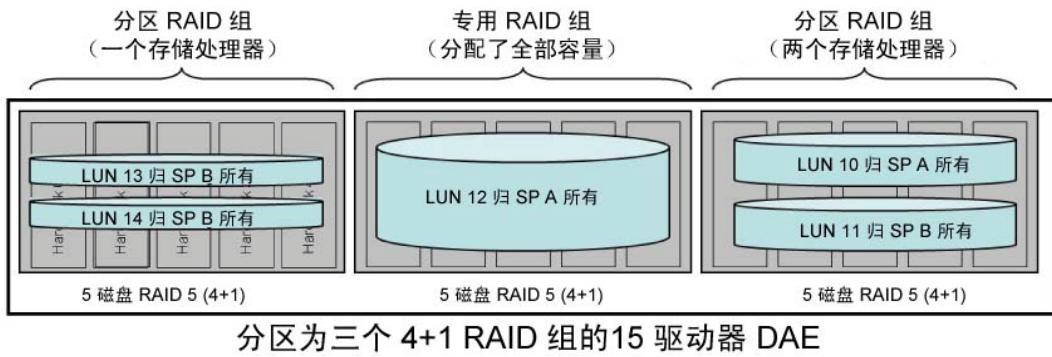


图 20 LUN 概念图

专用 RAID 组与分区 RAID 组

只有一个 LUN 并使用所有可用用户容量的 RAID 组是 **专用 RAID 组**。有多个 LUN 的 RAID 组是 **分区 RAID 组**。

由于单个驱动器的容量很大，单个 RAID 组的容量也可能很大。可以使用 LUN 将 RAID 组的可用容量分区为较小的部分。大部分 RAID 组的容量都是通过这种方式来共享的。

分区 RAID 组上的 LUN 包含连续的 RAID 条带集。LUN 跨 RAID 组的所有驱动器进行分条，并相互“堆叠”。使用大型 I/O 大小的顺序读取和写入以及 I/O 操作会导致基本 RAID 组的所有驱动器以并行方式工作。这是一种很有效地利用存储系统资源的方式，但是会带来高带宽。

按定义，虚拟池的所有专用 RAID 组都是分区 RAID 组。

绑定 LUN

创建 LUN 的过程称为 **绑定**。只有在创建 RAID 组后，才能绑定 LUN。

可以通过 Unisphere 手动创建 LUN，也可以通过功能自动创建。自动创建的 LUN 有时称为 **专用 LUN**，但某些特定专用 LUN 也可能是手动创建的。也可以称为 **保留 LUN**。用户对专用 LUN 有受限管理权或无权管理。

手动绑定 LUN 时，用户对 RAID 组中 LUN 的位置具有有限控制权。在未使用的 RAID 组中创建的第一个 LUN 会始终占用以最低编号的 LBA 开始的容量。这些 LBA 占用机械硬盘驱动器最外侧的磁道。后续创建的 LUN 会占用紧随之前创建的 LUN 所占用的容量之后的容量。如果两个 LUN 之间存在未使用的容量，由于删除了一个 LUN 并且新的 LUN 能够填补缺口，则会将新 LUN 绑定到该位置。当 RAID 组中没有足够的连续容量来容纳 LUN 时，便不能创建 LUN。可以对 RAID 组进行碎片整理，以获得连续空间并使容量利用率最大化。

绑定过程

LUN 创建后即可马上投入使用。称为 *FastBind* 的过程使得 LUN 可立即投入使用。但是，只有当所有绑定的存储都已准备就绪并且经过验证后，绑定才真正完成。准备和验证操作通过后台处理进行。

准备和验证步骤所需的时间各不相同。这取决于 LUN 的大小、验证优先级和存储系统的工作负载。

准备

在执行准备步骤期间，将使用二进制的零覆盖分配给 LUN 的存储。这些零会擦除存储中所有之前的数据并针对奇偶校验计算进行设置。当归零操作完成后，将为 LUN 扇区计算奇偶校验和元数据。如果是直接从 EMC 购买的新系统，则已经预先进行了存储“归零”处理。这样，在进行驱动器初始绑定时便无需再执行此步骤。此外，如果通过使用 Unisphere CLI 提出相关要求，可以手动对驱动器进行预先归零操作。

验证

验证包括背景验证 (BV)。BV 是读取 LUN 的奇偶校验扇区并验证其内容的过程。这是存储系统的一项可用性功能。绑定完成后，会默认执行 BV。

在初始化新 LUN 的过程中，如果工作负载在 BV 完成之前就开始使用 LUN，会对性能产生负面影响。此外，大型虚拟池的 BV 时间会比较长。在 BV 完成之前，这可能会导致池资源的高利用率。

可以在 Unisphere 中手动覆盖 BV 默认值，以加快绑定速度。当存储处理器检测到条带的奇偶校验和硬盘驱动器的扇区奇偶校验之间存在差异时，也会安排 BV。

LUN 所有权

LUN 由单个存储处理器管理和访问。这称为 *LUN 所有权*。当绑定 LUN 后，系统会在 Unisphere 内以循环调度的方式自动将 LUN 所有权分配给存储处理器。

可以通过 Unisphere 手动更改所有权。有时候出于性能或可用性原因，可能有必要将 LUN 所有权更改为对等存储处理器所有。更改所有权的过程称为主动变更。主动变更不会移动数据。新获所有权的存储处理器只需在其原始 RAID 组上管理主动变更的 LUN 的 I/O。

例如，可能需要进行主动变更来平衡存储系统内的存储处理器使用。

还可通过操作环境自动对 LUN 进行主动变更。这是一项可用性功能。这通常是出现故障情况时的结果，在这种情况下，存储处理器会丢失与一个或多个其 LUN 的连接。修复故障后，可以恢复也可以不恢复 LUN 的原始所有权。

(请参见第 122 页的“主动/被动所有权模式”。)

LUN 标识

LUN 通过其 LUN 名称或其 LUN ID 标识。

LUN 名称

*LUN 名称*是用户指定的标识符。名称可以是任意具有标识性的文本字符串。用户可以通过 Unisphere 创建最多包含 64 个字符的自由形式的文本字段，作为主机 LUN 标识。不会检查重复的 LUN 名称，也不会对重复名称进行限制。

LUN 在内部通过其 *LUN ID* 标识。CLARiiON LUN ID 的完整范围包括从零到基于机型的最大值。LUN ID 的范围与每个存储系统的最大 LUN 数一样，取决于机型，存储系统型号越高，能够承载的 LUN 的数量就越多。

Unisphere 使用由存储系统指定的 LUN ID 编号（参见下一节），而不使用可更改的 LUN 名称。但为了方便用户，系统会将 LUN 名称映射到 LUN ID。

用户 LUN ID 和专用 LUN ID

就像有用户 LUN 和专用 LUN 一样，LUN ID 也有两种类型：用户 LUN ID 和专用 LUN ID。可用的 LUN ID 编号（用户和专用）始终大于可在存储系统上创建的最大 LUN 数量。不过，使用中的用户和专用 LUN ID 的和不能超过存储系统机型的最大总 LUN 数。（始终有额外的 LUN ID。）

系统会自动为通过 Unisphere Manager 创建的用户 LUN 分配可用的用户 LUN ID，从 0 开始，以 1 为增量递增（每个创建的 LUN 都这样）。用户也可以在创建 LUN 时从机型的 LUN ID 范围中选择未使用的可用用户 LUN ID 编号。

在创建专用 LUN 时，可从 LUN ID 范围中可用的最高编号开始为专用 LUN 分配 LUN ID。例如，VNX7500with OE Block 31.0 及更高版本上的最大 LUN 数为 8192。可以分配的所有可能 LUN ID（用户和专用）的范围为 0 到 16833（总数为 16834），只要可用，用户便可以任意选择这些编号。

LUN 扩展

扩展是向基本 LUN 添加容量的过程。但 LUN 的容量不能大于其虚拟池的最大容量或其基本 RAID 组的容量。但可以创建包含存储系统总容量的 metaLUN，只需减去系统驱动器所需的容量。

取消绑定

取消绑定 LUN 会释放绑定到该 LUN 的组成驱动器的所有容量供重新利用，这会导致先前存储的数据丢失。当删除某个 LUN 或某个虚拟池被破坏时，将通过 Unisphere 执行取消绑定操作。

当某个 RAID 组的所有 LUN 都取消绑定时，用户可以选择销毁该 RAID。

RAID 组碎片整理

当分区 RAID 组的某个 LUN 取消绑定时，该 RAID 组可能会出现碎片。当 RAID 组的 LUN 之间存在未使用容量缺口时，就会出现 LUN 碎片。这些碎片会使 RAID 组中用于新 LUN 或用于扩展现有 LUN 的连续空间减少。此外，这种缺口还会增加 RAID 组内 LUN 之间的寻道距离。

可以通过 Unisphere 对 RAID 组进行碎片整理。碎片整理会从物理上将 RAID 组中 LUN 使用的容量移动到彼此相邻的位置。这样可以消除缺口并恢复 RAID 组中之前分开的 LUN 之间的未使用容量。如果 RAID 组中所有使用中容量都是连续的，则可以允许在该组上创建新的更大的 LUN。

请注意，RAID 组碎片整理不适用于虚拟池。从 OE Block 31 起，不能对 RAID 级别 6 RAID 组进行碎片整理。

请注意，RAID 组碎片整理不是文件系统碎片整理。（它们经常被混淆。） RAID 组碎片整理不会影响应用程序数据在 LUN 内的定位。

LUN 队列

对 LUN 的访问通过排队机制来调解。LUN 中每个到达的 I/O 都会占据一个队列位置。存储处理器从队列中移走 I/O 并进行处理。可用的队列条目数称为 LUN 队列深度。

LUN 的最大队列条目数取决于 LUN 中用户数据驱动器的数量。RAID 组中数据驱动器的数量越大，队列越深。

例如，基于 RAID 5 (4+1) 的 LUN 需要 88 个并发请求才会填满其队列。最佳做法¹ 列出了确定 LUN 队列深度的计算方法。如果超过队列深度，存储系统会向主机返回一个 *queue full* (QFULL) (队列已满) 的状态以响应 I/O。主机上 QFULL 的实际影响取决于 O/S，但它肯定会对性能带来负面影响。

QFULL 的情况很少见。主机总线适配器 (HBA) 队列深度设置通常可避免这种情况的出现。请注意，还可能会在 I/O 端口级别触发 QFULL。有关详细信息，请参见第 122 页的“前端”。

保留 LUN 池 (RLP)

保留 LUN 池是一组与 SnapView、增量 SAN Copy 和 MirrorView/A 等复制软件相关联的专用 LUN，用于存储完成复制任务所需的数据或信息。

保留 LUN 池包含一个或多个专用 LUN。LUN 在添加到保留 LUN 池后即成为专用 LUN。

LUN 类型

VNX 支持多个不同类型的 LUN。这些类型为：

- ◆ 虚拟池 LUN
- ◆ 传统 LUN
- ◆ MetaLUN

所有类型的 LUN 都执行相同的功能。但在数据的存储方法和数据存储的扩展方法上存在不同。

虚拟池 LUN

有两类可用的虚拟池 LUN — 没有精简属性的 LUN 称为 *密集 LUN*，具有精简属性的 LUN 称为 *精简 LUN*。池中的 LUN 经常被称为 *池 LUN* 以将它们与 RAID 组中的 LUN (RAID 组 LUN 或传统 LUN) 区分开来。精简 LUN 和密集 LUN 可以占用相同的池，也可以将某个池专门调配为包含一种 或另一种 LUN 类型。

使用池 LUN 可以执行以下操作：

- ◆ 扩展 LUN 的容量
- ◆ 压缩 LUN 的容量 (仅限于 Windows Server 2008 主机)
- ◆ 压缩 LUN 上的数据 (如果存储系统安装了压缩启用程序)
- ◆ 对 LUN 进行自动分层 (如果存储系统安装了 FAST VP 启用程序)

Block OE 31.0 中虚拟池 LUN 的最大容量为 16 TB。

¹ 《EMC Unified Best Practices for Performance and Availability:Common Platform and O.E.XX— Applied Best Practices》 (EMC Unified 通用平台和 O.E.XX 性能和可用性最佳做法 — 应用最佳做法) 可从 Powerlink 获得。

密集 LUN

系统会在创建 LUN 的池中的驱动器之间均衡分配密集 LUN 的容量。密集池 LUN 在创建时会保留其全部容量，但不会从池的调配容量中将这些容量分配给它们自己。这种保留确保在需要时有可用的容量。分配是将那些容量提交到 LUN 的过程。当给定 LBA 范围的 LUN 中收到写入 I/O 时，系统会将保留容量的存储片分配给 LUN。密集 LUN 的调配容量与提供给主机的用户容量相同。由于池中元数据的需要，密集 LUN 使用的存储容量比写入其中的用户数据量要稍微多一点。请注意，调配为专门包含密集 LUN 的虚拟池不会发布容量利用率警报。与精简 LUN 不同，密集 LUN 绝不会有空间不足的情况。

精简 LUN

安装了精简资源调配启用程序的存储系统支持精简 LUN。精简 LUN 提供最大的存储容量使用效率。精简 LUN 与池中其他基于池的 LUN 共享可用的池存储。也就是说，精简 LUN 中包含一个或多个主机 LUN 的使用中容量。

主机可见的精简 LUN 容量与池中的可用物理存储无关。这可以防止过度调配。过度调配是指在使用之前安装和提交物理存储容量。显示物理上不可用的存储可避免主机过度调配以及存储系统容量利用率不足的情况。当精简 LUN 最终需要额外的物理存储时，将从池容量中自动无中断地添加容量。池事先调配了一定余量的容量，以满足当前的存储需求。

例如，如果某个 LUN 在年底时需要 10 TB 的容量，但最初只需要 3 TB，则最初便为该 LUN 调配 10 TB 的容量意味着为单个 LUN 事先准备了多余的 7 TB 存储容量。这些提交但未使用的存储可能对于存储系统中的其他 LUN 更加有用。在此示例中，精简 LUN 将向主机呈现 10 TB 可用容量，但实际用到的池容量最初只有 3 TB。

对主机而言，精简 LUN 将显示为密集 LUN 或传统 LUN。与密集 LUN 或传统 LUN 不同的是，如果所属池用完磁盘空间，精简 LUN 可能会用完容量。此类事件是一种不可恢复的写错误，上次写操作的数据将不可用。默认情况下，当池容量已耗费 70% 时，存储系统将发出警告警报；当已用空间达到 85% 时，它将发出严重警报。由于精简 LUN 消耗的是池的调配容量，因此，警报将持续报告已用容量的实际百分比。与密集 LUN 类似，由于池所需的元数据的缘故，精简 LUN 使用的容量会比写入其中的用户数据量稍多。

池 LUN 类型之间的操作差异

密集和精简 LUN 都通过存储片获配池容量。用户首次访问 LUN 时，将从池容量中向 LUN 分配一个存储片。存储片分配跨后端 SAS 端口及池的专用 RAID 组进行自动平衡。

例如，第 73 页图 21 所示的配有 40 个驱动器的池将包含八个 4+1 专用 RAID 组。池中创建了五个用户 LUN（蓝色、橙色、绿色等）。假设每个用户 LUN 都正好有三个填满用户数据的存储片。另假设蓝色和橙色属于 SP A，而其余的归 SP B 所有。

此图中显示的是一种可能的基于池算法的容量分配。（跨池 RAID 组的蓝色条带是分配给蓝色用户 LUN 的存储片。）总共显示五个用户 LUN 中的十五个存储片，跨八个专用 RAID 组分配。

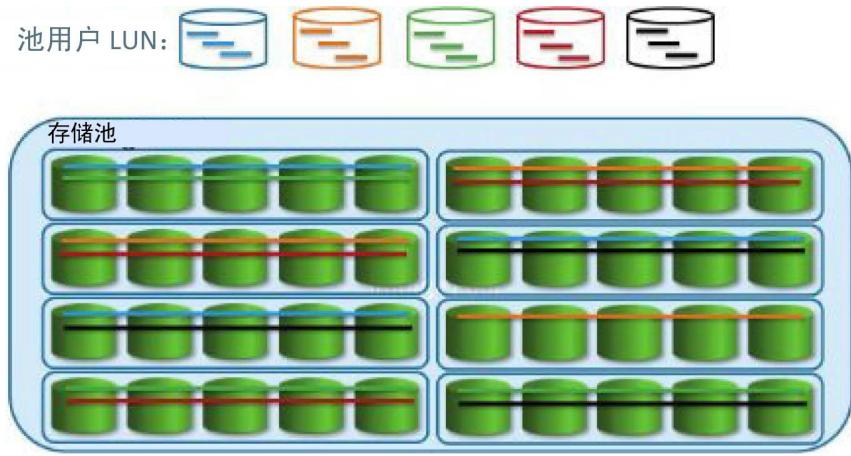


图 21 存储池容量分配示例

就密集 LUN 而言，容量将被保留，但不会分配给 LUN。密集 LUN 绝不会用完容量。保留的容量可防止发生这种情况。密集 LUN 存储片是十亿逻辑块地址 (LBA) 的连续集。这种 LUN 的保留容量将随着容量的分配而递减。

例如，如果新绑定的密集 LUN 收到对 LBA 1015 的写 I/O，将为此 LUN 分配 LBA 1000 - 2000 的容量存储片。若有对 LBA 1500 的后续写入，此地址的容量已经分配给该 LUN。寻址所分配容量不会造成开销。

但是，对在之前分配的容量范围以外的 LBA 的写入将需要从池分配额外的存储片到密集 LUN。

例如，如果对某个密集 LUN 的仅有两个访问针对 LBA 1500 和 2600，将导致向该 LUN 分配两个 1 GB 存储片。这是因为这两个地址的彼此间距比单个存储片的 1 GB 扩展区更远。

就精简 LUN 而言，绑定精简 LUN 时，既不保留也不分配容量。与密集 LUN 相同，对 LUN 执行写入时，将从池中分配 1 GB 存储片。但是，此存储片不是由连续的 LBA 构成，而是以一种更为随机的分配方式，由独立的地址和地址组填充。

在上面的示例中，如果访问的是相同 LUN 中的 LBA 1500 和 2600，这两个写操作仍会在同一存储片中进行。精简 LUN 将从池中获取 1 GB 存储片，并将其用于所有 LUN 访问，直到该精简 LUN 需要更多空间；此时，它将从池中获取另一个 1 GB 存储片。

对密集 LUN 的完全随机 I/O 工作负载将在短时间内耗尽从池中获配的所有存储片。这是因为随机 LBA 可能分配完所有的 LUN 容量；只要写入任何新的 1 GB 扩展区中的 LBA，即会向 LUN 分配新的存储片。要使精简 LUN 分配完所有容量，将需要对每个 LUN 地址执行顺序 I/O。

密集 LUN 的分配容量本质上具有连续寻址，而精简 LUN 存储片并非如此。这两种类型的池 LUN 总会有一些池开销。此开销是由 LBA 映射造成的。此映射将会提高池 LUN 对容量的使用效率。但是，精简 LUN 的映射开销更高。

基于池的 LUN 的调配容量和分配容量通过 Unisphere 进行报告。Unisphere 报告的容量可能与主机文件系统报告的容量使用情况不完全一致。这是因为并非所有主机文件系统都会在取消容量分配时通知存储系统。因此，主机报告的分配容量可能少于存储系统报告的分配容量。此外，主机文件系统与存储系统的虚拟资源调配之间的容量分配报告粒度也可能不同。这可能导致对容量的报告不足或过度，具体视所报告的离散容量值的数量级差异而定。

总之，从性能角度来看，密集 LUN 的性能比精简 LUN 更高，而精简 LUN 可提供更高的容量利用率。

压缩的 LUN

LUN 压缩功能可用于压缩任何 LUN 上的数据以释放存储容量。压缩将执行对密集和传统 LUN 的算法数据压缩。所有压缩的 LUN 将变成精简 LUN。

压缩的 LUN 需要配备 VNX OE 05.31 及压缩和精简资源调配启用程序的 VNX 系列存储系统。

压缩将分析磁盘上的数据，并通过相应算法缩小某些文件类型固有的重复比特序列。与重复数据消除不同的是，压缩与相同数据的多个拷贝毫无关系。存储系统在后台执行 LUN 级别压缩操作，同时持续服务于主机 I/O。

对密集或传统 LUN 的压缩会将 LUN 类型转为精简 LUN。压缩的 LUN 即使在解压缩后也仍是精简 LUN。与精简 LUN 一样，此 LUN 对主机可见的容量等于绑定此 LUN 时指定的大小。

压缩和解压缩流程是一种高度利用存储处理器 CPU 的活动，因此，可能严重影响整体存储系统性能。压缩功能适用于保持不变的用户数据。此类数据有时称为静态数据。压缩不适用于受随机 I/O 影响的数据及特定随机写入 I/O 中的数据。

LUN 压缩是所有 VNX 系统（型号 VNX5300 及更高版本）包括的基本功能。

虚拟池

虚拟池是逻辑存储对象。它由组织到专用 RAID 组中的众多驱动器构成。（请参见第 75 页的图 22 和第 59 页的“逻辑存储对象”。）池 LUN 在虚拟池中创建。

池涵盖一个或多个专用 RAID 驱动器组。驱动器数及支持的虚拟池数视型号而定。型号越高，支持的池数越大。池可由任意数量的驱动器构成。池中驱动器的数量上限是存储系统型号支持的最大数（减去系统驱动器数）。

池功能软件将自动执行存储资源调配、容量分配及池中的数据放置。此外，池功能的图形用户界面也简化了池的资源调配；当指定驱动器组创建或扩展池容量时，此界面将通过算法应用最佳做法。

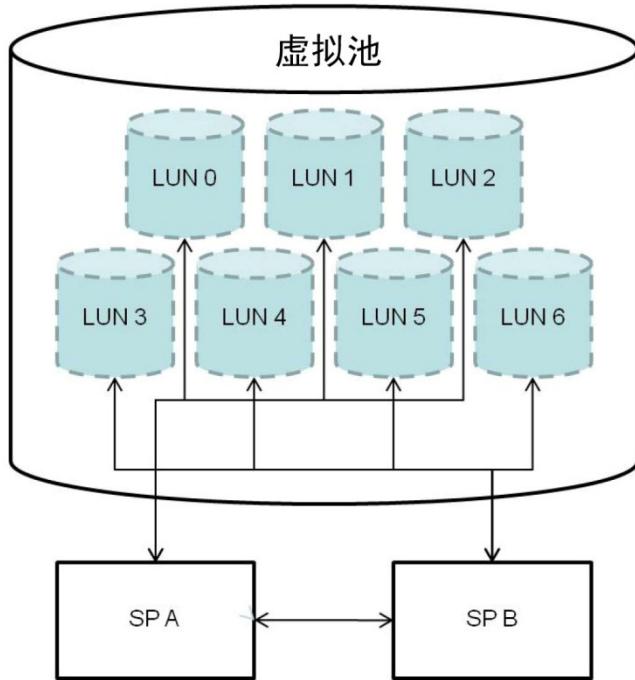


图 22 虚拟池概念图

池既可是具有单一性能级别的同构池，也可是异构池。异构池又名分层池，它使用的是 FAST VP 功能。创建异构池时也可不启用 FAST VP。但是，池中将不执行自动数据分层。除非要在创建这种池后的短时间内添加 FAST VP 启动程序，否则最好不要如此。

