



IBM Storwize V7000

CIM 代理程序开发者指南

V6.2.0

G151-1562-00





IBM Storwize V7000

CIM 代理程序开发者指南

V6.2.0

G151-1562-00

注

在使用本资料及其支持的产品之前，请先阅读第 63 页的『声明』中的信息。

本版本适用于 IBM Storwize V7000 的 V6.2.0 以及所有后续发行版和修订版，直到在新版本中另行声明为止。
本版本替代 GC27-2292-00。

© Copyright IBM Corporation 2010, 2011.

目录

图 v

表 vii

关于本指南 ix

本指南面向的读者 ix

更改摘要 ix

| 《Storwize V7000 CIM 代理程序开发者指南》

| (G151-1562-00) 的更改摘要 ix

强调 x

Storwize V7000 库和相关出版物 x

如何订购 IBM 出版物 xi

发送您的意见 xii

第 1 章 CIM 代理程序简介 1

存储管理计划规范 1

CIM 代理程序 1

CIM 代理程序组件 2

CIM 代理程序组件 2

针对 Storwize V7000 的 CIM 代理程序 3

 CIM 代理程序工作方式示例 3

Storwize V7000 CIMOM 编程 6

Storwize V7000 概念如何与 CIM 概念对应 7

公共信息模型代理程序的功能图 8

 访问点的子概要文件 9

 块服务器性能子概要文件 9

 块服务概要文件 11

 块服务的 RAID 支持 13

 集群系统子概要文件 13

 拷贝服务 14

 磁盘驱动器 Lite 概要文件 15

 磁盘驱动器备件 16

 机柜概要文件 18

 扩展数据块组合子概要文件 18

 FC 端口概要文件 19

| Interop 概要文件 20

 iSCSI 关系 21

 作业控制概要文件 23

 位置子概要文件 23

 屏蔽和映射概要文件 24

 多计算机系统概要文件 25

 对象管理器适配器概要文件 26

 物理软件包 27

 服务器概要文件 27

 软件概要文件 28

 存储卷镜像 29

第 2 章 执行存储器配置任务 31

创建新的存储池 31

修改存储池 32

创建新的存储卷 32

创建 RAID 阵列 33

第 3 章 确定拷贝服务关系 35

拷贝服务 35

在存储卷之间创建新的 FlashCopy 关系 35

针对同步集创建 FlashCopy 关系 36

在位于相同集群系统中的卷之间创建同步拷贝关系 36

在位于不同集群系统中的卷之间创建同步拷贝关系 37

创建和启动 FlashCopy 映射的代码样本 38

第 4 章 CIM 代理程序网络注意事项 43

RemoteServiceAccessPoint 实例 43

第 5 章 维护和诊断任务 45

获取类列表 45

获取 UnitTypes 列表 45

获取当前已安装单元的数目 45

使用指示服务 46

获取性能统计信息 46

收集 CIM 日志文件 46

第 6 章 CIM 代理程序类、方法和支持的 SMI-S 概要文件 49

第 7 章 返回码 51

附录. 辅助功能选项 61

声明 63

商标 64

索引 67



1. 工作中的典型 CIM 代理程序	4	16. FC 端口实例的类图	20
2. 集群系统上的 CIMOM	6	17. interop 实例的类图	21
3. 连接至 Storwize V7000 CIMOM 的 Java 程序	7	18. 目标设备特定于 iSCSI 的方面概述	22
4. CIM 代理程序的访问点子概要文件的高级概述	9	19. 作业控制实例的类图	23
5. 块服务器性能子概要文件	10	20. 位置和物理软件包之间的关系概述	24
6. 块服务的高级概述	12	21. 屏蔽和映射实例的类图	25
7. 块服务的 RAID 支持	13	22. 多计算机系统实例的类图	26
8. 集群系统实例的类图	14	23. 对象管理器适配器概要文件的统一建模语言元素概述	27
9. 拷贝服务实例的类图	15	24. CIM 代理程序(针对 Storwize V7000)的物理软件包的高级概述	27
10. 磁盘驱动器 Lite 概要文件的统一建模语言元素概述	16	25. CIM 代理程序(针对 Storwize V7000)的服务器概要文件的高级概述	28
11. 磁盘驱动器备件 CIM 结构	17	26. 软件实例的类图	28
12. IsSpare 关联	17	27. 镜像存储卷中的类和关联	29
13. 发生故障期间的关联	18		
14. 机柜概要文件的元素	18		
15. 扩展数据块组合子概要文件	19		

表

1. Storwize V7000 库.	x	4. CIM 支持的 SMI-S 概要文件和子概要文件	49
2. IBM 文档及相关 Web 站点	xi	5. 公共信息模型返回码.	51
3. Storwize V7000 概念与 CIM 概念之间的关系	7		