使用 OE Block 31.0 时，单一 RAID 数据保护级别将应用于池的所有专用 RAID 组。此规则对同构和异构池都适用。池的 RAID 类型可以是 RAID 5、RAID 6 或 RAID 1/0。

高级别池使用率

创建虚拟资源调配池时，需了解工作负载，平衡所需容量与可交付性能。通常，如果目标是：

- ◆ 基于池的高确定性性能 — 创建具有最大驱动器数的同构存储池。
- ◆ 让最常使用的数据获得最佳池性能 — 创建各层有着相应容量的 FAST VP 池，对最常用的数据使用高性能驱动器，而对不常用的数据使用容量驱动器。

选择基于池的存储

以下注意事项也适用于传统 LUN 性能；当实施基于虚拟资源调配池的存储时，应始终考虑这些注意事项

- ◆ **驱动器争用：**多个 LUN 将共享池中的驱动器容量。调配池时，无法手动控制池中的数据放置。
- ◆ **主机争用：**多个主机可能利用每一个存储处理器。两个存储处理器都具有对池的等效、独立访问权限。除非创建独立池，否则不能控制共享池中的主机访问。
- ◆ **应用程序文件系统布局：**多个应用程序的文件系统可以承载于一个池中。因此，需要规划池的容量和性能，以适应所存储的文件系统。

池的设计是为了便于使用。池对话框通过算法实施大量最佳做法。

平衡容量和性能

与所有资源调配决策一样，需要了解池工作负载的容量和性能要求。

基于池的 LUN 容量利用率

池 LUN 由元数据和用户数据构成，这两种数据都源于存储池。池中创建的每个 LUN 都有关联的固定容量开销。此开销不包括在用户数据容量要求之中。池的初始容量稍大于 LUN 用户数据容量之和。要准确估计需要的初始池容量，您需要了解要创建的 LUN 数量、其类型（密集或精简）及其容量。

基于池的 LUN 性能利用率

池 LUN 由存储池软件进行管理。此外，LUN 还会与其他 LUN 共享池的存储资源。这些都会给 LUN 带来少量性能开销。性能余量（通常是额外的 IOPS）需要维持在池的性能预算以内，以应对池开销和 LUN 之间对共享存储资源的争用。

池和工作负载

性能和容量的以下工作负载组合适用于虚拟池：

- ◆ 高容量/高性能
- ◆ 高容量/适中性能
- ◆ 低容量/适中性能
- ◆ 低容量/高性能

高容量和高性能是理想的池工作负载组合。池容量越高，为满足容量要求所需的驱动器数越大。驱动器数越大，池性能越高。高容量、高性能 SAS 驱动器可满足此类池的要求。同样，也可应用更多性能适中的 SAS 驱动器。SAS 驱动器数越大，可用的池性能余量也就越大。但是，需了解个别 LUN 的性能，以确保没有对池资源的争用。

在池中可轻松实现高容量和适中性能。少量容量极高、性能适中的 NL-SAS 驱动器即可满足此类池的要求。相反，可使用更多容量适中的 SAS 驱动器。SAS 驱动器数越大，池中的性能余量越高。

几乎所有池资源调配都能实现低容量和适中性能。既可使用少量 SAS 驱动器，也可使用更少的 NL-SAS 驱动器。

低容量和高性能是一个成问题的组合。这种要求可能导致为实现性能要求而过度调配容量资源，或者导致为更贴合容量要求而对 IOPS 的资源调配不足。如果可能的话，可使用高性能闪存驱动器来满足这种池工作负载的工作负载需求。

同构池

同构池需使用单一驱动器类型进行资源调配：闪存、SAS 或 NL-SAS。

同构池是最容易配置和维护的虚拟资源调配池。只有一种驱动器类型时，量化和预测性能也更轻松。

配备针对虚拟池的全自动存储分层 (FAST VP) 的同构池

利用 FAST VP，可在由多种驱动器类型构成的池中实现自动数据分层。

分层允许在一个层（而非整个池）中经济地调配存储设备。各层都可配备不同类型的驱动器。分层存储将基于性能在池中创建单独域。此功能的软件将利用算法，基于数据访问频率在层之间进行用户数据的升级和降级。访问较频繁的数据将移至较高性能层。不常访问的数据将按需移至性能适中的高容量层，以腾出更多的高性能驱动器空间供常访问的数据使用。随着时间推移，最常访问的数据将驻留于最快的存储设备上，而不常访问的数据将驻留于经济实惠、性能适中的大批量存储上。

FAST VP 是一种可单独许可的功能，型号 VNX5300 及更高版本的 VNX 系统上全都具备此功能。有了 FAST VP，即使是异构驱动器，也可对任何或所有池进行分层。

传统 LUN

传统 LUN（如第 77 页的图23 中所示）是直接实施在 RAID 组上的传统 LUN。它们支持：

- ◆ Celerra 文件系统
- ◆ 传统 CLARiiON 应用程序（例如，支持脚本和软件）
- ◆ 对更低级别的逻辑至物理数据映射的访问

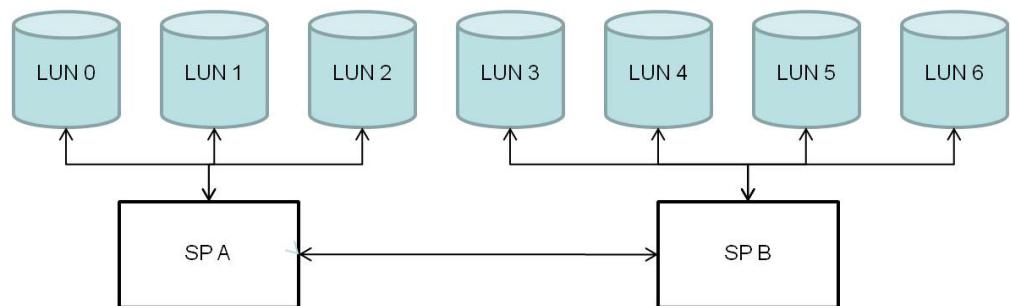


图 23 传统 LUN 概念视图

传统 LUN 的最大容量视 RAID 组的驱动器数及 RAID 级别而定。VNX 支持的所有 RAID 级别都可供使用。一个 RAID 组最多可由 16 个驱动器构成。

例如，一个使用整个 RAID 组容量的 15 驱动器、2 TB NL-SAS RAID 5 级传统 LUN 约有 27 TB 用户容量。注意：从可用性的角度来看，以这种方式调配 LUN 可能不够谨慎。

传统 LUN 的容量跨 LUN 所绑定的 RAID 组的驱动器均匀分配。此 LUN 可绑定到 RAID 组的全部或一部分可用容量。传统 LUN 的调配容量与呈现给主机的用户容量相同。

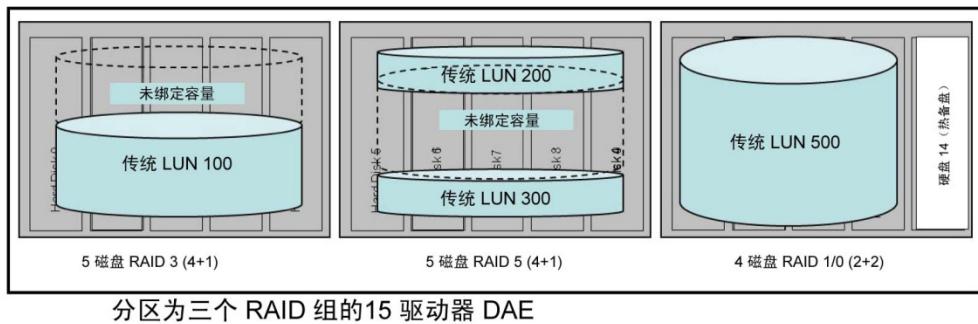


图 24 LUN 与物理驱动器的关系

MetaLUN

MetaLUN（第 78 页上的图 25）是从两个或以上传统 LUN 创建得来的 LUN。对于所需容量超过单个 RAID 组容量的 LUN，或者最大容量超过虚拟池可用容量的 LUN，MetaLUN 便是一种解决方案。

所支持的最大 metaLUN 容量是指存储系统上所有驱动器（减去系统驱动器）的容量。

借助 MetaLUN，基本 LUN 容量还可在不向主机额外添加独立 LUN 的情况下实现后期增长。这将减少主机和存储系统上需管理的 LUN 数量。此外，创建 metaLUN 还可通过添加驱动器提升 LUN 性能。



图 25 MetaLUN 概念视图

MetaLUN 包含基本 LUN 及组件 LUN。MetaLUN 由两个或以上组件构成，最多可包含 16 个组件。基本 LUN 是向 metaLUN 提供可寻址身份的特殊组件 LUN。MetaLUN 容量是基本 LUN 与所有组件 LUN 的组合容量。

MetaLUN 扩展

扩展是一种向基本 LUN 或现有 metaLUN 添加容量的流程。可创建一个包括存储系统上所有驱动器（减去系统驱动器）的 metaLUN。但是，由于主机寻址限制、资源调配限制及可用性问题，这或许不可行。

可通过向 MetaLUN 添加额外的 LUN，进一步提高其容量。一个 metaLUN 组件最多可包括 32 个 FLARE LUN。MetaLUN 扩展期间，基本 LUN（基本组件）数据始终可供访问。附加容量需等到扩展完成后才可用。MetaLUN 可以通过串联（第 79 页上的图 26）或分条（第 79 页上的图 27）方式扩展。

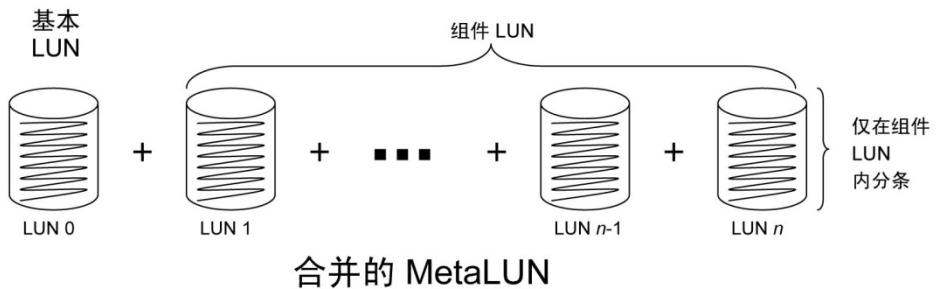


图 26

串联的 metaLUN

串联扩展仅向基本实体添加额外容量。此基本实体可以是混合 metaLUN 结构中的基本 LUN 或分条 metaLUN 组。串联组件 LUN 不必与基本 LUN 有着相同容量或相同 RAID 类型。串联 LUN 时，所有组件 LUN 必须具有相同的保护级别。要串联的 LUN 必须全都不受保护（如 RAID 0）、全都受 N+1 保护（如 RAID 5 或 RAID 3）或全都受 N+2 保护（如 RAID 6）。也就是说，RAID 6 LUN 只能与 RAID 6 LUN 串联。串联扩展非常快，但是额外的驱动器可能不会带来性能好处。

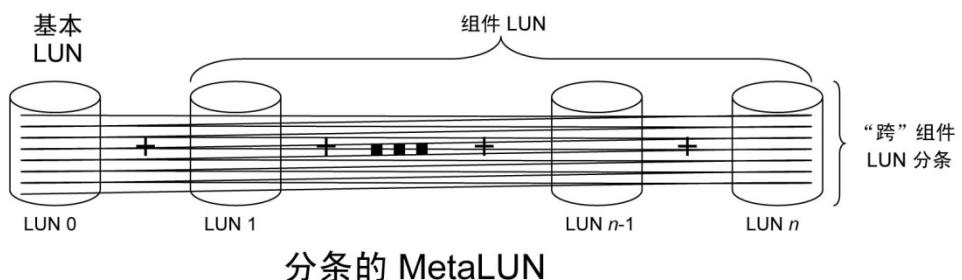


图 27

分条的 metaLUN

分条扩展将跨基本 LUN 及要添加的组件 LUN 对基本 LUN 的数据重新分条。一次可添加一个或多个组件 LUN（第 79 页上的图 28）。在分条扩展中，所有 LUN 的容量和底层 RAID 组级别必须相同。此外，所有 LUN RAID 组磁盘必须是相同的磁盘类型。随着要分条的驱动器数的增加，性能会随之提升（请参见第 89 页上的“分条性能”）。分条扩展是 VNX metaLUN 默认值。

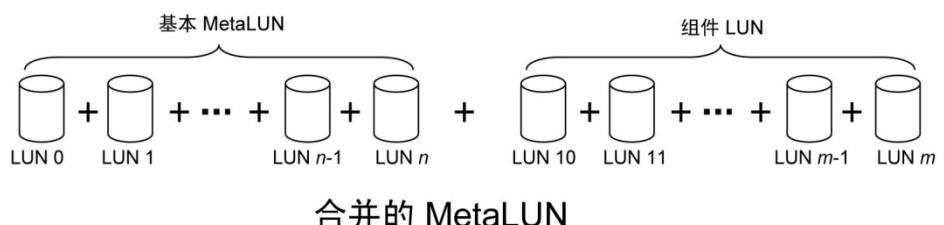


图 28

串联组件 (metaLUN)

所有 metaLUN 都有基本组件（称为组件 0）。MetaLUN 可通过将附加 LUN 串联到基本组件进行扩展，从而使卷变得极大。这将创建一种组件层次结构。

取消绑定

取消绑定 metaLUN 可释放并取消绑定其所有组件 LUN，而销毁组件 LUN 将导致之前存储的数据丢失。

存储组

存储组是 LUN 的一个访问控制机制。它将 LUN 组隔离开来，以避免特定主机访问。

配置存储组时，您需要标识一组仅由一个或多个主机使用的 LUN。然后，存储系统将强制从该（这些）主机访问 LUN。这些 LUN 仅提供给存储组中的主机，并且主机只能看到该组中的 LUN。

可向一个存储组分配大量 LUN，而存储系统可能有众多存储组。但是，每个存储组的 LUN 数量及每个存储系统的存储组数量视型号而定。有关当前的最高配置，请参见《最佳做法》文档。

物理资源调配示例

下面展示的是存储系统资源调配示例。此示例是为了展示逻辑至物理存储对象的映射方法。

所有工作负载都不同。存储系统的调配方法有很多。这里介绍的只是其中一种方法。此为演示性示例，只是为了展示规划存储系统资源调配的方法，因此并不全面，也不是决定性的。

存储系统的物理资源调配视以下几项而定：

- ◆ 工作负载容量要求
- ◆ 可用性和性能目标
- ◆ 存储系统型号
- ◆ 存储模块的类型和数量
- ◆ 驱动器的类型和数量
- ◆ 已安装的功能

鉴于所有这些因素，可能的资源调配组合非常多。

假设此示例的存储系统有多个工作负载，而且详细分析给出以下资源调配建议。简单地说，工作负载如下：

- ◆ 具备 RAID 6 FAST VP 池的高可用性分层存储
- ◆ 高容量驱动器上具备 RAID 5 传统 LUN 的“大批量、低成本存储”
- ◆ 具备 RAID 1/0 传统 LUN 的高性能、最低写入延迟存储

还包括 FAST Cache 功能、系统驱动器和可用性相关的热备盘。

有关如何执行此分析的说明，请参见《EMC Unified Best Practices for Performance and Availability: Common Platform and O.E.Block XX—Applied Best Practices》（EMC Unified 性能和可用性最佳做法：通用平台与 O.E. Block XX — 适用的最佳做法）（VNX 最佳做法）。此文档可从 [Powerlink](#) 获得。

Error! Reference source not found.（第 **Error! Bookmark not defined.** 页）汇总了满足示例工作负载要求的逻辑和物理存储对象。单元格内带括号的是 RAID 组配置。

表 6 物理资源调配示例工作负载

驱动器类型	FAST Cache (驱动器数)	RAID 5 传统 LUN (驱动器数)	RAID 6FAST VP 池 (驱动 器数)	RAID 1/0传 统LUN (驱 动器数)	按类型列 出的驱动 器总数
100 GB 闪存	4 (2x (1+1))				5
100 GB 闪存热 备盘数	1				
300 GB SAS 10K rpm 数据 驱动器数			16 (2x (6+2))	16 (2x (6+2))	37
300 GB SAS 10K rpm 热备 盘数				1	
300 GB SAS 10K rpm 系统 驱动器数			4		
2 TB NL-SAS 7.2K rpm 数据 驱动器数		5 (1x(4+1))	24(3x (6+2))		30
2 TB NL-SAS 7.2K rpm 热备 盘数			1		
工作负载驱动 器数 (减去热 备盘数和系统 驱动器数) :	4	5	40	16	72

有关使用和配置 FAST Cache 的说明, 请参见第 48 页上的“FAST Cache”。有关使用和配置热备盘的说明, 请参见第 124 页上的“全局热备份”。有关系统驱动器的说明, 请参见第 123 页上的“系统驱动器和写缓存可用性”。有关工作负载配置的论述, 请参见第 71 页上的“LUN 类型”。

在此示例中, 假设的是具备第 **Error! Bookmark not defined.** 页 “**Error!** **Reference source not found.**” 中所示的存储硬件配置的 VNX5300。

表 7 资源调配示例硬件配置

资源调配示例 VNX5300 配置			
	类型	机型	数量
存储模块	DPE	DPE7	1
	DAE	DAE6S	4
存储模块总数:			5

注意: 在此示例中, 仅显示了承载驱动器的存储模块。实际配置将包括附加存储模块。所有承载驱动器的存储模块都是 3.5 英寸 15 驱动器存储模块。共有 75 个驱动器插槽供存储系统使用。因此, 所有驱动器必须都是 3.5 英寸的外形尺寸。具有更多驱动器插槽和不同驱动器外形尺寸的存储模块也可供使用。(请参见第 34 页的“存储处理器存储模块”和第 37 页的“磁盘阵列存
储模块”。) 此外, VNX5300 是具有总线 0 和 1 的双后端 SAS 端口存储系统。

图 29（第 82 页）展示了存储模块上驱动器布局的直观浏览方法。在此表中，“HS”是热备盘的缩写词。行表示承载驱动器的存储模块，列表示驱动器插槽。此布局应包括与逻辑和物理存储对象属性相匹配的容量、性能和可用性目标。例如，在图中，闪存驱动器经过纵向资源调配后，跨所有可用后端 SAS 端口分配其 I/O。为确保最高可用性，每个 RAID 1 对及其镜像应分处于独立的后端端口上。有关此资源调配方法的论述，请参见第 106 页上的“后端总线性能测量”及《VNX 最佳做法》。

DAE/总线	驱动器插槽														
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
DAE4 (总线 0)	NL-SAS RAID 6 FAST VP 池 0 驱动器													FAST Cache 闪存驱动器	
DAE3 (总线 1)															
DAE2 (总线 0)	SAS RAID 1/0 传统 LUN 驱动器							SAS RAID 6 FAST VP 池 0 驱动器							
DAE1 (总线 1)															
DPE (总线 0)	SAS 系统驱动器	NL-SAS RAID 5 传统 LUN			NL-SAS HS	空			SAS HS	闪存 HS					

图 29 工作负载的物理资源配置示例 — 驱动器布局

第 9 章 存储对象性能

本章包括以下主题：

驱动器性能	84
RAID 组性能	89
RAID 级别性能差异	92
RAID 组性能计算	95
LUN 性能	95
百分比利用率	98
共享的 RAID 组	105
后端总线性能测量	106
可靠性	108
冗余	108
测量可靠性和可用性	110

驱动器性能

驱动器是主要的物理存储设备。VNX 可承载机械硬盘驱动器和基于非易失性内存的驱动器。

机械硬盘驱动器是传统存储。硬盘驱动器性能由旋转磁盘盘片 rpms、寻道时间及猝发传输率等因素决定。它们具有从适中到极高的容量。

VNX 系列还提供基于内存的驱动器（在 IT 行业称为 SSD）。在 VNX 上，这些驱动器称为闪存驱动器。它们较之传统硬盘驱动器有着高可用性和高性能的特点。在某种程度上，其高可用性得益于：无移动部件，因此没有磨损。这些驱动器也无需寻道。它们是随机访问设备，因此执行 I/O 的速度非常快。但是，就目前的技术而言，这些驱动器只能提供适中的容量。其读取要比写入快得多。此外，对顺序 I/O 的执行效率要逊色于随机 I/O。然而，不论是随机 I/O 还是顺序 I/O，其性能都比机械硬盘驱动器要快。

硬盘驱动器规格

有三种重要规格用于描述机械硬盘驱动器性能：

- ◆ 寻道时间 — 是指在磁道间移动读/写磁头所需的时间。
- ◆ 旋转延迟 — 是指磁盘盘片移到驱动器读/写磁头下方所需的时间。
- ◆ 传输率 — 是指硬盘驱动器的带宽。

了解这些规格之间的关系，有助于对硬盘驱动器的性能分类。这些值对于确定硬盘驱动器的以下两项重要性能指标很重要：吞吐量和响应时间。

寻道时间

寻道时间是指在磁道间移动读/写磁头所需的时间。寻道时间以毫秒为单位。寻道时间因驱动器而异。

平均寻道时间是常用的指标；但是也可同时指定读取和写入的寻道时间。典型的 15k rpm SAS 硬盘驱动器寻道时间为 3.5 ms（读取）和 4.0 ms（写入），因此平均寻道时间为 3.8 ms。寻道时间也可逐个磁道地进行测量，或者跨磁盘盘片半径（即，完整冲程）进行测量。

最大限度缩短硬盘驱动器的寻道时间可提升性能。I/O 类型会影响到驱动器的寻道时间。顺序 I/O 的寻道时间最短，因为读/写磁头在磁盘盘片上的寻道操作是连续的。设定驱动器磁头后，几乎不需要执行寻道。随机 I/O 的寻道时间较长，因为读/写磁头要针对每个新 I/O 在磁盘盘片的新位置上不断地进行定位与重新定位。

滞后时间

与硬盘驱动器性能密切相关的重要因素是旋转延迟。延迟是指磁盘盘片移到驱动器读/写磁头下方所需的时间。延迟以毫秒为单位。与较慢的驱动器相较而言，速度较高的驱动器（速度以 rpm 为单位）可提供明显更高的总体随机访问吞吐量和稍短的响应时间。表 8（第 87 页）展示的是平均延迟与转速之间的关系。

传输率

在驱动器级别，传输率是指硬盘驱动器的带宽。传输率以 MB/s 为单位。传输率分为内部速率和外部速率。

内部速率是指磁盘盘片的实际数据读写速度。外部磁道的传输率比内部磁道高。在相同的直线距离上，外部磁道拥有更多的扇区。

例如，对于使用 3.5 英寸 15k rpm SAS 硬盘驱动器的顺序带宽应用程序，典型的内部传输率为 50 MB/s（内部磁道）- 100 MB/s（外部磁道）。（请参见第 96 页的“短寻道”。）

外部传输率是指可在驱动器附件连接器和 HAB 或 NIC 之间传输数据的速率。RAID 组的多个驱动器共享此带宽。请注意，制造商提供的猝发传输率不适合使用共享后端端口的存储系统。制造商提供的猝发速率是指内部连接，而且是允许直接访问驱动器系统内存的内部连接。对于存储系统的总线实施，实际传输率更多地是由后端传输协议、仲裁时间和后端端口的容量决定。

硬盘驱动器队列

硬盘驱动器一次执行一个 I/O。如果将多个 I/O 读或写请求发送至某个磁盘，这些请求将临时排入磁盘队列，供稍后执行。

驱动器队列是为了能高效使用驱动器。队列设置供持续使用。驱动器队列深度至少需为 2，驱动器才能高效运行。此外，还可能优化队列条目，从而提高 I/O 性能。未优化的驱动器按接收顺序执行队列上存储的请求。

当磁盘的队列大小达到 6（较慢驱动器）和 20（较快驱动器）之间时，则将达到其性能限制。发生这种情况是因为驱动器连接的带宽通常会超出驱动器传输率。随着队列的延长，可针对队列条目执行一些优化。但总的来说，磁盘队列深度是决定驱动器响应时间的重要因素。

磁盘花在寻道或等待扇区旋转到其磁头下方的都是浪费掉的时间，因为此期间没有传输数据。访问磁盘后续扇区上的数据时，无需在寻道方面浪费时间。读取或写入磁盘后续扇区的访问模式称为 *顺序访问*，顺序访问的服务时间极短。顺序读取的服务时间不足 1 ms，因此可处理的队列较之随机读取更长，后者的服务时间长达 4-6 ms。

硬盘驱动器的磁盘控制器可通过一些操作缩短响应时间。这些优化将通过优化 I/O 充分利用磁盘驱动器操作的顺序特性，以最大限度提高带宽。

SAS 驱动器上的命令标记队列 (CTQ) 利用电梯间算法等优化来提高随机 I/O 性能。CTQ 会对磁盘队列中的 I/O 请求重新排序，以最大限度降低重新定位读/写磁头的需求，从而实现顺序访问。与其他类型的驱动器相较而言，企业级驱动器可更高效地执行这些优化。

计算硬盘驱动器性能

共有四个用于比较驱动器的基本驱动器性能指标。这些指标根据驱动器规格计算得出，指标如下：

- ◆ 平均服务时间
- ◆ 平均响应时间
- ◆ 吞吐量
- ◆ 带宽

平均服务时间是指硬盘驱动器对数据提供服务所需的平均时间。平均响应时间是指可对主机应用程序提供服务的数据量。带宽和吞吐量是两种对可读取或写入的数据量的不同描述。它们视工作负载的 I/O 特性而定。

平均服务时间

平均服务时间测量磁盘用于执行单个 I/O 的平均时间。服务时间以毫秒为单位。平均服务时间的一般公式为：

$$\text{平均服务时间} = \text{平均寻道时间} + \text{平均延迟} + \text{传输时间}$$

实际上，VNX 对小型 I/O 的传输时间为零。

例如，对读取寻道时间为 3.6 ms 且写入寻道时间为 4.2 ms 的 10k rpm 2.5 英寸 300 GB SAS 硬盘驱动器计算平均服务时间：

- ◆ 写入寻道时间：4.2 ms
- ◆ 读取寻道时间：3.2 ms
- ◆ 平均延迟：3.0 ms（第 87 页上的表 8）
- ◆ 传输时间：0 ms

$$\text{平均服务时间} = 6.7 \text{ ms} = ((4.2 \text{ ms} + 3.2 \text{ ms}) / 2) + 3.0 \text{ ms} + 0 \text{ ms}$$

注意：寻道时间是平均服务时间的重要组件。在寻道密集的随机 I/O 工作负载中，此测量组件可提供重要的驱动器性能比较。在寻道最少的顺序 I/O 工作负载中，平均延迟值的差异对于比较驱动器性能更为重要。

平均响应时间

平均响应时间是指从请求排入队列到磁盘执行此请求的间隔持续时间。平均响应时间的简单计算如下：

$$\text{响应时间} = (\text{队列深度} + 1) * \text{平均服务时间}$$

例如，驱动器的平均响应时间为 6 毫秒，而队列平均有 6 个条目，其响应时间：

- ◆ 队列深度：6
- ◆ 服务时间：6 ms

$$\text{响应时间} = 42 \text{ ms.} = (6+1) * 6 \text{ ms}$$

吞吐量和带宽

吞吐量和带宽是两种对硬盘驱动器可在特定 I/O 模式下读取和写入的数据量的测量标准。

驱动器吞吐量以 IOPS 为单位。驱动器的每秒 I/O (IOPS) 数由寻道时间和平均延迟决定。驱动器的 IOPS 受到工作负载读写比率的影响。

用于计算随机 I/O IOPS 的一般公式为：

$$\text{IOPS} = (1 \text{ IO} / (\text{平均寻道时间} + \text{平均延迟})) * 1024 \text{ ms/s.}$$

带宽是指在支持的 IOPS 数条件下一秒内可传输的数据量。带宽等于 I/O 大小乘以 IOPS。

用 I/O 大小（以 KB 为单位）计算带宽的一般公式为：

$$\text{带宽} = (\text{I/O 大小} / 1024 \text{ KB/MB}) * \text{IOPS}$$

例如，在 3.5 英寸 7.2k rpm 2 TB NL-SAS 硬盘驱动器的工作负载中，写入寻道时间为 9.5 ms，读取寻道时间为 8.5 ms，而 8 KB I/O 操作的读写比率为 70:30，则此驱动器的吞吐量和带宽为：

- ◆ 写入寻道时间：9.5 ms
- ◆ 写入比：0.3
- ◆ 读取寻道时间：8.5 ms
- ◆ 读取比：0.7
- ◆ 平均延迟：4.2 ms（第 87 页上的表 8）
- ◆ I/O 大小：8 KB

$$\text{吞吐量} = 78.8 \text{ IOPS} = (1 / ((9.5 \text{ ms} * 0.3) + (8.5 \text{ ms} * 0.7) + 4.2 \text{ ms})) * 1024 \text{ ms/s}$$

$$\text{带宽} = 0.6 \text{ MB/s} = (8.0 \text{ KB} / 1024 \text{ KB/MB}) * 78.8 \text{ IOPS}$$

常见计算错误

网络和总线速度计算容易出错。正确率很不稳定。网络速度始终采用十进制表示法。容量表示法采用的却是二进制。例如，千兆以太网为 1000 Mb，而非 1024 Mb。

MHz 始终为 1,000,000/秒；而 MB/s 为 1,048,576 字节/秒。1 MB/s（兆字节/秒）等于 8 Mb/s（兆字节/秒）。要计算带宽，请用总线带宽（以字节为单位）乘以十进制速度，再除以十进制单位。

例如，以 100 MHz 运行的 64 位总线（八字节）的理论带宽为：

$$763 \text{ MB/s} ((8 * 100,000,000) / 1,048,576)，不是 800 \text{ MB/s}$$

硬盘驱动器速度和性能

机械硬盘驱动器的转速对性能有很大的影响。高 rpm 硬盘驱动器的速度将影响 I/O 延迟和存储系统缓存使用效率，如第 87 页的表 8 中所示。

表 8 磁盘轴 rpm 与延迟的关系

磁盘轴 rpm	平均延迟 (ms)
5,400	5.6
7,200	4.2
10,000	3.0
15,000	2.0

转速越快，I/O 延迟越低。高 rpm 驱动器对随机读取性能有直接的积极作用，但对随机写入的作用相对较低。

顺序读取和写入从较快转速获得的好处不如随机 I/O 明显。不过，它们确实可增加硬盘驱动器的顺序读/写带宽。

驱动器的转速会影响存储系统缓存刷新和填充的速率。缓存刷新期间，数据必须从写缓存快速写入驱动器（请参见第 100 页上的“[写缓存管理](#)”）。在缓存被绕过或禁用的情况下，驱动器越快越好，尤其是闪存驱动器。同样，通常在缓存中不被命中的随机读取会更快。好处是存储系统获得更高的整体 I/O 速率。

硬盘驱动器容量利用率和性能

机械驱动器容量会影响到与寻道时间相关的性能。

给定类型和速度的大容量驱动器可提供优于较小容量驱动器的性能。出现这种情况的原因：在数据量相同的前提下，大容量驱动器的磁道间寻道距离更短，因为每个磁道都有更多的数据。大容量驱动器的连续数据占用更少的相邻磁道。这将缩短访问时间。在小容量驱动器上，相同数据量占用的磁道数更多。最糟糕的是，小驱动器可能需要跨整个磁盘盘片半径进行寻道。

在闪存驱动器上，容量和性能之间也存在某种关系。对于访问，驱动器容量跨驱动器通道均匀分配。也就是说，每条通道都有着相同数量的 NVRAM 芯片。通道争用可能会降低驱动器性能。一般而言，通道数越多，每条通道的 NVRAM 芯片数越少，闪存驱动器的性能也就越高。

此外，为内部管理保留的驱动器容量百分比也会影响性能。驱动器专用于内部管理的容量百分比越高，性能也越高。

驱动器性能比较

表 9（第 88 页）汇总了随 VNX 系列提供的机械硬盘驱动器的特性。

表 9 驱动器性能因素 (VNX)

	闪存	SAS		NL-SAS
转速 (rpm)	N/A	15k	10k	7.2k
连接速度 (Gb/s)		6		
典型的平均寻道时间 (ms)*	0.03	3.1	3.7	9.0

*闪存驱动器的读/写

图 30（第 89 页）基于服务时间展示了不同类型与速度的硬盘驱动器代表性示例的相对性能。

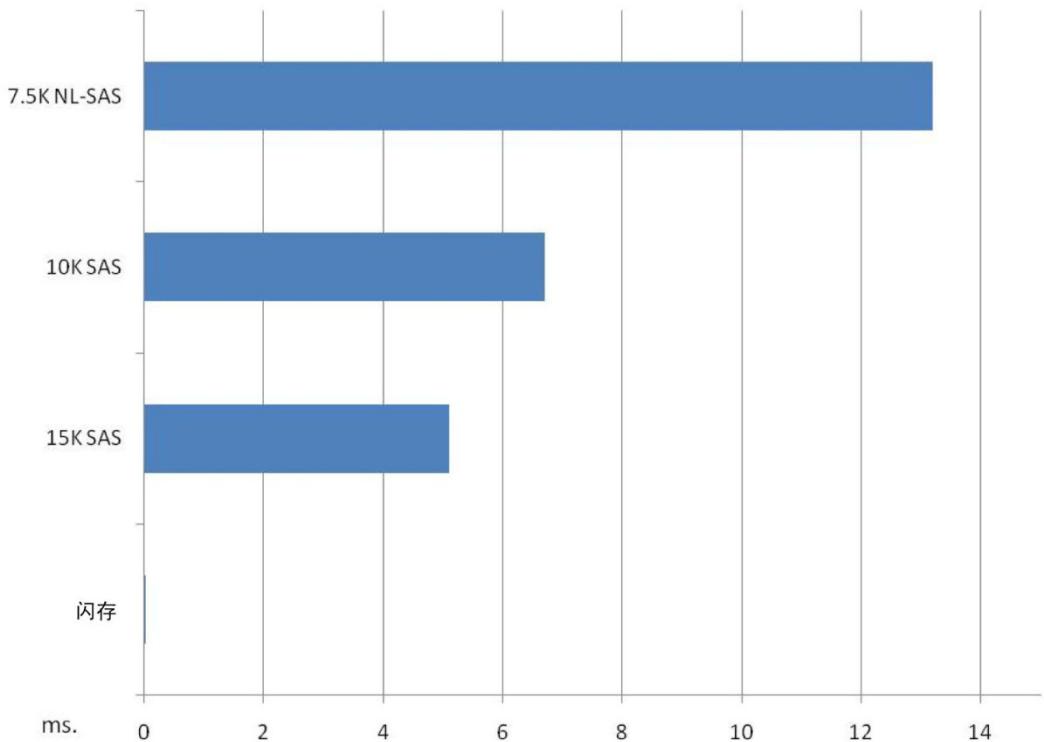


图 30 驱动器典型服务时间比较

在图中，您可看到硬盘驱动器转速对性能的影响最大。注意：闪存驱动器服务时间极短，只在图中的毫秒时间轴上隐约可见。闪存驱动器的极低读写时间赋予了它们最佳的性能。15k rpm SAS 驱动器具有机械硬盘驱动器中最短的服务时间。10k rpm SAS 驱动器的性能中等。较低 rpm NL-SAS 驱动器有着较为适中的性能。注意：服务时间差异对随机工作负载而言是重要的考虑因素，但是对顺序 I/O 工作负载的影响相对较低。

RAID 组性能

不同的 RAID 级别有着不同的性能功能。

分条是最影响 RAID 级别性能的一个因素。其余 RAID 级别性能差异仅在涉及存储系统缓存的特定情况下较为明显。

分条性能

在分条 RAID 级别的 I/O 中，主机 I/O 请求将根据 I/O 大小和位置细分为一个或多个到分条 RAID 组的独立磁盘的 I/O 请求。这些 I/O 有时称为 *后端 I/O* 或 *磁盘 I/O*。有关 RAID 实施如何执行后端 I/O 的特性对性能有一种可测量的影响。

RAID 1/0、RAID 0 和所有奇偶校验 RAID 级别都会分条。

通常，分条 RAID 级别的 *读取* 性能更胜于未分条 RAID 级别。这是因为它使用的后端带宽更少。奇偶校验型分条 RAID 提供的顺序写入带宽比镜像 RAID 类型更高。

磁盘分条会影响每个磁盘在对下一个 I/O 寻道之前传输的数据量。

分条有两个积极作用：跨多个驱动器分配随机 I/O，以及大型数据块并行访问。

通常，就具备随机 I/O 特性的任何给定工作负载而言，驱动器越多，性能越好。顺序 I/O 的性能提升则更为复杂。必须考虑对存储系统 RAID 操作的实施细节的依赖性。

高带宽（特别是大型数据块随机工作负载）视驱动器的 I/O 大小而定。与具有大量驱动器的组相较而言，驱动器较少的 RAID 组的单位驱动器带宽更高。这是因为大型 I/O 可发送到更少的驱动器。要实现最高带宽，请不要只用一个大型组，而是增加 RAID 组数量，而减少每个组的驱动器数量。这样也会减少线程争用，因为在小型 RAID 组上，给定 LUN 大小的 LUN 会更少。

条带元素大小

理想的条带元素大小可最大限度提高每个驱动器在单次 I/O 请求中传输的数据量。VNX 条带元素大小是一个固定值，此值为配合其硬件和软件体系结构进行了高度优化。条带元素大小为 64 KB。

条带大小

条带大小等于条带中的驱动器数乘以条带元素大小。

磁盘交叉（对齐）

单个条带元素无法容纳的 I/O 将跨越到第二个磁盘。这些称为 **磁盘交叉 I/O**。

（请参见第 90 页上的图 31。）磁盘交叉仅对基于机械驱动器的存储有不利影响。但是，磁盘交叉并不一定是有问题的迹象。超过条带元素大小的 I/O 跨越驱动器是正常的。如果 I/O 小于条带元素大小，磁盘交叉 I/O 可能会对性能造成一定的不利影响。

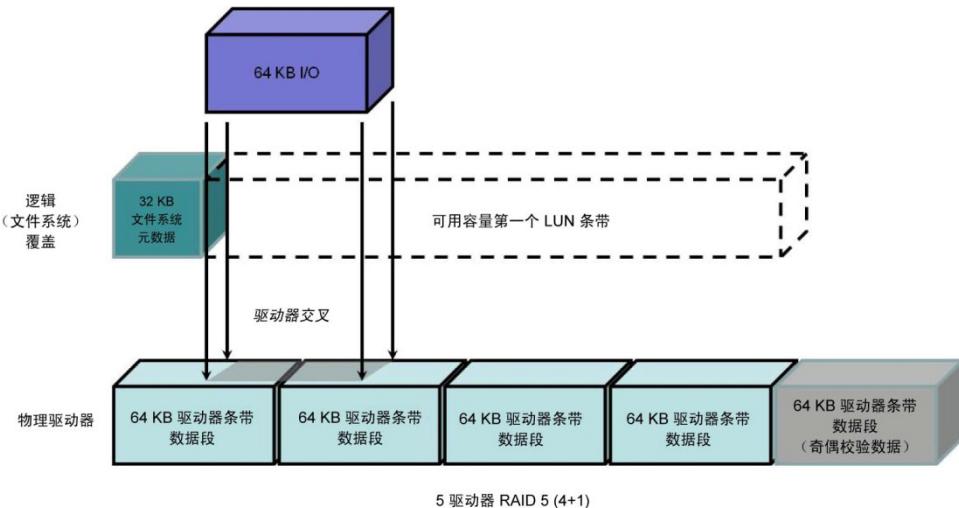


图 31 磁盘交叉，未对齐的 LUN

在磁盘交叉中，I/O 将跨两个驱动器进行拆分，而不像仅在一个驱动器上那样高效。如果跨两个驱动器拆分 I/O，将增加对驱动器和后端的负载。这将导致给定 I/O 速率的队列更长。更长的队列将会延长驱动器收到的任意 I/O 的响应时间。即使磁盘操作被缓存缓冲，拆分也可能有不利的影响，因为仍会使磁盘队列延长，进而影响到所有与磁盘相关的性能。大型数据块顺序 I/O 从对齐获得的好处最大；小型数据块随机 I/O 获得的好处则最少。本质上需要访问磁盘的随机读取也会受到直接（等待两个驱动器返回数据）和间接（使磁盘更忙）的影响。

务必注意：在使用默认的 64 KB 条带元素大小时，VNX 存储系统上任何大于 64 KB 的 I/O 都必须调用磁盘交叉。

例如，128 KB I/O 数据块大小必须跨越两个驱动器。

特定 I/O 数据块大小造成的额外 I/O 百分比的计算方式如下：

$$\text{磁盘交叉百分比} = (\text{I/O 数据块大小} / \text{条带元素大小}) * 100$$

例如，若是 8 KB I/O，预期的磁盘交叉百分比约为 13% ($(8 \text{ KB} / 64 \text{ KB}) * 100$)。注意：此百分比是不可避免的，应视为“正常”行为。

对齐可确保 I/O 更整齐地填入条带元素。这可提高 I/O 写入效率。文件系统元数据可能因不对齐而导致磁盘交叉。对齐的文件系统只需一个驱动器即可快速服务于 I/O，而未对齐的文件系统却需要两个驱动器才能满足 I/O 的一部分。

LUN 可使用文件系统应用工具进行对齐。（请参见第 91 页上的图 32。）在绑定 LUN 并将其呈现给主机之后，将执行对齐。对齐流程只能在没有数据的 LUN 上执行，而且不能是之前已分区的 LUN。

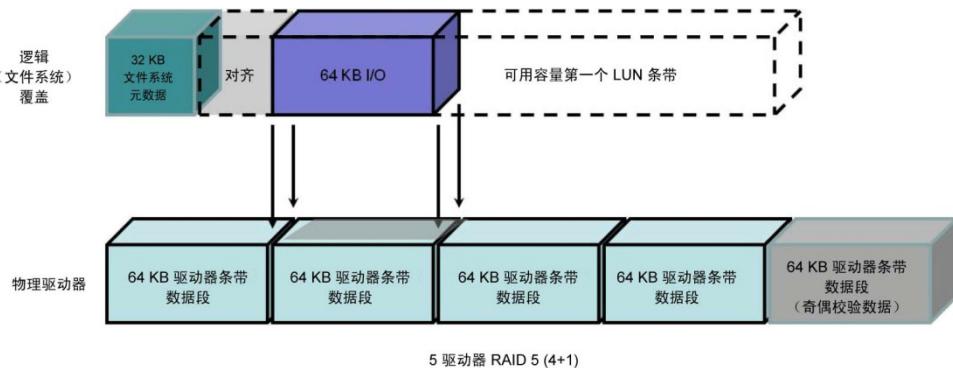


图 32 对齐的 LUN

MS Windows 2008 Server 卷将由主机自动对齐。诸如 Linux 等其他 O/S 可能需要手动对齐。注意：虚拟池密集 LUN 和传统 LUN 都可获益于对齐。但是，对齐对虚拟池精简 LUN 没有益处。

完整条带写入

执行常规缓存操作的存储系统将优化其驱动器 I/O，以尽可能执行完整条带写入。完全条带写入有时称为 MR3 写入。在完整条带写入中，不是传入 I/O 的大小足以填充条带并对齐，就是存储处理器将写入 I/O 累积于写缓存中。根据数据局部性，较小的 I/O 写入可合并为更少但更大的写入。写入前，可对各个写入请求排列顺序，直到缓存满整个条带为止。这样可更有效地使用后端端口和驱动器。

注意：如果在缓存绕过生效等情况下执行未缓存的 I/O，而且此 I/O 可完全填满条带，或者是条带宽度的整数倍，则将执行完整条带写入。但是，未缓存的话，将难以控制和确保完整条带写入。正因如此，要在禁用缓存时获得持续高的带宽十分困难。

缓存性能

在正常的完全缓存的操作情况下，不同 RAID 级别的性能基本都相同。存储系统的大型物理缓存和成熟缓存机制将带来最佳的 I/O 执行性能。此缓存将导致模糊 RAID 级别间的任何性能差异。

已满的写缓存发生刷新时，差异较大（请参见第 44 页上的“[内存](#)”）。通常，RAID 1/0 RAID 组在从写缓存刷新随机写入方面有着一定的优势。此优势也存在于正常操作中，但不明显。RAID 3、RAID 5 和 RAID 6 组则在从缓存刷新大型/顺序写入方面有优势。

未缓存的性能

在未缓存写入 I/O 或是禁用写缓存操作的罕见情况下，RAID 级别可影响性能。

用户禁用缓存时，可能发生未缓存的 I/O。对某些 LUN 禁用缓存可能是为了将缓存优先用于其他 LUN。I/O 也会根据 LUN 侧写设置绕过写入缓存。构成闪存驱动器的 LUN 可能禁用了写缓存，也可能没有。存储系统处于降级模式时，也可能禁用缓存。VNX 系列的高可用性缓存也使得这种状态非常少见。

有关未缓存 I/O 和禁用缓存的情况的信息，请参见第 45 页上的“[SP 缓存](#)”。

RAID 级别性能差异

此部分根据 I/O 类型和访问类型介绍了 RAID 级别之间的相对性能差异。注意：仅在未缓存 I/O 或缓存已满而“强制刷新”I/O 的情况下，才会观察到这些差异。

可从存储系统上的任何 DAE 或 DPE 位置选择要纳入 RAID 组的驱动器。RAID 组的所有磁盘必须属于相同的磁盘类型（NL-SAS、SAS 或闪存驱动器）。