关于本指南

本出版物介绍了 IBM® System Storage® Storwize® V7000的公共信息模型 (CIM) 代理程序。

本部分描述以下内容:

- 本书的内容及目标对象
- 要强调的字型
- 关于本书的信息
- 提供有关 Storwize V7000 或相关产品或技术信息的 Web 站点。
- 如何订购 IBM 出版物
- 如何发送您的关于本书的反馈

本指南面向的读者

本参考书供开发公共信息模型的应用程序员使用。

本参考书供要执行以下任务且基于 CIM 的应用程序员使用:

- 了解针对 Storwize V7000 的 CIM 代理程序
- 发现并连接到 CIM 代理程序服务
- 检索并抽取 CIM 代理程序对象类、属性和方法
- 为 Storwize V7000 上的基本存储器配置、逻辑单元号 (LUN) 屏蔽和拷贝服务创建新的对象实例

更改摘要

本文档包含术语、维护以及编辑上的更改。

对文本和插图的技术更改或添加通过在更改左侧加垂直线表示。此更改总结描述已经添加到此发行版的新功能。

| 《**Storwize V7000 CIM 代理程序开发者指南**》(G151-1562-00) 的更改摘要

| 此更改摘要提供了自本指南上一版本以来新的、已修改的和已更改的信息列表。

| 新增信息

| 本主题描述了自上一修订版 *IBM System Storage SAN Volume Controller CIM Agent Developer's Guide* (GC27-2292-00) 之后对该指南的更改。以下部分汇总了自上一版本之后实施的更改。

| 本版本包含以下一般新增信息:

- | • 已添加 interop 概要文件。
- | • 在大多数实例中, 术语“集群”已替换为术语“系统”或“集群系统”。

强调

本指南中的不同字型用于表示强调。

以下字型用来表示强调：

粗体	以粗体显示的文本表示菜单项。
粗体等宽字体	以粗体等宽字体显示的文本表示命令名。
斜体	以斜体显示的文本用于强调某个字。在命令语法中，它用于要提供实际值的变量，如缺省目录或系统名称。
等宽字体	以等宽字体显示的文本标识输入的数据或命令、命令输出样本、程序代码示例或来自系统的消息，或者命令标志名称、参数、自变量以及名称值对。

Storwize V7000 库和相关出版物

产品手册、其他出版物及 Web 站点都包含了与 Storwize V7000 相关的信息。

Storwize V7000 信息中心

IBM Storwize V7000 信息中心包含安装、配置和管理 Storwize V7000 所需的全部信息。在前后两次 Storwize V7000 产品发行版之间，该信息中心都会进行更新，以提供最新的文档。信息中心位于以下 Web 站点：

publib.boulder.ibm.com/infocenter/storwize/ic/index.jsp

Storwize V7000 库

除非另有指明，否则可从位于以下 Web 站点的 Storwize V7000 库中获取这些出版物的 Adobe 可移植文档格式 (PDF)：

Storwize V7000 支持 Web 站点：www.ibm.com/storage/support/storwize/v7000

表 1 中的每个 PDF 出版物都可通过单击“订单号”列中的编号从该信息中心内获取：

表 1. Storwize V7000 库

标题	描述	订单号
<i>IBM Storwize V7000 快速安装指南</i>	该指南提供了有关拆开装运箱和安装系统的指示信息。共有三章，第一章描述了验证您的订单、熟悉硬件组件并满足环境要求。第二章描述了安装硬件并连接数据电缆和电源线。最后一章描述了访问管理 GUI 以对系统进行初始配置。	G151-1337
<i>IBM Storwize V7000 Troubleshooting, Recovery, and Maintenance Guide</i>	该指南描述了如何对 Storwize V7000 系统进行维护、维修和故障诊断。	GC27-2291
<i>IBM Storwize V7000 CIM 代理程序开发者指南</i>	该指南描述了公共信息模型 (CIM) 环境的相关概念。这些过程描述了如下所述的一些任务：使用 CIM 代理程序对象类实例来完成基本存储器配置任务、确立新的拷贝服务关系，以及执行 CIM 代理程序维护和诊断任务。	G151-1562

表 1. Storwize V7000 库 (续)

标题	描述	订单号
<i>IBM Storwize V7000 Safety Notices</i>	该指南包含警告和危险声明的译文。Storwize V7000 文档中的每条警告和危险声明都具有一个编号，用于在 <i>IBM Storwize V7000 Safety Notices</i> 文档中查找您所用语言版本的相应声明。	GC27-3924
<i>IBM Storwize V7000 Read First Flyer</i>	本文档介绍了 Storwize V7000 系统的主要组件，并描述了如何开始使用 <i>IBM Storwize V7000 快速安装指南</i> 。	GC27-2293
<i>IBM System Storage SAN Volume Controller and IBM Storwize V7000 Command-Line Interface User's Guide</i>	该指南描述了可通过 Storwize V7000 命令行界面 (CLI) 使用的命令。	GC27-2287
<i>IBM 环境声明和用户指南</i>	该多语言版指南描述了 IBM 产品所遵守的环境策略，以及如何正确回收和处置 IBM 产品及 IBM 硬件产品中的电池。该指南中的声明描述了平板显示器、制冷、水冷系统和外部电源。	Z125-5823
<i>IBM 有限保证声明</i>	该多语言版文档提供了针对 Storwize V7000 产品的 IBM 保修信息。	部件号: 85Y5978
<i>IBM 机器代码的许可协议</i>	该多语言版指南包含了针对 Storwize V7000 产品的机器代码许可协议。	Z151-0698