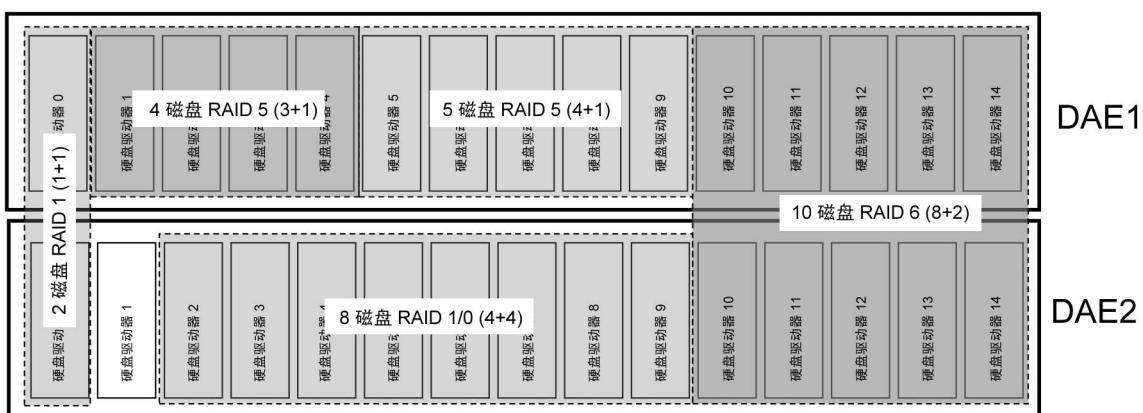


图 33 两个 15 驱动器 DAE 上绑定的概念 RAID 组

图 33（第 92 页）展示了两个 DAE 上创建的多个 RAID 组的概念视图。注意：RAID 组可跨越 DAE。

RAID 级别性能：奇偶校验与镜像

就各个 I/O 而言，奇偶校验与镜像 RAID 级别之间没有读取性能差异。但是，存在写入性能差异。

CPU 利用率

奇偶校验的计算将使用存储处理器 CPU 资源。镜像 RAID 级别不必计算奇偶校验。因此，使用镜像 RAID 时，存储处理器的利用率会稍低。

后端 I/O 操作

对 LUN 的读写请求不同于存储系统后端上的读写 I/O 操作。单个 LUN 读取或写入请求可能导致在后端端口上以及对 LUN RAID 组的驱动器执行一个或多个读取或写入 I/O 操作。

如果读取的大小超过条带元素大小，一个读取请求将生成一个或多个后端读取。

对存储的 I/O 操作。

单个写入请求可生成一个或多个 I/O 操作。存储系统后端上的写入请求生成的 I/O 数量是主要的 RAID 级别性能差异。

后端写入 I/O 的数量视 RAID 级别和缓存操作而定。对奇偶校验 RAID 级别的完整条带写入会执行一个后端写入 I/O 操作；此条带将被写入两次，一次是针对组中每个驱动器的数据写入，一次是针对组中的奇偶校验磁盘（请参见第 91 页上的“完整条带写入”）。对镜像 RAID 的完整条带写入也是对每个条带的后端写入，一个针对主要条带，另一个针对镜像条带。因为有更多的驱动器服务于任意大小的条带，所以会有更多的数据发送到后端。注意：就相同数量的用户驱动器而言，镜像 RAID 组较之奇偶校验 RAID 级别组在此情况下占用的后端端口带宽会更多。

随机 I/O 则不同。假设 I/O 小于条带元素，镜像 RAID 写入将生成两个后端写入 I/O — 镜像对的两个驱动器各一个。对奇偶校验 RAID 的随机写入请求将生成两个读取和两个写入 I/O 操作。

例如，未缓存的 RAID 5 的 8K 随机写入请求：

1. 从驱动器读取目标数据扇区。
2. 从为此条带保存奇偶校验的驱动器读取奇偶校验（新奇偶校验从此数据计算得出）。
3. 将含有数据的新扇区写入目标驱动器。
4. 将（新）奇偶校验写入为此条带保存奇偶校验的驱动器。

镜像 RAID 级别（RAID 1 和 RAID 1/0）对小型数据块写入和随机写入的执行性能更好。对于随机写入，奇偶校验 RAID 级别（RAID 3、RAID 5 和 RAID 6）需要存储处理器来读取、计算和写入每个写入操作的奇偶校验扇区。RAID 5 和 RAID 3 需要执行一次奇偶校验计算。RAID 6 需要执行两次奇偶校验扇区计算。执行此计算所需的时间和资源将会严重影响随机写入期间的奇偶校验 RAID 级别性能。

奇偶校验 RAID 级别对大型数据块写入和顺序写入的执行性能更好。对于顺序写入，只要有可能执行完整条带写入，写入较少的奇偶校验 RAID 的性能就会比镜像 RAID 更好。这是因为写入的冗余数据更少：一个奇偶校验元素相当于镜像级别 RAID 的两个完整条带。

镜像 RAID 级别（RAID 1 和 RAID 1/0）

RAID 1

RAID 1 可基于每个驱动器提供最佳带宽。随机写入性能比大多数 RAID 级别都要高，但它受限于两个驱动器的最大 RAID 组级别大小。顺序读取性能与单个驱动器相同。

在 RAID 1 中，读取是一个后端操作，而写入是两个后端操作。

RAID 1/0

RAID 1/0 可通过分条提升性能。RAID 1/0 包括一些读取优化，可借此充分利用两个数据相同的驱动器。随机读写性能的表现都非常突出。而且，顺序读写性能也很棒。

RAID 1/0 组可包含 2-16 个驱动器。同有两个驱动器的 RAID 1/0 与 RAID 1 RAID 组的性能旗鼓相当。

在 RAID 1/0 中，读取是一个后端操作，而写入是两个后端操作。

奇偶校验 RAID 级别（RAID 3、5 和 6）

RAID 3

RAID 3 可通过分条获得不错的随机读取性能，但要稍逊于 RAID 5。不会从奇偶校验驱动器中进行读取。对随机写入性能造成影响的因素包括奇偶校验计算及单个专用奇偶校验磁盘可能遇到的瓶颈。顺序读取性能同样可通过分条获益。顺序写入性能非常出色，因为 RAID 3 的代码路径是所有奇偶校验 RAID 类型中最短、最简单的。

在 RAID 3 中，读取是一个后端操作；而写入可能是一个后端操作（完整条带），也可能是四个后端操作（不足一个条带）。

RAID 5

RAID 5 具有绝佳的随机读取性能。增加 RAID 组中的磁盘数即会提高性能。随机写入性能因奇偶校验计算而略显平庸。RAID 5 随机写入性能要优于 RAID 3，因为没有专用奇偶校验磁盘，因而不会遇到瓶颈。顺序读取性能佳。顺序写入性能佳。

在 RAID 5 中，读取是一个后端操作；而写入可能是一个后端操作（已缓存的完整条带），也可能是四个后端操作（不足一个条带的单一小型 I/O）。

RAID 6

RAID 6 与 RAID 5 有着类似的性能。相较而言，RAID 6 会需要额外的奇偶校验计算。在所有 RAID 级别中，它的随机写入性能（相等的用户数据驱动器计数）最低，但有着绝佳的随机读取性能。顺序读取性能佳。性能可通过较小条带宽度来提升。顺序写入性能尚可。

在 RAID 6 中，读取是一个后端操作；而写入可能是一个后端操作（已缓存的完整条带），也可能是六个后端操作（不足一个条带的单一小型 I/O）。

RAID 0

RAID 0 较为特殊。它提供分条 RAID 的所有性能优势，也没有奇偶校验计算的计算开销。但是，RAID 0 却缺乏内置的数据保护。

RAID 0 拥有绝佳的随机读写性能，尤其是针对大型条带。顺序读写也有相当不俗的表现。

在 RAID 0 中，读取是一个后端操作；而写入对于完整条带或不足一个条带都是一个后端操作。

RAID 组性能计算

RAID 组的吞吐量或带宽视 RAID 组的资源调配及其执行的 I/O 而定。

吞吐量估计

RAID 组的可用后端 IOPS 及带宽视 RAID 组的以下因素而定：

- ◆ 驱动器数量
- ◆ 每个驱动器型号的 IOPS 或 MB/s 能力

估计的最大性能等于 IOPS 或带宽乘以 RAID 组中的硬盘驱动器数。

假设单个 15k rpm SAS 硬盘驱动器的 IOPS 约为 180。对于配有 5 个驱动器的 RAID 5 (4+1)，用户数据将跨这五个驱动器分散。此 RAID 组的潜在吞吐量为 900 IOPS (180×5)。

对于镜像型 RAID，读取期间读取的可能是任意条带（主要或镜像）。读取经过优化后，将以最快的访问速度使用条带。写入期间，主要条带和镜像条带都会被写入。因此，IOPS 要乘以二。然而，实际需要在计算中使用的只有主要条带的 IOPS。

您不应基于此类计算进行最终性能规划。在不确定 I/O 特性的情况下，无法准确计算 RAID 组的性能。

吞吐量计算

I/O 特性对 RAID 组性能有着重要影响。若要准确地估计 RAID 组的带宽、吞吐量及响应时间，需综合以下因素：

- ◆ I/O 类型
- ◆ 访问类型
- ◆ I/O 大小

I/O 类型（随机或顺序）、读写比率及 I/O 大小全都会影响性能（相关详情，请参见第 26 页上的“[I/O 特性和工作负载](#)”）。

有关用 RAID 组特性和 I/O 类型计算 RAID 组性能的具体方法及执行这些计算的信息，请参见《EMC Unified Best Practices for Performance and Availability: Common Platform and Block OE XX — Applied Best Practices》（EMC Unified 性能和可用性最佳做法：通用平台与 Block OE XX — 适用的最佳做法）白皮书。

LUN 性能

LUN 性能视底层 RAID 组的性能而定。

在具有最大驱动器数的池或具有最大驱动器数的传统 RAID 组中创建虚拟池 LUN，是确保对小型随机 IOPS 的高性能 LUN 的最简便方法 — 驱动器越多，性能越高。在其当前虚拟池或传统 RAID 组中增加 LUN 容量不会提高性能。

在具有多个密集或精简 LUN 的虚拟池中，或者已分区 RAID 组的传统 LUN 中，LUN 将共享底层 RAID 组的 IOPS 和带宽。也就是说，如果各应用程序分别使用同一 RAID 组上创建的一个 LUN，这些应用程序的组合需求可能超出 RAID 组的性能能力。

争用是指绑定到同一 RAID 组的两个或两个以上 LUN 的 I/O 相互影响性能。
避免 **链接争用**和**驱动器争用**。链接争用是指对不同 LUN 的 I/O 迫使 RAID 组的驱动器磁头在 LUN 之间大量移动。驱动器争用是指多个 I/O 流需要同时访问同一驱动器。

建议您限制每个池或 RAID 组的 LUN 数量，因为随着 LUN 数量的增加，要不用耗时的分析和调试就预测或确定 I/O 是补充还是争用会越来越困难。

务必记住：驱动器越多，性能越好。

短寻道

短寻道是对由机械硬盘驱动器构成的 RAID 组上的 LUN 增加吞吐量的资源调配方法。

每当要针对寻道移动读/写磁头时，传统硬盘驱动器都会造成额外的 I/O 服务时间。短寻道可通过限制驱动器上要使用的磁道数，缩短这些磁头的重新定位延迟。驱动器上最先使用的是最外层磁道。与内层磁道相较而言，最外层磁道每单位长度的扇区更多。通过限制磁道数，我们将缩短每个 I/O 的服务时间。这样会直接影响到各个磁盘及整个 RAID 组的响应时间。但是，通过限制可访问磁道数来实现此性能收益，将会降低 RAID 组的可用容量。

具有众多线程的小型数据块随机 I/O 工作负载从短寻道获得的好处最大；大型数据块顺序工作负载的则最少。短寻道的 RAID 组可通过在 RAID 组上创建单一传统 LUN 来设置。这样会将此 LUN 定位于 RAID 组驱动器的**最外层磁道（快速磁道）**上。短寻道 LUN 相对于 RAID 组总体容量的容量越小，性能提升越高。

在理想情况下，仅当使用短寻道时才应将单一传统 LUN 绑定到 RAID 组，以免造成可能的 LUN 争用。注意：短寻道不适用于虚拟池。此外，闪存驱动器的问世也让短寻道成了过时的方法，除了作为性能基准。

第 10 章 存储系统性能

本章包括以下主题：

百分比利用率	98
共享的 RAID 组	105
后端总线性能测量	106

要实现最佳存储系统性能，需平衡存储系统的两个存储处理器之间的工作负载。工作负载平衡涉及 LUN 数、分配给每个存储处理的主机数及这些主机引导至这些 LUN 的 I/O 数。平衡工作负载需要了解存储系统中的以下相关子系统：

- ◆ 前端 — 包括将存储系统连接到主机的前端 I/O 端口。
- ◆ 存储处理器 (SP) — 包括存储系统的 CPU 和内存。内存包括重要的存储系统缓存。
- ◆ 后端 — 包括将存储处理器连接到构成 LUN 的设备的 I/O 端口。

目的是为了避免这些子系统出现任何瓶颈。瓶颈是指可能限制从驱动器到前端端口的数据流的情况。

百分比利用率

百分比利用率是资源繁忙程度的一个测量标准。它用于确定系统的性能功能。高利用率不一定就是坏事。诸如 LUN 之类的逻辑设备没有物理限制。存储处理器、前端端口、驱动器和后端端口之类的物理设备则不然。对于这些物理设备，高利用率可能意味着需要最大吞吐量或 IOPS。相反，如果目标是低主机响应时间，则需寻求低利用率。这是因为物理设备的高利用率伴有更长队列及由此而来的更高响应时间。

利用率是一段时间内得出的平均值。在任何时刻，设备都处于繁忙或闲置状态。此测量标准的计算方法是用设备的繁忙时间除以总的已用时间。

例如，如果硬盘驱动器在 1 秒的间隔内有 300 ms 在接收和服务于 I/O 请求，则利用率为 $300 \text{ ms} / 1000 \text{ ms} = 0.30$ 或 30%。

如果收到的请求足以让设备或系统没有闲置时间，则表示利用率为 100%。物理资源的 100% 利用率不可取。这种利用率下可能出现难以察觉的超额预订。资源的超额预订可能不容易检测到，因为指标并不明确。

请注意在确定逻辑与物理存储对象利用率方面的差异。物理存储对象的性能易于量化。硬件可提供准确的计算指标。创建含有物理存储设备的逻辑存储设备将引入众多依赖项，有些依赖项很明显，有些则不然。

例如，如果物理存储对象已达到 100% 利用率，但还接收了另外 50% 的请求，则利用率仍为 100%。这表示资源严重地被超额预订。但是，LUN 之类的逻辑存储对象即使达到 100% 利用率也可能仍有一些可用性能，具体视工作负载及提供支持的物理驱动器而定。

利用率保留

从规划角度来看，高利用率的系统可能达到性能极限。如果系统进入降级模式或面临工作负载高峰，这会让它们没有任何性能余量。低利用率的系统拥有更多的开销来应对猝发或故障。

最好不要同时让冗余资源的利用率长时间超过 70%。保留 50% 是较为保守的想法。这是在降级模式下支持操作所必需的。在全局或局部故障的情况下，需要有足够的性能余量来完全承担故障的资源负载，而无明显的性能降级。

高峰是指 I/O 或 CPU 利用等活动的不可预测的突然急增。要应对高峰，需要保留一定余量的存储系统性能资源。否则，如果在繁忙期间出现高峰，用户响应时间可能会受到严重影响。此保留包括未提交的 SP 利用能力、I/O 带宽和存储容量。此保留应经过合理的调整，以处理高峰期的“最迫切”需求。

存储系统资源利用率

存储系统有三种需密切监视其百分比利用率的物理资源。这些资源全都与工作负载密切相关。它们细分为存储系统的 CPU、内存和 I/O 利用率。资源包括：

- ◆ 存储处理器
- ◆ 缓存
- ◆ 存储

存储处理器利用率可衡量存储处理器的 CPU 的繁忙程度。存储处理器利用率将影响存储处理器可处理的任务数及其执行速度。

缓存利用率是一个内存利用率指标。缓存利用率直接影响存储系统对 I/O 的实时处理能力。

存储利用率包括后端总线利用率和存储设备利用率。若不借助分析工具，将无法直接测量后端总线利用率；此值必须从磁盘利用率推算出来。如果磁盘利用率长时间超过 70%，则表示存在性能问题。

存储处理器

为确保高性能和高可用性，存储系统设计应以大约 50% 的存储处理器利用率为目。但是，可允许一个或两个存储处理器的 CPU 利用率在短时间内达到 70% 之高。最好不要同时让两个存储处理器的利用率长时间超过 70%。这是因为在存储处理器发生全局或局部故障时，存储处理器将没有足够的性能余量，以在不明显降低性能的前提下完全接管其对等存储处理器的负载。

例如，如果两个存储处理的利用率都超过 70%，万一发生存储处理器故障，存储处理器将没有足够的性能余量接管其对等存储处理器的负载。一旦发生大规模 LUN 主动变更，对等 SP 的额外负载可能导致主机响应时间长得令人无法接受。

随着多核 CPU 的问世，了解存储处理器利用率指标变得尤为重要。在多核 CPU 中，每个核心在一定程度上都具备独立的执行能力。每个 CPU 核心的利用率可能各不相同。各个核心的利用率可能由于执行个别功能（例如，奇偶校验计算、LUN 压缩/解压缩和 FAST VP 重新定位）而升高。多核 CPU 的平均 CPU 利用率代表不了存储处理器的真实状态。

例如，就四核存储处理器 CPU 而言，各个核心的利用率可能分别为 10%、100%、50% 和 20%。因此，平均存储处理器利用率为 45%。乍一看，存储处理器利用率很好地处于限制以内。但是，有一个核心的利用率却达到了 100%。它将无法执行其他功能。其完全利用率是一个潜在的瓶颈，但这无法从指标看出。

读/写缓存

读缓存优化

就顺序读取请求而言，缓存命中几率可通过预取（也称“预读取”）方法获得大幅提升。在此，当 SP 检测到顺序访问时，将在预计稍后要请求的实际主机请求之前预先填充读缓存。

得益于预读取，顺序 I/O 工作负载具有高缓存命中率。这将大幅缩短响应时间。此方法对随机 I/O 工作负载没有作用。

读缓存命中率

读缓存命中率 (RCHR) 是指读缓存中找到的读取 I/O 请求的百分比。它是用于确定读取 I/O 工作负载是随机型还是顺序型的其中一个指标。

如果工作负载的读取 I/O 全都是随机的，RCHR 最多可达 5%。如果此值更高，测量对象可能是随机与顺序 I/O 的混合工作负载，或者是多线程顺序工作负载。也有可能是重复访问了相同地址，或者 I/O 具有高局部性。

例如，文件系统卷的第一个 64 KB 上的元数据是通常接收重复访问的地址集。

对于纯顺序 I/O，此指标的值视执行 I/O 的线程数而定。对于单线程的顺序 I/O，RCHR 应趋近 100% 且不低于 80%。这是预取以峰值效率工作的示例。

多流多线程的顺序访问可能降低预取的有效性，因此 RCHR 稍低。多流多线程的顺序 I/O 可能导致大量读取 RAID 组驱动器的众多不同区域。这样将降低预取流程的速度，从而使 RCHR 变低。RCHR 处于范围下限时，可通过修改预取参数来获得提升。但是，需要进行深入读缓存使用情况分析，以确定较高 RCHR 的缓存参数设置。RCHR 值偏低的最后一个可能原因是工作负载不是真正的顺序工作负载。工作负载中的随机 I/O 组成部分越大，RCHR 值越低。

刷新概述

刷新是指将写缓存页写出到存储。缓存算法将刷新这些页，以确保有充足的缓存余量来处理猝发事件。在理想情况下，主机的写入 I/O 数及频率应与存储系统将数据刷新到存储的能力实现平衡。此类平衡源于合理的性能规划。

写缓存优化

与缓存读取 I/O 相较而言，缓存写入 I/O 对内存的利用效率更高。

顺序写入一经检测到即会获得优化。多个小型 I/O 写入可合并为更少但更大的存储写入。在理想情况下，它们可合并为一个或多个完整条带写入。此外，对缓存中相同位置的写入将被最后一次写入取代。这样，对存储系统后端资源的利用会非常高效。

回填是一种从存储读取数据的写缓存优化，它可实现更高效的写入 I/O 大小。例如，在基于 RAID 5 (4+1) RAID 组的 LUN 中，如果有一个挂起的 192 KB 写入，可读取目标 LUN 中相邻的 64 KB，以实现更理想的 256 KB 完整条带写入。

存储系统还有一种将数据直接写入磁盘的选项。此方法称为侧写，又名直写。侧写可在 LUN 级别通过 Unisphere 进行调整。

侧写允许大型 I/O 绕过已启用的缓存。I/O 将直接进入存储设备。这样可防止超大型写入 I/O 占用过多的缓存页。失去这些页将会严重影响整个工作负载的写缓存带宽。绕过 I/O 会由于未缓存而需要稍长的响应时间。但是，由此换来的却是大型 I/O 的处理效率收益。

写缓存管理

适当地调整缓存以支持它的调配的驱动器和端口数量，是一项重要的调整任务。在此调整中，存储系统按顺序执行写缓存页刷新及读缓存预取的能力比主要缓存容量更为重要。

刷新详细信息

刷新有三种类型：

- ◆ 闲置刷新
- ◆ 高水位刷新
- ◆ 强制刷新

闲置刷新是指向闲置 LUN 执行挂起的 I/O。当 LUN 持续两秒以上没有 I/O 时，将会进入闲置状态。闲置刷新是定期运行的后台进程。一旦 LUN 进入闲置状态，其页将从缓存进行刷新，为活动 LUN 的 I/O 腾出空间。如果没有闲置的 LUN，或者 LUN 之前经过刷新后已没有脏页，则不会执行闲置刷新。

脏页百分比达到预设限制时，将激活高水位刷新。此限制称为高水位线。设置高水位线通常是为了维护缓存中的保留容量，此容量用于处理写入猝发。触发高水位刷新时，SP 会将缓存刷新速率增加到闲置刷新的刷新速率以上。高水位刷新对 SP 性能的影响最小。注意：如果脏页百分比长时间保持在 90%，则表示存在性能问题，不过可通过调整高水位线来纠正此问题。

写缓存将尝试让内存随时可供传入的写入使用。应避免强制刷新，不过即使在合理设计的系统中也偶尔会发生这种刷新。如果在收到写入 I/O 请求时缓存已满，即会触发强制刷新，以写入接收当前请求的目标 LUN 的页并清空页，供以后的请求使用。新写入请求将接收新清空的页上的空间。强制刷新可连续触发多次，直到缓存提供的页足以容纳传入的写入请求为止。一个 LUN 耗尽其可用页时，偶尔会发生强制刷新。在合理调整的存储系统上，针对所有 LUN 的强制刷新应极为罕见。

强制刷新表示主机等待物理磁盘操作和完全 RAID 操作，因此主机响应时间会延长。在存储系统执行繁重的刷新操作期间，性能会受到严重影响，因为将缓存内容写入驱动器会与收到的其他 I/O 请求争用资源。

若使用基于闪存的存储（例如，基于 FAST 缓存和闪存的 LUN 及 15k rpm 机械硬盘驱动器），将最大限度降低强制刷新的影响。这些设备较短的服务时间允许在短时间内刷新更多的页。

缓存启用/禁用

在某些情况下，可能不需要缓存。因此，可适时禁用缓存。

缓存可在如下两个级别启用或禁用：存储处理器和 LUN。存储处理器是较高级别。如果在存储处理器处禁用缓存，LUN 级别将没有缓存。如果在存储处理器级别启用缓存，个别 LUN 可自行启用或禁用缓存。

VNX 系列具有非常强劲的缓存。在大多数使用缓存的操作中，它都是以完全缓存的状态运行。仅当未满足下列条件时，才会自动禁用存储处理器缓存：

- ◆ 至少有一个充满电的 SPS（仅禁用写缓存）。
- ◆ DPE/SPE 中至少有一个正常运转的电源。（某些型号可应付三个 PS 故障）。在 SPE 型号中，两个 DAE O/S 电源都正常运转。

存储系统的缓存通过 Unisphere 进行控制。

LUN 缓存

缓存通常在 LUN 级别进行配置。写缓存页将根据 LUN 遇到的 I/O 数从可用页进行动态分配。繁忙 LUN 将获配大部分页；无活动的 LUN 不会获配页。有大量繁忙 LUN 时，缓存算法将适度调整 LUN 接收的页数。

LUN 可以具有任何已启用或禁用的读写缓存组合。默认情况下启用了 LUN 缓存。基于闪存的 LUN 则不同。这些 LUN 的缓存视操作环境版本而定。对于 OE Block 31.0，缓存默认是禁用的。

在对存储处理器缓存造成负担的情况下，需手动禁用 LUN 缓存以保留缓存页。例如，当运行多个生成大型 I/O 的应用程序时。又比方说，在运行多个完全 SAN Copy 会话之前禁用缓存。（SAN Copy 是一种大批量数据复制分层的应用程序；请参见第 30 页上的“[复制分层的应用程序](#)”。）

缓存性能

存储系统缓存具有高可配置性。对于大多数工作负载，默认设置都能很好地适应。但是，对于非典型的工作负载，或要实现最高级别的性能，可通过调整下列几个参数来调整缓存：

- ◆ 缓存页大小（全局）
- ◆ 预取（本地）
- ◆ 水位线（全局）
- ◆ 侧写大小（本地）

注意：某些参数是本地参数，适用于单个 LUN；某些参数是全局参数，适用于所有 LUN。

缓存页大小

缓存页大小适用于读写缓存。缓存页具有固定的大小（以 KB 为单位），可能的值包括 2、4、8 或 16 KB。读写缓存都使用相同的页大小。8 KB 页大小是默认值。

在理想情况下，缓存页大小与存储系统接收的最常见的 I/O 写入请求大小一致。对于主要执行大型或顺序 I/O 的系统，较大的缓存页大小可提高整体系统性能。但若混合 I/O 请求大小，默认大小最佳。

预取

预取仅适用于读取缓存。存储系统对读取 I/O 请求提供持续或可变预取。

持续预取对于密切控制预取数据量以避免填满读缓存很有帮助。使用持续预取时，若检测到顺序访问，将在读取 I/O 请求之前预取相同数量的 KB。缓存算法将刷新页，并且维护存储系统的效率，同时保留充足的缓存余量应对猝发行。持续预取最适合 I/O 大小统一的情况。

可变预取结合使用读取 I/O 请求大小、段乘数及预取乘数来发挥作用。可变预取是默认值。

- ◆ 读取 I/O 请求大小：是指主机中的读取请求大小（以 KB 为单位）。
- ◆ 预取乘数：用于计算预取的总大小。
- ◆ 段乘数：确定对后端的 I/O 大小。

如果读取请求大小小于缓存页，将使用缓存页大小。预取乘数用于计算此请求中要预取的总量（预取乘数 * 读取请求大小）；然后使用段乘数（段乘数 * 读取请求大小）计算可满足整个请求的后端 I/O 大小。

如果段乘数和预取乘数相等，将对后端执行单个请求。如果段乘数小于预取乘数，将对存储执行多个请求。段乘数绝不会大于预取乘数。

例如，假定：

- ◆ 缓存页大小：4 KB
- ◆ 读取 I/O 请求大小：8 KB
- ◆ 预取乘数：4
- ◆ 段乘数：2

读取 I/O 请求大小超过缓存页大小。将使用请求大小。预取总量为 32 KB ($4 * 8 \text{ KB}$)。此 32 KB 将分为两次 16 KB ($2 * 8 \text{ KB}$) 请求从存储中读取。

对于小型 I/O 大小，将段乘数和预取乘数设置为同样大的数字可提高性能，因为后端的工作效率更高。读缓存的大小足以保存结果时，此设置在顺序读取期间特别有效。对于大型 I/O，将这两个乘数设为相同可能导致预取占用宝贵的后端带宽。在生产环境中，将段乘数设置为预取乘数的一半或四分之一可拆分预取序列，以允许执行存储系统上的其他 I/O。段乘数越小，读取吞吐量越低。

LUN 预取设置

可对预取应用限制参数以防止存储请求过多，因为这样会严重影响整体性能。这些设置在 LUN 级别单独设置。它们是：

- ◆ 最大预取（数据块）
- ◆ 预取禁用大小（数据块）
- ◆ 预取闲置计数。

最大预取设置是对所请求数据的限制（以数据块为单位）。这是在实际请求之前可预先填入读缓存的最大数据块数。例如，若将最大预取设置为 4096，则会将预取限制为 2 MB ($512 \text{ 字节} * 4096$)。

预取禁用大小将阻止禁用大小或以上的 I/O 触发预取。这会导致此大小或以上的 I/O 直接从未缓存的磁盘中读取。

预取闲置计数将限制对极为繁忙的 LUN 执行预取。预取闲置计数也用于缓存刷新；仅应在 EMC 服务人员的指导下更改此参数。

水位线

水位线用于管理写缓存刷新。存储系统具有两个水位线：高和低。这些参数共同管理刷新情况。水位线仅在激活写缓存时适用。

高水位线以上的缓存容量表示保留的页可处理写入 I/O 猝发。超过高水位线时，将启动并持续执行水位线刷新，直到达到低水位线为止。再次达到高水位线之前，水位线刷新将会停止。

设置高水位线以上的缓存余量是为了应对写入 I/O 猝发并且防止强制刷新。低水位线设置的是缓存中维护的最小写入数据量。这也是存储系统停止高水位或强制刷新的设定点。低水位线以下的缓存量是确保正常情况下高缓存命中数所需的最小缓存页数。系统不忙时，此水位线可能因闲置刷新而下降。

对于数据块使用情况，存储系统的默认低水位线是 60%。默认的高水位线为 80%。对于文件使用情况，存储系统默认值可能不同。

高水位线和低水位线之间的差异可确保刷新活动的速率和持续时间。差异越大，水位线刷新频率越低。超过高水位线时，刷新活动最频繁。密集刷新将增加存储中的 LBA 排序、合并和并发性，但可能对读取 I/O 有着不利影响。水位线之间的差异越小，刷新活动越稳定。较低的刷新密度允许执行其他 I/O（特别是读取）。

在猝发工作负载环境中，降低这两个水位线将增加缓存“保留”的可用页数，允许系统不执行强制刷新即吸收写入请求猝发。

图 34（第 104 页）展示了高低水位线如何共同管理写入缓存：

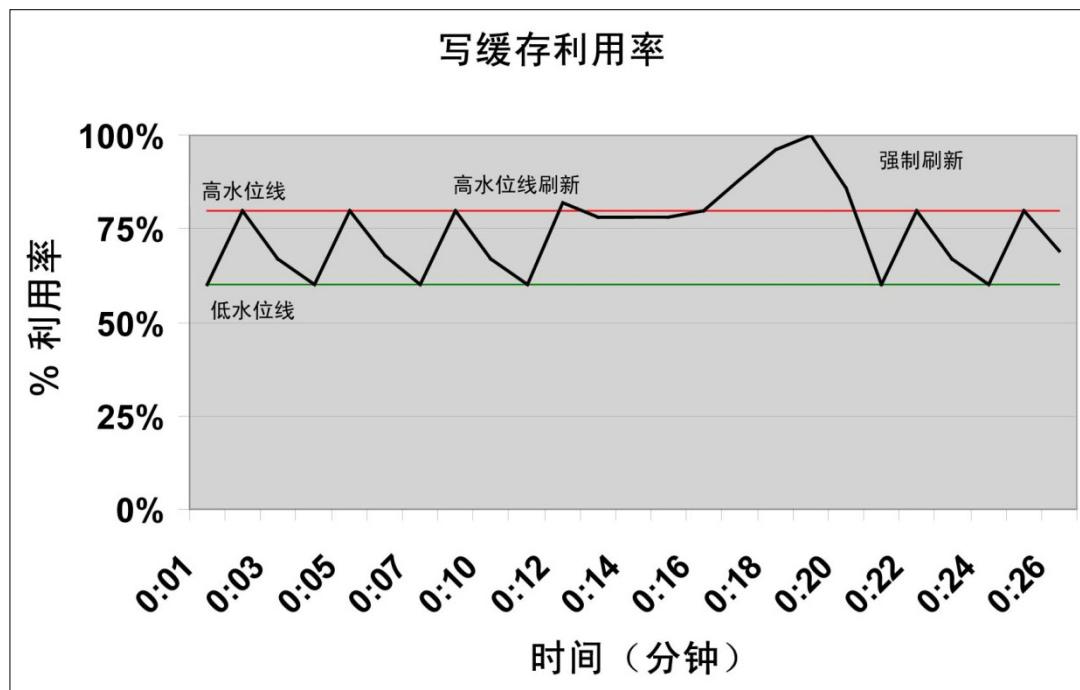


图 34 写缓存利用率

在第 104 页上的图 34 中，黑线表示缓存利用率或脏页百分比。展示的工作负载活动将缓存填满到高水位线（红线）。随后，通过高水位刷新使缓存降至低水位线（绿线），这样允许再次填满缓存。再次超过高水位线之前，没有发生刷新。

随着负载的增加（图中 12 分钟刻度处），由于工作负载对缓存页的使用及刷新速率达到平衡，因此缓存使用趋于稳定。如果有一次 I/O 猝发填满缓存并且超出高水位刷新将页写出到存储的能力，将发生强制刷新。（17 分钟刻度处展示的正是此情况。）强制刷新和高水位刷新的组合使新页随时可供使用。猝发结束时，强制刷新和高水位刷新使缓存使用降至低水位线。

设置水位线的目的是在最大限度提高写缓存命中数的同时避免强制刷新。如果强制刷新长时间超过 20/s，则表示存在性能问题。使用水位线，可通过调整缓存延长响应时间，同时维护保留的缓存页数，以满足工作负载猝发和系统维护的要求。

侧写大小

侧写大小参数将设置可缓存的最大 I/O。注意：I/O 需要超过此参数才会绕过缓存。侧写在 LUN 级别单独设置。只能通过 CLI 更改侧写。

大型写入会由于写缓存镜像而主要影响写缓存带宽。对侧写配置的大小应设得足够大，以免遇到此瓶颈。

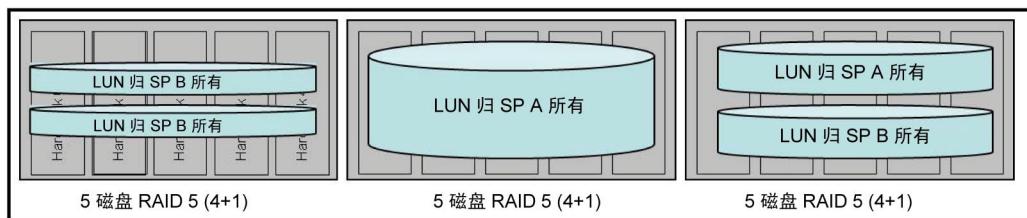
侧写大小参数以数据块为单位。例如，4096 个数据块导致执行 2 MB 侧写（512 字节 * 4096）。

默认侧写为 1 MB。也就是说，1 MB 的写入请求将会缓存，而 1 MB 以上的写入请求则会进行直写（直接写入存储）。

如果侧写设置过高带宽写入应用程序发来的 I/O，可能导致缓存被快速填满，进而发生强制刷新。偏低的侧写设置会增加超过侧写大小的 I/O 的响应时间。如果目标 LUN 属于奇偶校验 RAID 类型，绕过缓存的 I/O 的大小应足以填满奇偶校验条带，以便不会执行奇偶校验操作。

共享的 RAID 组

驱动器是存储处理器之间的共享资源。RAID 组可能包含一个或多个 LUN。RAID 组承载的 LUN 可能属于任一存储处理器。两个存储处理器可同时拥有 RAID 组上的独立 LUN。



调配时分区为三个 5 驱动器 (4+1) RAID 组的 15 驱动器 DAE

图 35

共享的 RAID 组驱动器

最简单的组织是由单个存储处理器独立使用 RAID 组（如第 105 页上的图 35 中间所示）。也就是说，RAID 组的 LUN 属于单个 SP。在存储处理器之间共享 RAID 组，将允许驱动器拥有更长的队列。单个存储处理器限制给它可发送至驱动器的请求数使用。对驱动器的双存储处理器访问是一种从驱动器获得并发性和实现最大吞吐率的方法。更长磁盘队列的不足之处是稍微延长响应时间。

虚拟池出于固有特性，会让 LUN 共享它们的专用 RAID 组。

后端总线性能测量

最重要的后端性能指标是驱动器带宽。（带宽很可能超出 IOPS 限制。）从驱动器接收的带宽暗示了后端端口的带宽使用。

可根据从硬盘驱动器接收的带宽来诊断后端瓶颈。使用 Unisphere Analyzer，此信息可在 RAID 组的总带宽指标中找到。注意：RAID 组可跨越两个或以上后端端口。此类资源调配会使分析更为复杂。

通常，整个后端的 I/O 需求应尽可能均匀地跨所有端口分布。默认情况下，Unisphere 会尽可能均匀地跨后端端口分布 RAID 组。按照循环调度顺序，每个 RAID 组都调配于一个单独的端口，所有成员驱动器位于同一总线。注意：此自动分配不考虑创建的 LUN 以及这些 LUN 上收到的 I/O。

总线性能影响

横向资源调配是指将 RAID 组的所有驱动器放置于单一后端端口上的做法。对此的最常见实现方式是创建一个完全包括在单个 DAE 中的 RAID 组，不过，跨越同一后端端口分布数个 DAE 也有同样的效果。

Unisphere 中的默认分配方法是横向资源调配。默认情况下，Unisphere 会跨后端端口均匀分配 RAID 组。Unisphere 按循环调度顺序将每个 RAID 组调配于一个单独的端口上，所有成员驱动器均位于同一端口。

纵向调配是多端口 RAID 组调配策略。RAID 组的驱动器置于两个或以上后端端口。这总会将此组的驱动器放置在多个 DAE 中，而且每个 DAE 都处于单独的后端端口上。

多总线调配有着一定的性能优势。但是，VNX 系列 SAS 后端的内置并行度对性能的提升略逊于传统 CLARiiON 存储系统。然而，对于大型 RAID 组（ ≥ 8 个驱动器）、RAID 1/0 组及具有高吞吐量工作负载的基于闪存的 RAID 组，跨若干后端端口分布会很有帮助。

多端口资源调配适合下列情况：

- ◆ 带宽要求使得需谨慎分布负载。
- ◆ 猝发主机 I/O 经诊断会造成后端端口瓶颈。
- ◆ 对 RAID 组 LUN 的写缓存转储经诊断会因端口饱和而影响主机读取。
- ◆ 特定 RAID 组中的大型数据块 I/O 请求经诊断会影响主机 I/O 响应时间。
- ◆ 经调配后的闪存驱动器服务于高吞吐量工作负载。
- ◆ 调配了大型（ ≥ 6 个驱动器）FAST Cache。

注意：与单端口资源调配相较而言，多端口资源调配在设置和维护方面需要更多的维护训练和规划。

第 11 章 可用性

本章包括以下主题：

可靠性	108
冗余	108
测量可靠性和可用性	110

VNX 中的可用性是通过高可靠性组件及组件与数据冗余的组合实现的。

可靠性

可靠性是指存储系统在操作压力下长时间运行且不出现硬件或软件故障的能力。VNX 系列体系结构包含许多专为确保高级别可靠性而设计的功能。高质量的组件、组装、设计验证及质量测试为此设计提供坚实后盾。经过成千上万个小时的现场验证，存储系统正常运行的系统可靠性评级高达五个九（99.999%）。精确的可靠性测量视资源调配中使用的型号和组件而定。

存储系统经过专门设计和构建，可通过完全数据路径保护，对整个存储系统的用户数据提供强有力的保护。此数据路径保护可防止未报告的数据不匹配错误。UltraPoint 硬盘驱动器阵列存储模块 (DAE) 在驱动器存储模块中包括了点到点设计。数据块操作环境将在驱动器级别监视低级别诊断，包括故障检测、隔离和纠错。RAID 保护与全局主动热备份、针对所有用户数据的基于扇区的校验和以及驱动器介质的自动后台验证相辅相成。

冗余

冗余是指系统在故障期间持续运行、不丢失主机对数据的访问并且维护数据完整性的能力。冗余是通过复制存储系统关键组件以及调配 RAID 类型和前端连接等选项来实现的。

在内置了冗余部件和组件的系统中，故障事件从统计意义上是相互独立的一个故障的发生毫不影响其他故障的发生可能性。仅当两个或更多相互依赖的组件同时发生故障的情况下，用户才会丢失数据访问。统计上不相关的两个故障同时发生的概率等于一个故障的发生概率乘以另一个故障的发生概率。换言之，由于一个故障的发生概率本来就很低，因此同时发生两个故障的可能性微乎其微。这样可大幅提高总体运行时间。

主动/被动体系结构

存储系统将基于主机中的每个 LUN 实施主动/被动体系结构。主动/被动表示从存储处理器到 LUN 有一个主动操作接口。注意：此为逻辑接口。两个存储处理器对于各自的 LUN 通常都是主动的。对等存储处理器可物理共享 RAID 组。在故障切换（主动变更）中，被动存储处理器将临时接管其对等 LUN 的所有权。此过程由存储系统中的内部重定向来完成，因此主机通常察觉不到。

但是，建议在主机或网络和存储系统前端端口之间建立多个连接，以便即使有一个路径或存储处理器发生故障，也可实现可用性。

多路径

如果有两条或多条到存储系统上 LUN 的路径，多路径允许从故障 I/O 路径故障切换到备选路径。I/O 路径开始于主机上的主机总线适配器 (HBA)，且包括到存储系统的这一路径上途经的所有组件和连接。I/O 路径上的硬件组件包括线缆、交换机、前端端口、存储处理器、后端总线和驱动器。此路径上也有软件层，包括硬件驱动程序、设备固件和主机上运行的应用程序。路径上的任何组件都可能影响对主机公开的存储的可用性。

原始 I/O 路径通常是性能最佳的路径。故障切换可用性功能可能对性能有着不利影响。例如，原始路径的性能可能得益于网络或存储系统的特殊调整或最佳物理路由。

基于主机的应用程序支持

路径故障切换需要基于主机的应用程序。

EMC PowerPath[®] 是一种基于主机的应用程序，而且它还具备多路径及多种其他性能和可用性功能。例如，PowerPath 对性能的增强功能是：借助多条到 LUN 的主动路径，I/O 负载可跨存储处理器的前端端口及主机的 HBA 实现平衡，以增强带宽和避免瓶颈。PowerPath 的可用性功能是自动回切。回切会在检测到故障 I/O 路径被修复时将存储系统恢复到原始 I/O 路径配置。

此外，O/S 也有本机的多路径应用程序，例如，Microsoft 多路径 I/O (MPIO[®]) 和提供类似功能的 HP-UX 物理卷链路 (PV Links[®])。最新版 MPIO 和 PV Links 都支持回切。

ALUA (非对称逻辑单元访问)

存储系统支持行业标准的非对称逻辑单元访问 (ALUA) 协议。ALUA 允许 I/O 流经一个或两个存储处理器传输到某个 LUN，以此来提供路径管理。ALUA 可以降低某些前端和后端故障对主机的影响。在内部，如果组件故障导致无法访问某个 LUN，ALUA 将阻止所连接的主机故障切换和主动变更 LUN。实现此目的的方式是通过对等存储处理器将 I/O 路由到 I/O LUN 所属的存储处理器。在外部，如果发生网络故障，I/O 可在仍有连接的情况下通过对等端口连接到其源。不论是否在前端端口上设为使用 ALUA，所有连接的主机都会获得此好处。

原始路径为最佳路径。故障切换的目标路径为非最佳路径。（请参见第 109 页上的图 36。）ALUA 不执行自动回切。始终应尽快恢复到使用最佳路径。非最佳路径的 I/O 路由对单个 I/O 使用两个存储处理器。这将延长主机响应时间，而且对存储系统性能可能有不利影响。

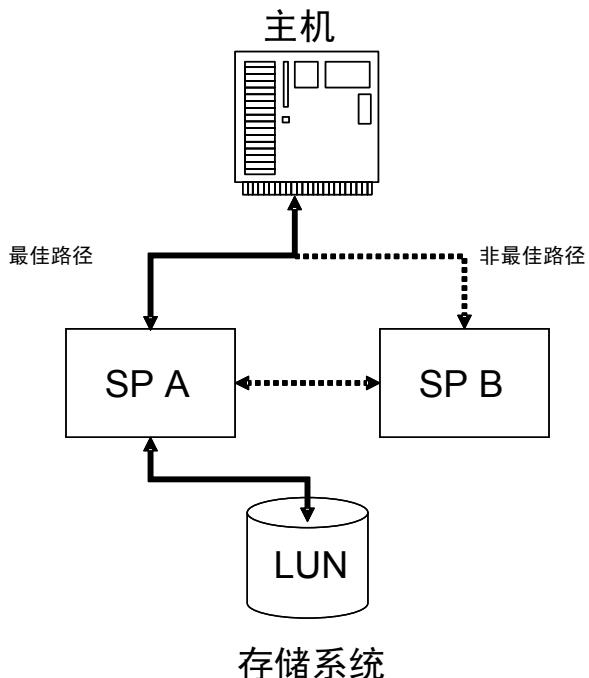


图 36 最佳和非最佳 I/O 路径

主机 O/S 必须支持 ALUA。PowerPath、MPIO 和 PV Links 都支持 ALUA。

测量可靠性和可用性

可靠性是指设备可按需执行意向功能的持续时间段。可靠性基于平均无故障时间 (MTBF) 和平均更换间隔时间计算得出。

预期可用性是指每年中系统可访问或传输任何所需数据块的时间与宕机分钟数的百分比。各组件的可靠性指标用于计算可用性指标“平均数据丢失间隔时间 (MTTDL)”。

可靠性指标

MTBF

MTBF 是可靠性的基本测量标准。MTBF 指标一般难以准确掌握。臆断该指标可导致作出对可用性产生不利影响的决策。

MTBF 是指组件或系统故障之前经过的时间段。制造商使用此测量标准，其值通常为“最佳情形”下获得。此时间段一般以小时为单位。

例如，磁盘制造商对现代 15k rpm SAS 硬盘驱动器提供的 MTBF 为 200 万个小时。

AFR

年故障率 (AFR) 是 MTBF 的替选指标。AFR 是指在指定用户群的条件下每年可能发生的故障数。AFR 是一个数字。它可用于确定备份策略。

例如，磁盘制造商对现代 15k rpm SAS 硬盘驱动器提供的 AFR 为 0.44%。

MTBPR

预期可靠性数据是指平均部件更换间隔时间 (MTBPR)（以小时为单位），它反映系统需更换部件的平均频率。最好不要等到设备发生故障再更换。磁盘故障通常可以预测到（例如，基于介质错误率），该产品应在发生故障之前更换。

可用性指标

MTTR

与 MTBF 相关的是 MTTR（平均修复时间）。这是对可用性极为重要的可维护性指标，该指标定义组件或子系统处于降级状态的时长，它以小时为单位。它是指在标识故障后将系统恢复到正常运转所需的时间量。MTTR 一般是假设已正确标识故障的条件下指定的修复时间。

在存储系统上下文中，最常见的 MTTR 是指完成重建所需的时间。当 RAID 组驱动器发生故障时会发生重建。在完成重建的过程中，RAID 组处于降级状态。MTTR 既指明重建如何影响可用性，也是对此的一个例子。

第 12 章 存储对象可用性

本章包括以下主题：

物理存储对象可用性	112
RAID 可用性差异	114
LUN 可用性	118
可靠性	120

RAID 专为提供数据冗余而开发。数据冗余是一种让数据在驱动器故障情况下可重新构建的技术。在 RAID 基础之上，EMC 等供应商开发了其他一些技术，可降低存储系统中介质错误及数据传输错误的影响。可以检测到丢失或损坏的数据，并且在可能时重新构建。注意：在存储系统上，错误检测和重新构建的粒度一般是磁盘-扇区级别。也就是说，扇区是可检测到其中错误并重新构建的最小数据量。

奇偶校验与镜像 RAID 级别在如何处理数据冗余以提供可用性方面存在差异。

物理存储对象可用性

驱动器经过专门设计和制造，可确保可靠性和高可用性。

硬盘驱动器故障模式

硬盘驱动器是高度可靠的存储，其平均无故障时间 (MTBF) 长达数十万乃至数百万个小时。但是，硬盘驱动器是复杂的机械设备。驱动器故障会随着驱动器的老化而增多。机械驱动器最终会因磨损而报废。撞击、震动及过热也都可能对它们造成损害。闪存驱动器更为可靠，因为它们没有移动部件。但是，闪存驱动器有其独特的故障特性，因此也存在磨损问题。

硬盘驱动器具有多种故障问题。众多的故障类别有时称为 **故障模式**。下面是常见的故障模式：

- ◆ 机械
- ◆ 介质
- ◆ 电子

机械

机械部件故障通常与磁盘轴电机或其轴承（包括读/写磁头传动装置）相关。例如，机械故障可能是磁盘轴电机在运转期间达不到完全操作速度。过热或震动都是机械故障原因，另外一个原因就是老化。

读/写磁头组件故障可以是机械或电子故障。声名狼藉的 **磁头崩溃** 属于机械故障，它是指读/写磁头划坏磁盘盘片的磁介质。

旋转震动 (RV) 可能导致机械硬盘驱动器中发生故障。RV 以弧度/秒²为单位。震动可能源于存储系统内部，例如，源于相邻的硬盘驱动器和系统风扇。它也可能源于存储系统外部，例如，源于散热通风和空调 (HVAC) 单元之类临近数据中心的重型设备。此外，猝发的随机工作负载可能使驱动器磁头快速移动，进而造成驱动器震动。

多驱动器环境中的某些驱动器包含板载优化（包括旋转震动向前补偿 (RVFF)），可尝试处理驱动器级别的机械错误。RVFF 将检测并补偿可能造成驱动器磁头定位错误的环境震动。

硬盘驱动器震动可能导致机械和介质故障。例如，震动可能延长硬盘驱动器磁头的定位时间，进而增加寻道时间，或者导致驱动器传动装置偏离磁道，进而引发要报告的介质错误。

机械故障是最常见的故障模式，也是可预测的错误，性能降低和错误增多都是这种故障的征兆。注意：闪存驱动器没有此类故障。

介质

介质故障可能是因传动装置的读或写错误造成的。介质故障也可能与机械硬盘驱动器上的读/写磁头故障相关。在闪存驱动器上，它可能是内存页或数据块的故障。

在驱动器的扇区或页中发生读取、写入或验证错误都是介质故障的征兆。发生读取错误时，将重新映射扇区。在重新映射期间，扇区或页的数据将传输到一个专用扇区。随着机械和闪存驱动器的老化，介质错误将会增多。大量重新映射扇区或页预示着驱动器故障。

电子

当重要电子组件发生故障时，驱动器电路板中会发生电子故障。驱动器控制器中的软件错误也是引发此故障的一种可能性。读/写磁头的故障可能是电子或机械故障。过热和震动是常见的电子故障原因。另外一个原因就是老化。此故障模式十分罕见，而且是不可预测的。

硬盘驱动器可靠性分类

企业级是驱动器可靠性的最高级别。这些驱动器经过专门设计与制造，可长时间提供 100% 使用周期的可靠操作。近线次之，但仍是极高的可靠性标准。最低级别是使用者可靠性或桌面可靠性。桌面硬盘驱动器是零售店出售的产品。桌面驱动器可能会出现在主机中，但绝不会出现在 EMC 存储系统中。

企业级和近线硬盘驱动器之间存在诸多差异。例如，企业级驱动器通常采用双端口（而不是单端口）连接，它们拥有速度更快的电机（15k rpm 与 7.2k rpm），其介质拥有验证磁属性的“特性”，而且其结构拥有更强的抗震动能力。企业级驱动器还可能拥有双控制器：一个用于磁盘盘片/磁头跟踪，另一个用于总线仲裁和数据出队/入队及缓冲。

最重要的差异是基于驱动器错误率的驱动器可靠性。此指标根据不可恢复的错误率 (UER) 进行报告。表 10（第 113 页）展示了典型的制造商报告的机械硬盘驱动器 UER。

表 10 制造商报告的机械硬盘驱动器 UER

典型的制造商报告的 UER 值		
驱动器可靠性评级	UER (位)	字节
用户	<1:10^14	<12.5 TB
近线	1:10^15	125 TB
企业	1:10^16	1.25 PB

如表所示，近线驱动器每读取 125 TB 即会出现一个错误。统计表明，600 GB 企业级驱动器约要经过 2,500 次全驱动器读取后才可能出现一个错误。

务必了解，UER 是统计上的测量标准。也就是说，特定近线级驱动器可能从不会有 UER。同样，企业级驱动器在经过 125 TB 读取之后出现 UER 的概率也很低。

自我监视、分析和报告技术 (SMART)

自我监视、分析和报告技术 (SMART) 是行业标准化的基于驱动器的故障检测和监视系统。

SMART 是 VNX 上所用驱动器的内置可用性功能。SMART 监视由驱动器的板载磁盘控制器独立执行。它使用驱动器的板载传感器和制造商的诊断算法来持续地监视和评估驱动器运行状况。

如果某些属性（例如，读取错误率或不可纠正的扇区数）超出制造商预设阈值，驱动器的 SMART 功能将向存储系统的系统软件通知即将发生的驱动器故障。遗憾的是，SMART 不能预测所有可能的故障。一般而言，SMART 对潜在的机械和介质故障的检测已经很到位。但是，机械和闪存驱动器的多数组件都是电子的。电子故障不像其他故障类型那样容易预测。

存储系统将使用此信息启动主动备份。请参见第 124 页上的“[全局热备份](#)”。

RAID 可用性差异

除 RAID 0 以外的所有 RAID 级别都具有针对单个驱动器故障的保护。磁盘发生故障时，数据保护将会降低。在 RAID 组中的一个驱动器发生故障时，大多数 RAID 类型都会考虑降级。某些 RAID 级别可针对多个驱动器故障提供一定程度的保护。例如，RAID 1/0 组可承受每个主要或镜像对的一个驱动器发生故障。RAID 6 组可承受两个驱动器故障，并且仍提供数据访问。数据保护通常会在完成重建操作之后恢复到热备盘。

镜像 RAID 组可以承受多个磁盘故障，前提是这些故障发生在独立的主要/镜像对中；不过，RAID 6 可承受任意的两个磁盘故障，因此它是可用性最高的 RAID 类型。

镜像 RAID 级别可用性

受支持的镜像 RAID 级别包括：

- ◆ RAID 1
- ◆ RAID 1/0

利用镜像，两个数据拷贝将存储于不同的驱动器上。随着数据写入驱动器，存储系统会自动将第二个拷贝写入到独立的驱动器。仅当数据安全写入这两个驱动器后，写入才算是完整的。可从任一驱动器读取数据。若使用完整数据拷贝进行保护，即表示任何镜像组的可用容量都只有总可用容量的一半。

对于镜像的数据保护，如果某个驱动器发生故障，或者检测到错误，将从仍在运行的对等驱动器上的拷贝重新构建丢失或损坏的数据。重建期间简单的驱动器到驱动器数据拷贝都会影响对拷贝中调用的驱动器的访问，但是不影响同一后端端口上的其他驱动器。

为了确保最高可用性，请将每个磁盘对的主要和镜像磁盘定位于不同的后端端口上。注意：驱动器到 RAID 组的添加顺序即为其绑定顺序，第一个加入对的为主要驱动器，第二个为镜像驱动器。

镜像取代不了数据备份或时间点数据复制。错误信息、用户损坏的数据和数据删除全都会拷贝到镜像对的两个驱动器。

RAID 1

RAID 1 是单一磁盘镜像。RAID 1 具有良好的数据可用性。RAID 1 是 FAST Cache RAID 级别。

RAID 1/0

RAID 1/0 是磁盘条带的镜像。RAID 1/0 具有绝佳的数据可用性。单个磁盘故障不会造成数据丢失。即使发生多个磁盘故障，也仍可能正常运行。但是，主要磁盘和其镜像不能同时发生故障，否则会导致数据丢失。请注意，这种级别不是双磁盘故障保护。

例如，在四磁盘 RAID 1/0 (2+2) 中，包含两个主要磁盘 0 和 1 及其镜像 0' 和 1'。主要磁盘 0 和 1 可能同时发生故障，这时会在镜像上保存数据。同样，镜像 0' 和 1' 可能同时发生故障，这时会在主要磁盘上保存数据。但是，如果 0 和 0' 或者 1 和 1' 同时发生故障，将造成数据丢失。

奇偶校验 RAID 级别可用性

受支持的奇偶校验 RAID 级别包括：

- ◆ RAID 3
- ◆ RAID 5
- ◆ RAID 6

奇偶校验是一种数据保护功能。奇偶校验 RAID 级别使用错误检测和重新构建算法来维护数据完整性。奇偶校验技术将创建称为奇偶校验数据的信息，并且产生在 RAID 组中存储奇偶校验数据的需求。奇偶校验数据是一种元数据形式。这将减少此组的可用数据容量。但是，维护奇偶校验数据所使用的容量少于镜像所需。与镜像相比，每个 RAID 组的奇偶校验数据最多需要两个驱动器容量 (RAID 6)，镜像则需要保存用户数据完整拷贝所需的容量。

所有奇偶校验 RAID 级别都会分条。在每个条带中，一个条带元素包含奇偶校验数据。如果某个驱动器发生故障，或者检测到介质错误，RAID 组中工作驱动器的奇偶校验元素及数据元素将用于重新构建故障驱动器的内容。

从已分配的奇偶校验信息重建磁盘对同一总线上其他驱动器的访问可能有不利影响。影响程度视组中的驱动器数及重建速率而定。重建是优先执行的操作。如果优先级是 ASAP，重建的运行速率较高，而大型 RAID 组可能占用后端总线的所有带宽。

RAID 3

RAID 3 具有良好的数据可用性。在 RAID 3 中，所有奇偶校验数据均位于一个专用的奇偶校验驱动器上。即使此组中的一个驱动器发生故障，也不会发生数据丢失。

RAID 5

RAID 5 具有良好的数据可用性。即使此组中的一个驱动器发生故障，也不会发生数据丢失。RAID 5 奇偶校验数据跨整个组的驱动器分配。奇偶校验使用的容量相当于此组的一个磁盘。

RAID 6

RAID 6 可提供存储系统上的最高数据可用性。RAID 6 是双向分配的奇偶校验 RAID 级别。将对每个条带计算两个奇偶校验元素。组中的一个或任意两个驱动器发生故障不会导致数据丢失。请注意，这是真正的双磁盘故障保护。RAID 6 奇偶校验数据跨整个组的驱动器分配。奇偶校验使用的容量相当于此组的两个磁盘。

不受保护的 RAID 级别 (RAID 0)

RAID 0 较为特殊。RAID 0 缺乏内置的数据保护。它仅提供分条 RAID 的性能和容量优势。我们建议不要将任何有商业价值的数据存储于 RAID 0 组上。

降级模式中的 RAID 组操作

RAID 组中某个驱动器故障将使 RAID 组进入降级模式。RAID 组处于降级模式时，不会丢失数据。但是，性能和可用性都会降低。

降级模式中的性能

故障驱动器对 RAID 性能的影响程度视 RAID 类型而定。

镜像 RAID 级别

单个故障驱动器不会影响镜像 RAID 级别组的性能。对 RAID 组的读取和写入仍会继续，不过是通过故障驱动器的仍在运行的对等驱动器处理负载。

奇偶校验 RAID 级别

一般而言，奇偶校验 RAID 级别组的性能会因故障驱动器而降低。

奇偶校验 RAID 组中的某个驱动器发生故障将导致 RAID 组的数据片段（奇偶校验元数据和用户数据）变得不可用。

例如，如果故障驱动器的 RAID 组包含 5 个驱动器 (4+1)，组内容的 20%（元数据和用户数据）将变得不可用。

缺失的驱动器不会影响写入性能。数据将像 RAID 级别 0 那样写入到不要计算奇偶校验的组。这实际会带来比正常操作下更好的组写入性能。

平均读取性能将会降低。性能降低视个别读取的位置是否在故障驱动器上而定。对故障驱动器上用户数据的读取将通过奇偶校验计算重新生成数据。此为资源密集型操作，需要读取此组中仍在运行的所有驱动器，而且存储处理器 CPU 资源需专用于重新生成奇偶校验计算。因此，读取的主机响应时间将会延长。如果故障驱动器包含某个卷中读取频繁的元数据（通常位于 RAID 组的第一个驱动器上），对性能的影响尤为严重。若是读取此组中仍在运行的驱动器上的用户数据，则不会影响整体存储系统性能。这些只是从各自驱动器中读取并传输。

RAID 组中的驱动器越少，对故障驱动器的读取请求的可能性越大，因此组性能越低。

注意：如果故障驱动器是 RAID 级别 3 组的专用奇偶校验驱动器，则没有性能影响。**另请注意：**RAID 级别 6 组与 RAID 级别 5 组中单个驱动器故障的影响大致相同。双驱动器故障的性能影响更为严重。双奇偶校验计算是需要更多 CPU 资源的操作。

降级模式中的可用性

在降级模式中，可用性将会降低。所有 RAID 级别（RAID 0 除外）均可在不丢失数据的情况下承受单个驱动器故障。某些 RAID 组可承受两个驱动器故障。但是，潜在的数据丢失可能由于 RAID 组丧失提供数据冗余的能力而增多。

例如，如果故障驱动器的 RAID 组包含 5 个驱动器 (4+1)，此组中的任何其他驱动器故障都是不可恢复的，而且会导致数据丢失。

尽快修复故障至关重要，因为这样将减少潜在的数据丢失。驱动器故障的概率极为复杂。但是，存储系统中一个驱动器的故障将会增加另一个驱动器发生故障的可能性。相反，自上次故障以来经历的时间越长，截下次故障的时间也越长。此关系对重建流程有重要影响。

RAID 组重建

在故障情况下重新构建 RAID 组驱动器的数据，即称为 **重建**。有时，这也称 **数据恢复**。从驱动器错误或故障中恢复是个耗时流程。RAID 组重建期间，数据很容易丢失。此外，还可能造成一些性能损失。

在重建期间，将恢复 RAID 组 LUN 中的数据（请参见第 67 页上的“[LUN](#)”）。重建将用运行正常的驱动器替换 LUN RAID 组的故障驱动器。借助奇偶校验 RAID 组中的驱动器，即将发生故障的驱动器中的数据将通过此组的奇偶校验重建到替换的驱动器上。借助镜像 RAID 组中的驱动器，数据将从故障驱动器的运行正常的对等驱动器拷贝到替换的驱动器上。

重建时间是重要的可用性注意事项。对重建时间构成影响的因素有许多。影响最大的因素包括：

- ◆ LUN 中使用的驱动器容量
- ◆ 驱动器类型（SAS、NL-SAS、闪存）和速度
- ◆ RAID 组的 RAID 类型和大小（仅限奇偶校验组）
- ◆ 工作负载
- ◆ 优先级

大部分容量都在使用中的大型驱动器比小型驱动器需要更长的重建时间；要重建的数据更多。

诸如闪存驱动器等的高速高性能驱动器比 NL-SAS 驱动器之类的较慢驱动器重建得更快。较高 rpm 驱动器比同类型的较低 rpm 驱动器重建得更快。

所选的 RAID 级别也可延长或缩短此过程。奇偶校验 RAID 类型必须从组中的所有驱动器读取奇偶校验信息，并且计算要重建的奇偶校验；RAID 组中的驱动器越多，要读取的数据也越多。RAID 5 需要对每个条带执行一个奇偶校验计算；RAID 6 需要对每个条带执行两个奇偶校验计算。额外的奇偶校验计算需要更多的时间和资源。

后台工作负载及其存储系统资源要求可缩短或延长此过程。如果要重建的 RAID 组在重建的同时执行读写操作，将会延长重建时间。

最后，我们要考虑的是优先级。重建可能的优先级分为四种：尽快 (ASAP)、高、中和低。ASAP 非常快地执行重建，但要使用大量存储系统资源。此优先级对工作负载性能可能有不利影响。“高”、“中”和“低”优先级所用的资源相对较少。“高”优先级的完成间隔时间自然要比“低”更短。默认重建优先级是“高”。

资源调配时，重建时间可成为主要的注意事项。在重建过程中，系统性能通常会降级，而且，数据在此期间容易遭受其他驱动器故障影响。谨慎做出资源调配选择，在提供最强数据保护的同时帮助尽快完成重建。

受重建影响最大的资源是后端总线和驱动器。对 CPU 利用率的影响相对较低。ASAP 重建会明显延长驱动器级别的主机响应时间，而且在奇偶校验重建中，可在总线上放置足够的数据，以降低此总线甚至是到不要重建的 RAID 组的总线上的主机带宽。

可通过用多个端口支持奇偶校验 RAID 组，最大限度降低重建对后端端口的影响。若是跨其他端口分布奇偶校验 RAID 组的驱动器，重建的 I/O 同样会跨这些端口分布。这样将降低重建对应用程序生成的工作负载的性能影响，因此可缩短重建时间。

LUN 可用性

FLARE 使用称为“验证”的流程检查存储系统 LUN 的完整性。“验证”流程将通过以下操作逐个位置地检查并尽量修复 LUN 位置的所有内容：

- ◆ 检测含有数据和奇偶校验不一致情况的磁盘位置
- ◆ 在 LUN 进入降级状态且数据无法重新构建之前，用冗余数据纠正有错的位置
- ◆ 重新映射已纠正数据的位置
- ◆ 报告不可纠正的磁盘位置

位置错误

LUN 中找到的错误要么可通过“验证”纠正，要么不可纠正。位置中的错误可从 RAID 组的冗余数据纠正。

任何通过“验证”流程找到的不可纠正错误都表示 LUN 上存在主机系统不可读取的位置。例如，当 LUN RAID 组因单个故障驱动器而进入降级模式时，如果另一个驱动器在此期间遇到介质错误，即为不可纠正的错误位置。（注意：对于 RAID 级别 6，这是可纠正的错误。）存储系统的“验证”流程无法确定 LUN 上不可纠正的错误位置是否正被主机占用。但是，不可纠正的错误通知应视为可能丢失数据的严重事件。

LUN 验证

LUN 上执行的“验证”分为两种类型。两种 LUN 验证流程是：

- ◆ SNiFF (SNiFFER)：按最低速率在所有存储系统 LUN 上连续循环地后台执行。这只是介质检查；它会要求驱动器读取数据块序列，但不执行对奇偶校验或数据的一致性检查。
- ◆ 后台验证 (BV)：仅当被故障触发或被用户请求时以较高速率执行。BV 执行数据一致性检查。

SNiFFER

SNiFFER 不会验证 LUN 的状态，而是提供一种对主机 I/O 性能没有任何显著影响的后台介质检查机制。其好处是：可在主机请求信息之前主动检测驱动器错误。

绑定 LUN 时，SNIFF 的启用优先级极低。SNIFF 不影响后端总线带宽利用率。它需要读取 LUN 中每个磁盘上的 512 KB 区域，大约每秒一次。只要某个 LUN 空闲，SNIFF 就会提高其验证速率，这样可缩短完成整个过程所需的总体时间。LUN 中驱动器的数量无关紧要。SNIFF I/O 在 LUN 中的每个驱动器上并行执行。由于执行速率极低，因此完全验证单个 LUN 可能需要好几天时间，具体视 LUN 的容量而定。

后台验证 (BV)

如果数据的有效性或完整性存在问题，BV 可用于立即检查 LUN 的一致性。

BV 是一种仅需数小时的操作。BV 的持续时间由诸多因素决定，例如，优先级、主机 I/O 加载、LUN 大小、RAID 组类型和驱动器类型。

BV 是优先执行的操作。优先级包括“低”、“中”、“高”或“ASAP”。可通过将优先级提升到“高”或“ASAP”，缩短 BV 的持续时间。注意：这样将占用更多的存储系统 I/O 资源，主要是磁盘利用率和后端总线带宽。因此，可能影响主机性能。默认 BV 设置为“中”。

BV 可以自动或手动进行计划。BV 将在故障 SP 主动变更到对等 SP 之后自动启动。如果对 LUN 的有效性存有疑问，并且需要比 SNiFFER 流程更新的 LUN 状态，也可通过 Unisphere 手动启动 BV。

虚拟池可用性

通常，对池适用的可用性注意事项与调配传统 LUN 时一致。常见注意事项包括：

- ◆ 底层存储设备的 MTBF
- ◆ RAID 级别数据保护
- ◆ RAID 组的数量
- ◆ 重建时间和其他 MTTR 功能

底层存储设备

应谨慎考虑设备级别可用性。闪存驱动器和 SAS 驱动器是所有 VNX 存储设备中可用性最高的。闪存驱动器没有移动部件，而且功耗比机械硬盘驱动器更低，因此可用性比机械硬盘驱动器更高。为了确保最高可用性，请用闪存驱动器和 SAS 驱动器调配虚拟池。在极大的同构池中使用 NL-SAS 时，最好搭配 RAID 6 级别保护

RAID 级别数据保护

绑定到一个虚拟资源调配池中的所有 LUN 都会因为池 RAID 组中的全面故障而丢失数据。该池中专用 RAID 组的数量越多，故障的影响就越大。

务必根据池中内容的价值为池选择合适的保护级别。

可供池使用的数据保护分为三个级别：

- ◆ RAID 5 具有良好的数据可用性。如果专用 RAID 组中的一个驱动器发生故障，不会发生数据丢失。RAID 5 适合中小型池。它还可以用于仅调配了 SAS 和高可用性闪存驱动器的小池或大池。
- ◆ RAID 6 可提供最高的数据可用性。对于 RAID 6 而言，专用 RAID 组中最多可以有两个驱动器发生故障而不会导致任何数据损失。注意：这是真正的双磁盘故障保护。RAID 6 适合任何大小的池，包括最大的池。

- ◆ RAID 1/0 可提供高数据可用性。专用 RAID 组中的单磁盘故障不会导致任何数据丢失。一个 RAID 组在其中多个磁盘出现故障后可能仍能工作。不过，主磁盘和其镜像不能同时发生故障，否则会导致数据丢失。请注意，这种级别不是双磁盘故障保护。RAID 1/0 适合中小型的池。

用户需要确定优先考虑事项：可用性、性能还是容量利用率。如果优先考虑的是可用性，则应调配 RAID 6。若是容量利用率或性能，而且您确定已经具备完善的数据保护策略和程序（备份、热备盘等），则考虑使用 RAID 级别 5 或 1/0 调配 FAST 池也是一个不错的选择。

RAID 组的数量

故障域涉及数据可用性。虚拟资源调配池由一个或多个专用 RAID 组组成。池的故障域就是一个池的专用 RAID 组。也就是说，一个池的可用性取决于任何一个专用 RAID 组的可用性。除非 RAID 6 是池的保护级别，否则 RAID 组超多的池的可用性会比 RAID 组数适中的池更低。

重建时间和其他 MTTR 功能

与传统 LUN 体系结构相比，基于池的体系结构如发生故障，所影响的 LUN 可能更多。将 RAID 组从降级模式快速恢复到正常操作状态，对存储系统的总体操作的重要性更胜于传统资源调配。

务必使用合适类型的热备盘。主动热备份的操作将减轻重建对后端性能的不利性能影响。此外，始终应尽快更换故障驱动器，以维持可用热备盘的数量并且缩短 RAID 组处于降级模式的时间间隔。

可靠性

可靠性是指存储系统在操作压力下长时间运行且不出现硬件或软件故障的能力。存储系统体系结构包含许多专为确保极高级别可靠性而设计的功能。高质量的组件、组装、设计验证及质量测试为此设计提供坚实后盾。经过成千上万个小时的现场验证，存储系统正常运行的系统可靠性评级高达五个九 (99.999%)。精确的可靠性测量视所用的型号和组件而定。

存储系统经过专门设计和构建，可通过完全数据路径保护，对整个存储系统的数据提供强有力的保护。此数据路径保护可防止未报告的数据不匹配错误。UltraPoint 硬盘驱动器阵列存储模块 (DAE 和 DPE) 包括驱动器存储模块中的点到点设计。数据块操作环境将在驱动器级别监视低级别诊断，包括故障检测、隔离和纠错。RAID 保护与全局主动热备份、针对所有用户数据的基于扇区的校验和以及磁盘介质的自动后台验证相辅相成。

第 13 章 VNX 可用性

本章包括以下主题：

前端	122
存储处理器	122
后端	123

前端

所有 VNX 系列型号的每个存储处理器都标配了多个 iSCSI 和光纤通道前端端口。支持同时连接到 iSCSI 和光纤通道主机。

高可用性需要多个连接。在高度可用的配置中，来自主机的 I/O 应连接到存储处理器上的多个端口。此外，主机和对等存储处理器之间的 I/O 连接也应设置到位。

允许主机连接到多个存储系统前端端口的配置是一种称为 **多路径** 的技术。这些附加端口可用于提供对一个 LUN 的主要访问及对另一个 LUN 的备选访问。存有备选路径，让存储系统可在路径或端口发生故障时进行前端端口到前端端口的重定向。**故障切换**是指在发生故障时使用备选路径。**回切**是指在纠正路径故障后返回到原始路径。

故障切换通常起因于网络错误。但是，存储系统组件发生故障时也可能发生故障切换。在存储系统上，故障切换由基于主机的 PowerPath 或其他基于主机的多路径应用程序启动（相关详情，请参见第 108 页上的“[多路径](#)”）。在故障切换中，正常运行的端口将开始服务于主机 I/O 请求。

与冗余网络配置结合时，资源调配最好创建到每个存储处理器上至少两个前端端口的连接。此外，应至少有一个或多个连接指向存储系统上对等 SP 的前端端口。此配置除了供 SP 故障切换使用外，还允许在自有 SP 上进行端口到端口故障切换。

存储处理器

存储系统拥有两个高度可靠的存储处理器。只要配置恰当，存储处理器就能提供冗余的存储访问操作环境。

主动/被动所有权模式

为了平衡工作负载，存储系统的 LUN 应跨存储处理器分配。因此，存储处理器将以并行而高效的方式服务于工作负载。

LUN 一次仅分配给一个存储处理器。这视为**主动所有权**。任一存储处理器都可访问存储系统后端上的 LUN。但是，I/O 不定向到非自有的存储处理器。对非自有 LUN 的现有非活动路径归对等存储处理器**被动拥有**。注意：这要求存储处理器对彼此的写缓存建立镜像（请参见第 44 页上的“[内存](#)”）。

如果对一个存储处理器的所有 I/O 发生网络故障，或者可能是存储处理器故障，LUN 将**主动变更**到对等存储处理器。主动变更会将 LUN 的所有权类型从被动改为主动。主动变更由 PowerPath 或其他基于主机的多路径应用程序透明地启动（请参见第 108 页上的“[多路径](#)”）。仍在运行的存储处理器将接管所有权，并且在完成与新 LUN 的缓存页关联后开始服务于主机 I/O 请求。

注意：为了确保高可用性，单一存储处理器的 CPU 资源要足以处理整个存储系统的工作负载。

后端

后端包括多个配有冗余端口和存储以确保数据安全性的磁盘存储模块。

系统驱动器和写缓存可用性

DPE 或 DAE O/S 中的前四个硬盘驱动器 (0 - 3) 具有一些为存储系统操作保留的容量。这些驱动器称为系统驱动器。它们包含存储区。存储区是指针对以下项保留的容量：写缓存（以便于在某些故障期间快速写入此区域）、存储系统操作环境的系统文件、永久存储管理器 (PSM) 以及操作环境的配置数据库。

存储系统将启动所有系统驱动器。实际上，系统驱动器中的专用数据远比存储区要多；电源或某些组件发生故障时，写缓存将转储到存储区。这些驱动器上具有多个专用 LUN，包括用于以下项的 LUN：

- ◆ 用于 RAID 1 中两个 SP 的操作系统
- ◆ 永久存储管理器 (PSM)
- ◆ RAID 1 三层镜像中的操作环境数据库
- ◆ RAID 5 中的存储区

这种驱动器组织提供的保护可承受两个驱动器故障。

系统驱动器可以像系统上的任何其他驱动器那样使用。但是，系统驱动器存在保留容量，因此可用容量比数据驱动器更少。此外，存储系统对系统驱动器的使用情况可能影响到此处存储的应用程序数据的响应时间。有关系统驱动器资源调配性能规划建议，请参见《最佳做法》白皮书。注意：系统驱动器不能用作虚拟池的组成部分。池资源调配要求整个驱动器的容量都可供使用。

任意这些专用 LUN 发生单个驱动器故障时，O/S、PSM、操作系统数据库或存储区都不会重建到热备盘。而只有系统驱动器用户空间中创建的用户 LUN 会重建到热备盘；专用 LUN 则是在更换故障驱动器后重建。注意：在特定故障情况下，将禁用读/写缓存，禁用状态一直持续到更换故障系统驱动器为止。禁用的读/写缓存对整体存储系统性能有不利影响。

例如，如果总线 0 存储模块 0 上的驱动器 0 发生故障，而且没有用户 LUN，交换中就不会用到热备盘。如果跨此驱动器绑定了用户 LUN，只有此用户 LUN 数据会重建到热备盘。更换故障驱动器之后，所有数据将拷贝/重建到驱动器 0（镜像中的 OE DB、奇偶校验中的存储区、镜像中的 PSM 及热备盘中的用户 LUN）。

此外，请参见第 35 页上的“备用电源”，以了解对 DAE O/S 应用的电源故障保护。

后端端口

如第 123 页上的“[后端](#)”中所述，存储系统的后端端口会将两个 SP 冗余连接到 DPE 或 DAE 中承载的存储系统驱动器。SAS 后端提供两条到驱动器数据的路径。

每个 DAE 都有两个 LCC。每个 SP 都会通过其 SAS 端口及 LCC 连接到一个 DAE。如果 DAE 的 LCC 发生故障，SP 将失去与此 DAE 中所有驱动器的通信。此外，与 DAE 链中更下方的 DAE 的通信也会中断。

从 LCC 故障中恢复可能导致使用后台验证 (BV) 或重建和主机主动变更。注意：从 LCC 故障中恢复对主机 I/O 性能可能有不利影响。有关 LCC 可用性的详细信息，请参见《最佳做法》白皮书。

全局热备份

操作环境会持续监视其驱动器的状态。热备盘是指用于更换即将发生故障的驱动器的驱动器。主动备份是指在驱动器借其 SMART 报告功能指示有待处理的故障时自动激活热备盘。

热备盘将更换任何具有绑定 LUN 的冗余 RAID 组中的故障驱动器。热备盘不会更换未绑定的驱动器、RAID 级别 0 RAID 组中的驱动器或个别磁盘。

每个存储系统可配置的最大热备盘数视型号而定。入门级存储系统可分配的热备盘可能比较高型号更少。

驱动器发生故障时，将执行热备份。备盘是通过算法从之前指定的热备盘池中选择的。可用作热备盘的驱动器类型存在一些限制。

在主动备份中，即将发生故障的可疑驱动器将在它发生故障之前拷贝到备盘。在此流程前进的同时，可疑驱动器将继续服务于 I/O。拷贝完毕后，可疑驱动器即会停用（断电），热备盘将取代它在 RAID 组中的位置。如果驱动器在此流程完成之前发生故障，剩余的缺失数据将通过本节中所述的流程重建到热备盘。

主动将数据拷贝到备盘不需要太多的资源，而且通常要比从奇偶校验和仍在运行的驱动器重建数据更快。与奇偶校验重建相比，磁盘到磁盘拷贝具有更低的后端负载和存储处理器负载。此优势将扩展到镜像类型 RAID 级别，该级别会在 RAID 组不处于降级模式时执行拷贝。这是可用性优势。

例如，如果拷贝期间遇到不可纠正的磁盘错误 (UER)，可重新构建缺失的数据。在重建类型的数据恢复中，由于故障驱动器不再可用，因此 UER 将导致数据丢失。（RAID 6 重建则属例外，它可承受两个驱动器故障。）

手动更换故障驱动器后，称为 **均衡** 的流程会将数据从热备盘恢复到 RAID 组的更换驱动器。均衡是磁盘到磁盘拷贝操作，它与 RAID 1 重建或主动拷贝有着相同的后端端口和驱动器影响。均衡结束后，热备盘将返回到全局池，供重复用于更换。

注意：RAID 组的热备盘操作不是正常操作。正常操作要到更换故障驱动器、均衡其更换项并且开始处理 I/O 之后恢复。如果在驱动器发生故障时没有配置适当类型和大小的热备盘，则不执行重建。RAID 组的降级状态会持续到更换故障驱动器为止，故障驱动器的 RAID 组随后从奇偶校验或其镜像驱动器重建，具体视 RAID 级别而定。

主动热备份可降低在重建期间发生另一个磁盘故障的风险。主动备份的 RAID 组始终有完整的存储设备数。注意：此组仍视为降级状态。此外，还需执行均衡（但无需重建）来恢复 RAID 组的状态。均衡不像重建那样要使用资源。

全局热备份不是操作环境中强制执行的正常操作的要求。但是，若要提高整体可用性，它不失为慎重之选。

重建日志

如果驱动器似乎要超时，重建日志将允许它有额外的时间从当前执行的任意操作中恢复，而不会让主机 I/O 执行长时间延迟。这样，还可避免对驱动器执行完整重建操作。重建日志也称为 **驱动器试用**。

仅在特定情况下才调用重建日志。如果驱动器实际发生故障 — 例如，它不再接受命令，报告致命条件（例如，“硬件错误”）或从总线被绕过 — 将立即停用此驱动器，并使用热备盘（如果有）。如果已发给驱动器的命令未能在合理的时间内完成（即，这些命令似乎要超时，但是驱动器实际仍存在，而且在任何提及的条件中也未发生故障），将调用重建日志。

软件和固件更新

操作环境可通过更新，整合新功能并且纠正已发现的问题。

磁盘阵列操作系统可使用无中断更新 (NDU) 流程在线升级。DAE LCC 固件 (FRUMON) 使用（操作环境升级中所含的）NDU 流程在线升级。磁盘固件一般不建议升级，也不要求升级（除非实施现场更改命令或 ETA），而且它的升级需要计划宕机。

第 14 章 结论

阅读此文档之后，您现在应该熟悉了 VNX 的基本概念和功能。这有助于理解《EMC Unified Best Practices for Performance and Availability — Applied Best Practices》（EMC Unified 性能和可用性最佳做法 — 适用的最佳做法）中所述的性能和可用性调整的详细信息。（此白皮书也称为《VNX 最佳做法》。）

本文提供了大量信息。但是，有几个要点要求您充分理解，以帮助您从 VNX 获取最大价值，并且了解《VNX 最佳做法》中的调整建议。它们是：

- ◆ 存储基础架构是指成本、容量、性能和可用性的平衡。
- ◆ 满足工作负载容量要求的方式有许多。
- ◆ 性能是可评估的行为。为此，您需要：
 - 了解如何描述您的工作负载。
 - 具有预计、过去和当前的性能指标。
- ◆ VNX 是一种有资源需要管理的、基于计算机的设备。
- ◆ 可用性关乎冗余和数据完整性。

而且，有多种物理和逻辑存储技术可保护您的数据。成本、容量和易用性通常是驱动器性能和可用性决策的依据。

为了获得最高性能，您需要了解存储系统的工作负载。这包括与主机应用程序相关的知识。您应了解应用程序如何生成 I/O。工作负载可能很简单，来源于单一应用程序；它也可能很复杂，即由一个或多个具有不同 I/O 特征的主机上的多个应用程序生成。您需要了解 I/O 的吞吐量、带宽和所需响应时间，以确定所需性能。如果您可确定工作负载特征，则调整将是一个简单的流程。除了确定工作负载特征之外，所收集的信息对创建历史基准（供以后调整和调查使用）也很重要。注意：如果工作负载的需求超出底层存储系统的性能功能，则应用性能调整的作用不大。

VNX 实际是两台连接了大量驱动器的计算机。它需要管理和调配 CPU、内存（缓存）和 I/O（SAS 后端端口和存储）资源，以实现最佳性能和最高可用性。务必注意后端和前端端口的可用带宽、可用 CPU 周期以及驱动器的吞吐量、带宽和存储容量。这些资源由工作负载占用一部分。另一部分则用于整体系统维护，例如，LUN 扩展之类的备份和存储系统管理任务。此外，还需要维护保留资源，以确保在操作处于降级模式及资源用量意外增多的情况下仍有可接受的性能。

VNX 凭借内置的可靠性和冗余硬件成为了高度可用的系统。冗余可扩展到数据保护。需谨慎调配资源，以确保最高的数据保护和性能。请记住：为确保高可用性而配置存储系统可能要牺牲一定程度的性能。（性能和可用性都视相同资源而定。）评估满足工作负载即时需求所需的资源分配，同时维持用于维护以及确保高可用性的保留资源，是一个连续流程。

本文档及此处提及的其他文档均可在 [Powerlink](#)[®]（EMC 面向客户和合作伙伴的受密码保护的外联网）上找到。我们建议您阅读有关工作负载应用程序及其主机硬件和软件的最佳做法信息。网络元素供应商的最佳做法也应加以关注。其他网络信息资源是 Powerlink 上找到的《EMC Networked Storage Topology Guide》（EMC 网络存储拓扑指南）。

最后，请注意存储系统是众多大型系统中重要的相关部件。与主机和网络相关的基础知识不在本文档的范围内。但是，存储系统的性能和可用性在很大程度上受到将其连接到主机、主机应用程序以及主机硬件和操作系统软件的网络所具备的性能和可用性影响。在大多数情况下，提高网络或主机的性能或可用性较之调整存储系统会带来更高的性能和可用性。

“当我使用一个字词的时候，它的意思恰恰就是我要它意指的意思，既不多也不少。”

— Humpty Dumpty, 源自 1871 年 Lewis Carroll 出版的《Through the Looking Glass (And What Alice Found There)》

10 GbE — 每秒 10 GB 以太网协议。

ABQL — 平均忙队列长度（每个驱动器）。

AFR — 年故障率。

ALUA — 非对称逻辑单元访问协议。

ANSI — 美国国家标准协会。

AVM — 自动卷管理。

BBU — 电池备份单元。

BURA — 备份、恢复和归档数据存储安全域。

BV — 后台验证。

CAS — 内容可寻址存储；由 EMC Centera® 实施的基于对象的存储。

CBFS — 通用数据块文件系统。

Celerra — 传统文件存储系统的名称。

Celerra Network Server — Celerra 产品的官方名称，指 File 和 NS 产品系列。

CHAP — 挑战握手认证协议。

CIFS — 通用 Internet 文件系统。

CLI — 命令行界面。

CMI — 配置管理接口。

CPU — 中央处理单元。

DAE — 驱动器阵列存储模块。

DART — 实时数据访问。

DAS — 直连存储。

Data Mover — 运行 VNX Operating Environment for File 的 VNX 存储模块组件 (X-Blade)。

DBMS — 数据库管理系统。

DLU — 直接 LUN，也称为虚拟资源调配池密集 LUN。

DPE — 磁盘处理器存储模块。

DR — 灾难恢复。

EFD — 企业闪存驱动器。

ESM — EMC Support Matrix。

ESX — VMware 企业级服务器虚拟化产品。

FAQ — 常见问题解答。

FAST — 全自动存储分层。

FAST Cache — 由闪存驱动器构成的辅助 I/O 缓存。

FC — 光纤通道。

FCoE — 以太网光纤通道流量。

FCP — SCSI 光纤通道协议。

FLARE — 光纤逻辑阵列运行时环境。传统 CLARiiON 的操作系统名称。

FLU — FLARE LUN，现称为传统 LUN。

FMP — 文件映射协议。

GB — 吉字节。

GB/s — 每秒吉字节。

Gb/s — 每秒千兆位。

GbE — 千兆位以太网（1 Gb/s 以太网）。

GHz — 千兆赫。

GigE — 1 Gb/s 以太网。

GMT — 格林尼治标准时间。

GUI — 图形用户界面。

HA — 高可用性或高度可用。

HBA — 主机总线适配器：在主机系统总线和存储系统之间充当桥梁的设备。

HP-UX — UNIX OS 的 Hewlett-Packard 公司专用版本。

HVAC — 加热、通风和空调。

Hz — 赫兹。

ICA — 映像拷贝应用程序。

IEEE — 国际电气与电子工程师学会。

IETF — Internet 工程任务组

iFCP — 允许 FC 设备将 IP 网络用作光纤交换基础架构的协议。

IHAC — 我有一个客户。

Internet 工程任务组 — 建立 TCP/IP 协议套件的国际组织。

Internet 协议 — 与 TCP 共同用于通过以太网网络传输数据的协议。

IOPS — 每秒输入/输出操作。

IP — Internet 协议。

IPSec — Internet 协议安全性。

IPv4 — Internet 协议版本 4。

IPv6 — Internet 协议版本 6。

iSCSI — Internet SCSI 协议。将 SCSI 命令发送到存储系统上驱动器的标准。

ISL — 交换机间链路。连接网络中的两个或多个交换机。

ISO — 国际标准组织。

IT — 信息技术。另外也指管理计算机系统的部门。

JBOD — 简单磁盘捆绑。

Kb — 千位。

KB — 千字节。

Kb/s — 每秒千位。

KB/s — 每秒千字节。

LACP — 链路聚合控制协议。

LAN — 局域网。

LBA — 逻辑块地址。

LCC — 链路控制卡。

Linux — 与硬件无关的开放系统的操作系统环境。

LUN — 逻辑单元编号。

LVM — 逻辑卷管理器。

MAN — 城域网。

Mb — 兆位。

MB — 兆字节。

Mb/s — 每秒兆位。

MB/s — 每秒兆字节。

MCM — MPFS 配置管理器

MetaLUN — 通过分条或合并多个 LUN 对象所构建的 LUN 对象。

MHz — 兆赫。

MIB — 管理信息库。

MirrorView — CLARiiON 和 VNX 灾难恢复应用程序

MPFS — 多协议文件系统

MPFS 配置管理器 — 用于自动配置客户端上 MPFS 协议的文件工具。

MPIO — Microsoft 多路径 I/O

MR3 写入 — 当整个 RAID 条带收集到缓存中并且一次性写入时 VNX RAID 引擎执行的操作。

MTBF — 平均无故障时间。

MTTDL — 平均数据丢失间隔时间。

MTTR — 平均修复时间。

MTU — 最大传输单位

NAS — 网络连接存储。

Navisphere — 适用于 30.0 以上的 FLARE 版本的 CLARiiON 资源管理系统软件。

Navisphere Analyzer — CLARiiON 性能分析系统软件。

NCQ — 本机命令排队

NDU — 无中断更新

NFS — 网络文件系统

NIC — 网络接口卡。

OLTP — 在线事务处理系统。

OOB — 带外。

OS — 操作系统。

PATA — 并行 ATA 磁盘 I/O 协议。

PB — 拍字节。

PC — 个人计算机

PCI Express — 基于计算机的系统所使用的一种总线协议。

PCI — 外围组件接口。

PCIe — PCI Express。

PCI-X — 扩展 PCI。

PDU — 配电装置。

Powerlink — EMC 面向客户和合作伙伴的受密码保护的外联网。

PowerPath — EMC 基于主机的多路径软件。

PSM — 永久存储管理器。

QA — 质量保证。

QFULL — 队列已满。

QoR — 结果质量。

QoS — 服务质量。

RAID — 独立磁盘的冗余阵列。

RAID 级别 — 提供容错及容量和性能提升的驱动器组织。

RAID 组 — 具有相同 RAID 级别的 2 - 16 个驱动器之间的逻辑关联。

RDBMS — 关系数据库管理系统。

RFC — 注释请求。

RFI — 信息请求。

RFP — 建议请求。

RFQ — 报价请求。

RLP — 保留 LUN 池。

ROM — 粗数量级。

ROT — 经验法则。

Rpm — 每分钟转数。

RPQ — 产品限定符请求。

RTO — 恢复时间目标。

RV — 旋转震动。

SAN — 存储区域网络。

SAN Copy — 从 CLARiiON 存储系统到存储系统的拷贝应用程序。

SAP — 软件公司 SAP AG 推出的企业资源规划应用程序。

SAS — 串行连接 SCSI。

SATA — 串行 ATA 磁盘 I/O 协议。

SCSI — 小型计算机系统接口。

SLA — 服务级别协议。服务提供商和客户之间提供可测量服务或资源访问级别的合同。

SLIC — 小型 I/O 卡。

SnapView — CLARiiON 和 VNX 时间点拷贝应用程序。

SNMP — 简单网络管理协议

SP — 存储处理器。

SPE — 存储处理器存储模块。

SPS — 备用电源系统。

SSD — 固态磁盘。

TB — 太字节。

TCP — 传输控制协议：一种与 IP 共同用于通过以太网网络传输和接收数据的协议。

TCP/IP — 用于 Internet 及其他类似网络的通信协议对

TCP/IP 减负引擎 — 一种连接到以太网网络的基于协处理器的主机组件。

TLU — 精简资源配置 LUN。

TOE — TCP/IP 减负引擎。

UER — 不可恢复的错误率

Unisphere — 适用于 FLARE 版本 30.0 和更高版本的 VNX 资源管理系统软件。

UNIX — 开放系统的操作系统环境。

UPS — 不间断电源。

VLAN — 虚拟局域网。

VLAN 标记 — 用于分隔 VLAN 的机制。

VM — 虚拟机。

VMware — EMC 虚拟机应用程序系列。

WAN — 广域网

WCA — 写缓存可用性。

Windows — Microsoft 专有操作系统环境。

Wintel — 用于描述基于 Intel 硬件体系结构和 Microsoft Windows 操作系统的计算机的行业术语。

WORM — 一写多读。

WWN — 全球通用名称。

百分比利用率 — 对所用资源量的测量标准。

绑定 — 将驱动器整合到 RAID 组。

饱和度 — 存储系统资源的加载量达到一定的程度，若继续添加 I/O，将显著延长系统响应时间，但不带来额外吞吐量的情况。

保留 LUN — 请参见“专用 LUN”。

保留 LUN 池 — 支持所安装应用程序的 LUN 组，例如，MirrorView 和 SnapView。

报价请求 — 一种采购文档类型，用于请求潜在卖家提供有关普通或标准产品或服务的报价。有时也用作建议请求。

备份 — 将数据拷贝到另一个性能通常较低的驱动器，作为原始驱动器故障的预防措施。

本机命令排队 — 基于驱动器的 I/O 执行优化技术。

边缘交换机 — 核心边缘配置的 SAN 周围的光纤通道交换机。

标准 — 一份经认证机构批准的公认文档，可为常见和重复用途提供活动或其结果的规则、指导准则或特性，旨在实现指定的上下文中的最佳结果。

标准 — 作为判断或决策依据或作为产品、服务、结果或流程的评估依据的标准、规则或测试。

并发 I/O — 在一个共享资源上同时有多个活动的 I/O 请求。

并行 ATA — 传统 CLARiiON 上使用的磁盘 I/O 协议。

不可恢复的错误率 — 硬盘驱动器的位错误可靠性指标。

部门系统 — 支持业务组织内单一部门的存储系统。

操作环境 — 操作系统。

操作系统 — 计算机上用于控制应用程序和资源管理的软件。

侧写 — 绕过写缓存，其中 RAID 引擎将写入立即调度到磁盘。

侧写大小 — 为特定 LUN 写入缓存的最大请求大小（以数据块为单位）。

差异 — 与已知基准或预期值的量化偏差、偏离或分歧。

超额预订容量 — 超出调配池容量的精简 LUN 配置容量。

程序 — 按照明确定义的顺序执行的一系列步骤，用于完成某项任务。

池 — 由 CLARiiON 虚拟资源配置功能管理的驱动器组。

池 LUN — 虚拟资源配置池上调配的 LUN。

传统系统 — 没有最新硬件和软件的较旧存储系统。

串行 ATA — CLARiiON 上使用的磁盘 I/O 连接。

串行连接 SCSI - 用于将数据移动到驱动器的点到点串行协议。

磁道 — 硬盘驱动器磁盘盘片上组织和存储数据的环形区域。

磁盘处理器存储模块 — 包含 CLARiiON 存储处理器和驱动器的机柜。

磁盘交叉 — I/O 因其地址和大小的缘故而需要访问磁盘中的不止一个条带元素，从而导致两个后端 I/O（而非一个）。

磁盘卷 — 从存储系统导出的 VNX File 物理存储单元。

磁盘控制器 — 用于控制硬盘驱动器的基于微处理器的电子单元。

磁盘盘片 — 硬盘驱动器的组件；它是用于存储磁数据的圆形磁盘。

磁盘阵列存储模块 — 最多包含 15 个 CLARiiON 驱动器的架装存储模块。

磁盘轴 — 硬盘驱动器的组件；它是安装磁盘盘片的轴。此外，磁盘轴有时也指硬盘驱动器。

磁头崩溃 — 读/写磁头与磁盘盘片物理接触的灾难性硬盘驱动器故障。

粗数量级 — 精确到数量级的估计。

猝发 — I/O 量随时间推移而高度变化，或者随合理定义的峰值定期变化。

存储池 — 支持读写数据访问的驱动器的逻辑结构。

存储处理器 — CLARiiON 体系结构的逻辑分区，包括 CPU 和内存。

存储处理器存储模块 — 包含 CLARiiON 存储处理器的物理架装机柜。此存储模块不包含驱动器。

存储对象 — 支持读写数据访问的逻辑结构或物理设备。

存储密度 — 可存储于指定存储卷中的信息量（以 GB 为单位）的测量标准。

存储模板 — 用于配置 VNX File 存储驱动器的预定义参数集。

存储区 — DAE0 驱动器上用于存储 CLARiiON 系统文件和缓存转储的特殊区域。

存储区域网络 — 专为共享驱动器而设计与构建的网络。

存储系统 — 一个包含多个硬盘驱动器、缓存及相关智能的系统，可实现信息和应用程序的安全、经济存储。

存储阵列 — 存储阵列。

大数据块 — 容量大于 64 KB 的 I/O 操作。

带宽 — 存储系统性能的一种测量标准，以每秒兆字节 (MB/s) 为单位。

低层控制器 — SPA 到 SPB 的直接通信总线连接。

订阅的容量 — 为池中的精简 LUN 配置的总容量。

读/写磁头 — 将信息记录到磁盘盘片或从中读取信息的硬盘驱动器组件。

读缓存 — 专用于优化读取 I/O 的缓存内存。

端口 — 存储系统与其他计算机和设备之间的接口设备。另外也指驱动器和总线之间的接口。

短寻道 — 仅使用 RAID 组的一部分的 LUN 性能优化技术。

- 堆栈** — 分层协议。
- 队列已满** — 发送给主机的指明端口或 LUN 队列无法再接受条目的 iSCSI 协议信号。
- 多路径** — 对 LUN 间多个主机 I/O 路径的调配。
- 多线程** — 并发 I/O 线程。
- 多协议文件系统** — EMC 对 NFS 或 CIFS 使用的备选 NAS 协议。
- 方法** — 由精于一门学科的人员使用的做法、技术、程序和规则的系统。
- 非对称逻辑单元访问** — 一种行业标准的多路径协议。
- 非最佳路径** — 从主机到 LUN 的 ALUA 故障切换 I/O 路径
- 分层应用程序** — CLARiiON 和 VNX 安装的应用程序软件。
- 分层应用程序** — 分层的应用程序。
- 分配容量** — 当前分配给基于池的 LUN 的物理总容量
- 分条** — 跨 RAID 组中的大量驱动器分配顺序存储块。
- 风险** — 不确定的事件或情况，当发生此类事件或情况时，可能对目标造成正面或负面影响。
- 服务时间** — 驱动器或资源执行单一 I/O 所需的间隔时间。
- 服务质量协议** — 系统或网络中定义的承诺性能级别。
- 负载平衡** — 跨可用资源均匀分配数据或处理。
- 附件** — 驱动器硬件连接器或接口协议。在 CLARiiON 上，它可以是光纤通道、SAS 或 SATA。
- 富介质** — 允许接收方积极参与的工作负载。有时称为交互介质。
- 高度可用** — 在发生一个故障后仍可提供数据访问的系统的特性。
- 高峰** — 存储系统上显著的突发性负载急增。
- 工作负载** — 发给存储系统以执行一组应用程序任务的 I/O 请求模式的特性，包括 I/O 量、地址模式、读写比率、并发性和猝发性。
- 估计** — 对可能的数量或结果的量化评估。通常应用于成本、资源使用和持续时间。
- 固态磁盘** — 对数据存储使用非易失性半导体内存的驱动器。

故障 — 软件程序的操作中的错误。

故障 — 系统中硬件组件的故障。

故障模式 — 故障的原因，或者启动故障引发流程的事件的原因。

故障切换 — 因原始路径故障而使用备选路径。

关系数据库管理系统 — 通常承载于存储系统上的数据库，例如，Oracle、DB2、Sybase 和 SQL Server。

光纤通道 — 串行数据传输协议：ANSI X3T11 光纤通道标准。

广域网 — 跨大面积（可能是全球）的地理区域扩展的计算机网络。

规范 — 以一种全面、准确且可验证的方式详细介绍系统、组件、产品、结果或服务的要求、设计、行为和其他特性的文档。通常，在规范中会涵盖用于确定是否已满足这些规定的程序。

国际标准组织 — 维护标准的国际组织。

国际电气与电子工程师学会 — 一个国际标准组织。

合并 — 在将较小的缓存 I/O 发送到驱动器之前将其合并为较大的 I/O。

核心 — 共同驻留在 CPU 芯片上的处理器单元。

核心交换机 — SAN 体系结构中间配有数百个端口的大型交换机或控制器。

赫兹 — 每秒一次。

后端 — VNX 体系结构从 SP 到后端总线和驱动器的逻辑分区。

后端 I/O — 存储处理器与驱动器之间通过后端总线的 I/O。

后端总线 — 将存储处理器连接到驱动器的 VNX SAS 后端端口。

后台验证 — 由 CLARiiON 自动读取 LUN 奇偶校验扇区并验证其内容，以预防故障。

环境 — 计算机的硬件平台、系统软件和应用程序。

环路 — SPA 和 SP B 对相同编号后端总线的共享连接。

缓冲 — 将数据保存在临时区域中，直到其他设备或流程准备好接收和处理此数据；这样可优化数据流。

缓存 — 存储系统用于缓冲读写数据和隔离主机与大量驱动器访问的内存。

缓存错过 — 在缓存中未找到由主机请求的数据，因此需要驱动器请求。

缓存命中 — 在缓存中找到从主机写入的数据或由主机请求的数据，以避免等待驱动器请求。

缓存页大小 — 一个缓存页的容量。

恢复时间目标 — 在发生故障后将系统恢复到完全操作的估计时间量。

回切 — 在纠正故障后恢复原始数据路径。

活动数据 — 由应用程序寻址的工作数据集。

吉字节 — 十亿字节或一千兆字节。

技能 — 运用知识、所制定属性和/或功能轻松而高效地执行某活动的能力。

技术 — 一种定义的、可使用一个或多个工具的系统化程序，人力、硬件或软件资源借此执行可生成产品、结果或交付服务的活动。

建议请求 — 一种采购文档类型，用于请求潜在卖家提供有关产品或服务的建议。

降级模式 — 当故障后继续操作可能出现性能损失的情况。

降速 — 将非活动硬盘驱动器设置为低功耗的“睡眠”模式。

交换机 — 在端口之间提供专用带宽并且在存储网络设备之间切换功能的第 2 层设备。

结果质量 — 用于评估技术流程或实施的术语。

介质 — 硬盘驱动器用于存储数据的磁盘盘片的磁性表面。

经验法则 — 用于创建估计的指标或计算。

精简 LUN — 其容量不低于主机的可查看容量的逻辑存储单元。

镜像 — 现有数据的复制副本。

局部性 — 应用程序在海量存储中占用的 LBA 的邻近性。

局域网 — 跨小地理区域扩展的计算机网络。

卷 — LUN。

卷配置文件 — 定义从一组磁盘卷构建大段基于文件的存储的标准方法。

决策支持系统 — 用于“数据挖掘”活动的数据库应用程序。

均衡 — 将数据从热备盘拷贝到更换故障 RAID 组驱动器的驱动器。

可靠性 — 产品在特定条件下执行意向功能且持续指定时间段的概率。

可用容量 — 精简 LUN 池中未分配给精简 LUN 的容量。。

可用性 — 基于计算机的系统遇到故障后的持续操作能力。

克隆 — 源 LUN 的完整拷贝。

客户端 — 客户端/服务器体系结构的组成部分；客户端是指与主机通信的用户计算机和应用程序。

客户端/服务器 — 服务使用者（客户端）和提供商（主机）之间的网络体系结构。

控制 — 用于比较实际性能与规划性能、分析差异、评估趋势以改进流程、评估可能的备选方案及按需推荐适当的纠正措施的技术。

快照 — 有关 LUN 在特定时间点的外观的备份拷贝。

利特尔法则 — 稳定系统中用户的长期平均数 L 等于长期平均到达率 λ 乘以用户在系统中消耗的长期平均时间 W ；或者用代数表达： $L = \lambda W$ 。

利于精简 — 在安全或启动期间不预分配容量的应用程序和文件系统。

链路 — 基于计算机的设备或计算机中的设备之间的连接或数据路径。

链路聚合 — 组合具有类似特性且对单个虚拟链路具有相同源和目标的独立链路。

流程 — 为实现指定的产品、结果或服务集合所执行的一组相关操作和活动。

逻辑单元号 — I/O 寻址到的 SCSI 协议实体。

逻辑卷管理器 — 诸如 Microsoft 逻辑磁盘管理器等基于主机的存储虚拟化应用程序。

逻辑块地址 — 驱动器扇区对 SCSI 块地址的映射。

忙时 — 一天中 I/O 最多的小时。

美国国家标准协会 — 国际认可的标准组织。

名称服务器 — 符号和网络地址（包括光纤通道和 IP）间的转换流程。

命令行界面 — 允许用户使用基于文本的命令与应用程序或操作系统通信的界面。

内存模式 — 对线程如何通过内存交互的描述。

拍字节 — 一千万亿字节或一千太字节。

盘架 — DAE。

配电装置 — 将数据中心电源中继连接到存储系统的 CLARiiON 组件。

配置管理接口 — 用于对等存储处理器通信。

平均数据丢失间隔时间 — 对导致 RAID 组丢失数据的故障时间的统计估计。

平均无故障时间 — 设备或系统无故障运行的平均时长。

平均修复时间 — 对故障修复时间的估计。

平台 — 支持某系统的硬件、系统软件和应用程序软件，例如 DSS 或 OLTP。

瓶颈 — 某个流程中以最大能力运行的资源；瓶颈将导致整个流程变慢。

企业级系统 — 支持整个业务组织需求的存储系统。

企业闪存驱动器 — 用于 SSD 类型驱动器的 EMC 术语。

启动器 — iSCSI 客户端。

千位 — 一千位。

千兆赫 — 每秒十亿次 (1,000,000,000 Hz)。

千兆位 — 一千百万位。

千字节 — 一千字节。

前端 — 存储系统体系结构的逻辑分区，包括从主机到 SP 的通信端口。

强制刷新 — 将数据优先写入驱动器以清空完整写缓存。

切片卷 — 在 VNX for File 上，用于创建小型存储单元的卷区域。

请求大小 — 在文件系统中，从驱动器实际读取的数据块大小。

驱动器 — 您可在其中读写数据的硬件组件。通常是硬盘驱动器，但也可以是闪存驱动器。

全球通用名称 — 光纤通道或 SAS 存储网络中的唯一标识符。

热备盘 — 存储系统可用于自动更换故障驱动器的备用驱动器。

日志 — 用于记录和描述或表示流程或活动执行期间标识的选定项的文档或文件。

容错 — 系统在硬件或软件发生故障后继续运行的能力。

冗余 — 存储系统在发生故障后通过使用备份组件或数据保护机制持续服务于数据访问的能力。

闪存驱动器 — 固态磁盘存储设备。

扇出 — 将少量存储系统端口连接到大量主机。

扇区 — 硬盘驱动器中最小的可寻址单元；一个扇区包含 512 字节的数据。

扇入 — 将大量主机连接到少数存储系统端口。

身份认证 — 验证通信身份以确保其声明的来源。

实时数据访问 — 传统 Celerra 操作环境。

授权 — 确定请求是否有权访问资源。

输出 — 由流程生成的流程、结果或服务。可能是对后续流程的输入。

输入 — 流程在继续前所需的内部或外部的任何项或操作。可能是来自先前流程的输出。

数据 — 计算机处理或存储的信息。

数据仓库 — 支持 DSS 功能的相关数据库集合。

数据块 — 硬盘驱动器中最小的可寻址单元；它包含 512 字节的数据。

数据链接 — 一个位置到另一个位置的数字通信连接。

数据挖掘应用程序 — 一种可分析数据库内容以找出数据中的模式、趋势和关系的数据库应用程序。

数据中心 — 用于承载计算机系统、存储系统和关联组件的工具。

刷新 — 将写缓存中的数据写入驱动器。

水位线 — 缓存利用率设定点。

顺序 I/O — 一个 I/O 请求集，其地址和大小模式导致串行访问一个完整区域中地址不断增加的数据。

随机 I/O — 写入到跨文件系统或分区广泛分布的位置的 I/O。

所有权 — LUN I/O 的 SP 管理。

太字节 — 一万亿字节或一千吉字节。

条带大小 — RAID 组条带的可用容量。

条带交叉 — 如果后端 I/O 没有全都包括在一个条带中，由于 I/O 占用的空间超过一个条带，因此会发生条带交叉。

条带卷 — 在 VNX for File 上，多个卷的一种排列方式，使多个卷作为单个卷显示。

条带宽度 — RAID 组条带中硬盘驱动器的数量。

条带元素 — 分配给单一一条带设备的容量。

通用 Internet 文件系统 — Microsoft Windows 文件共享协议。

通用数据块文件系统 — CLARiiON 基于池的 LUN 的文件系统。

图形用户界面 — 允许您使用监视器上的直观对象与软件程序通信的界面。

吞吐量 — 一段时间内的 I/O 性能的测量标准；常以每秒 I/O (IOPS) 为单位。

托架 — DAE。

拓扑 — 系统组件、子系统或系统的部件的排列和内部关联方式。

网络 — 两个或多个相互链接的计算机或基于计算机的系统。

网络接口卡 — 将其连接到以太网网络的主机组件。

网络文件系统 — UNIX/Linux 文件共享协议。

网络元素 — 用于实施网络的设备。通常是指交换机或路由器。

位 — 最小数据单元。它有一个二进制值 0 或 1。

文件 — 数据的集合。

文件存储池 — 用于将可用存储分配到基于 VNX 文件的存储系统的磁盘卷组。

文件服务器 — 通过文件共享协议访问共享存储的 NAS 文件服务器。

文件系统 — OS 用于组织和管理计算机文件的系统。

文件映射协议 — File MPFS 堆栈的组成部分。

无中断更新 — 在最大限度降低应用程序性能影响的前提下，在应用程序运行的同时升级系统软件。

系统 — 由一组定期交互的组件或相互独立的组件组成的集合，为实现规定的目标而创建，成员之间规定并维持着特定关系，整体的产出或操作优于其成员简单相加的和。系统可能基于物理或逻辑流程，更多的情况是这两种流程相结合。

系统软件 — 用于计算机管理的操作系统和应用程序。

线程 — 可与其他请求并行执行的独立 I/O 请求。

线速 — 无协议或软件开销的硬件上的最大数据传输带宽，也称为“线速度”。

响应时间 — 性能的一种测量标准，包括从主机测量的完成 I/O 所需的累计时间。

小型 I/O 卡 — CX4 UltraFlex I/O 模块的通用名称：光纤通道或 iSCSI。

小型计算机系统接口 — 用于在主机和驱动器之间物理连接和传输数据的标准集。

小型数据块 — 最多 16 KB 的 I/O 操作。

协议 — 设备通信的规范。

写缓存 — 专用于向主机提供快速确认，同时在后台将数据转储到磁盘，以此来缩短主机写入 I/O 响应时间的缓存内存。

信息请求 — 一种采购文档类型，买家借此请求潜在卖家提供与产品、服务或卖家能力相关的信息。

虚拟机 — 模拟服务器硬件环境的软件应用程序。

虚拟资源调配 — 将逻辑地址空间显式映射到任意物理地址。例如，向应用程序提供比物理分配更多的容量（基于池的存储）。

旋转延迟 — 磁盘驱动器将所需扇区移到读/写磁头下方所需的时间。

寻道 — 跨磁盘盘片移动硬盘驱动器的读/写磁头。

验证 — 用于在阶段或项目结束时评估组件或产品以确保或确认它满足所要求条件的技术。

要求 — 系统、产品、服务、结果或组件为满足合同、标准、规范或其他正式要求的文档而必须满足或处理的条件或能力。

页 — 缓存分配单元。

已使用容量 — 池中基于池的 LUN 所使用或保留的总容量。

以太网 — 通过局域网的高速带宽连接技术。IEEE 802.3 标准。

应用程序软件 — 执行某种功能的程序或相关程序组。

用户 — 拥有或操作存储产品或应用程序软件的个人或组织。

用户容量 — 可供主机使用的物理或逻辑存储对象的总存储容量。

预读取 — 请参见“**预取**”。

预防措施 — 用于执行可提高可用性并降低故障风险的活动的有记录的指令。

预取 — 提前读取并缓存当前读取之外一定数量的数据块（供以后使用）的缓存方法。

预热 — 活动数据提升到 FAST Cache 中的持续过程。

阈值 — 用作一个参数且可能包括在产品规范中的成本、时间、质量、技术或资源值。超过阈值应触发某些操作。

元数据 — 用于描述或表征其他数据的任何数据。

原始驱动器 — 没有文件系统的硬盘驱动器。

在线事务处理 — 由一个或多个可处理大量小读写操作的数据库所支持的多用户系统。

脏页 — 尚未写入存储的缓存页。

兆赫 — 每秒一百万次循环 (1,000,000 Hz)。

兆位 — 一百万位。

兆字节 — 一百万字节或一千千字节。

阵列 — 存储系统。

质量 — 固有特性集满足要求的程度。

质量保证 — 用于验证所承诺性能和可用性的部门或策略和程序。

重建 — 通过奇偶校验或镜像重新构建故障驱动器的数据。

主动-被动 — 冗余组件准备就绪，且处于备用操作模式。

主动变更 — 因故障或命令而引发的多路径主机启动的 SP LUN 所有权变更。

主动-主动 — 冗余组件处于主动状态且正常操作。

主机 — 通过网络访问存储系统的服务器。

专用 LUN — 由 FLARE 管理且主机不可寻址的 LUN。

转储 — 将缓存内容拷贝到存储区。

转储 — 将数据从缓存移到驱动器。

资源 — 资深人力（精于特定学科的个人、小组或团队）、设备（硬件或软件）、服务、能源、商品、材料、预算（包括持续时间）或资金。

字节 — 八个计算机位。

自动卷管理 — 一种用于自动创建文件系统的 VNX 文件功能。

总线 — 计算机系统中承载设备或组件间数据的内部通道。

组件 — 复杂整体的构成部件、元素或元件。

最大传输单位 — 可通过基于数据包或帧的网络（如 iSCSI SAN）发送的最大[数据包或帧](#)，以字节为单位。

最佳路径 — 从主机到 LUN 的正常操作 I/O 路径

最佳做法 — 一种可使用一个或多个工具和技术以确保最佳地执行某个流程的特定类型专业或管理活动。