IBM 文档及相关 Web 站点

表 2 列出了提供有关 Storwize V7000 或相关产品/技术的出版物及其他信息的 Web 站点。

表 2. IBM 文档及相关 Web 站点

Web 站点	地址
针对 Storwize V7000 (2076) 的支持	Storwize V7000 支持 Web 站点: www.ibm.com/storage/support/storwize/v7000
针对 IBM System Storage 和 IBM TotalStorage 产品的支持	www.ibm.com/storage/support/
IBM Publications Center	www.ibm.com/e-business/linkweb/publications/servlet/pbi.wss
IBM Redbooks® 出版物	www.redbooks.ibm.com/

相关辅助功能选项信息

要查看 PDF 文件，您需要使用可从 Adobe Web 站点下载的 Adobe Acrobat Reader:

www.adobe.com/support/downloads/main.html

如何订购 IBM 出版物

IBM Publications Center 是 IBM 产品出版物和市场营销资料的全球中心资料库。

IBM Publications Center 提供了定制的搜索功能，可帮助您查找所需的出版物。您可以免费查看或下载某些出版物。也可以订购出版物。出版物中心以您本地的货币来显示价格。您可通过以下 Web 站点访问 IBM 出版物中心：

www.ibm.com/e-business/linkweb/publications/servlet/pbi.wss

发送您的意见

您的反馈对于帮助我们提供最准确而优质的信息非常重要。

要提交关于本书或任何其他 Storwize V7000 文档的任何意见，请执行以下操作：

- 请转至位于以下地址的 Storwize V7000 信息中心 Web 站点上的反馈页面：publib.boulder.ibm.com/infocenter/storwize/ic/index.jsp?topic=/com.ibm.storage.Storwize_V7000.console.doc/feedback.htm。您可在此使用反馈页面来输入并提交意见，或者浏览主题并使用该页面中正在运行的页脚内的反馈链接来识别要提供意见的主题。
- 通过电子邮件将您的意见发送至 starpubs@us.ibm.com。包含本出版物的以下信息，或者使用要提供意见的出版物的标题和书号作为替代：
 - 出版物标题：*IBM Storwize V7000 CIM 代理程序开发者指南*
 - 出版物书号：G151-1562-01
 - 要提供意见的页面、表格或插图编号
 - 任何应更改的信息的详细描述

第 1 章 CIM 代理程序简介

公共信息模型 (CIM) 提供一种用于设计和实施存储系统、应用程序、数据库、网络 and 设备的开放式方法。 CIM 代理程序是由分布式管理任务组 (DMTF) 开发的一组标准。

以下信息介绍了存储管理计划规范 (SMI-S)、CIM 代理程序、IBM Storwize V7000 以及针对 Storwize V7000 的 CIM 代理程序。 CIM 代理程序对象模型的功能视图说明了 CIM 代理程序的体系结构和特定功能。

存储管理计划规范

存储管理计划规范 (SMI-S) 是由存储网络行业协会 (SNIA) 发起的存储管理计划 (SMI) 的设计规范。

SMI-S 可指定一个安全且可靠的接口，存储管理系统使用该接口标识、监控和控制存储区域网络 (SAN) 中的物理及逻辑资源并加以分类。该接口集成 SAN 中管理的各种设备和管理这些设备的工具。

SMI-S 基于多种现有的技术或工业标准，包含以下内容：

公共信息模型 (CIM)

由分布式管理任务组 (DMTF) 开发的一种用于数据存储和管理的对象模型。 CIM 使得以面向对象的方式来组织设备和设备组件成为可能。

基于 Web 的企业管理 (WBEM)

同样由 DMTF 开发的一种分层企业管理体系结构。此体系结构提供的管理设计框架由设备、设备供应程序、对象管理器以及用于在客户机应用程序和对象管理器之间通信的消息传递协议组成。对于 CIM，对象管理器为 CIM 对象管理器 (CIMOM)，消息传递协议为 CIM over HTTP 技术。 CIM over HTTP 方法指定 CIM 数据以 XML 编码，并通过 SAN 中的因特网协议网络在客户机应用程序和 CIMOM 之间的特定消息中发送。

服务位置协议 (SLP)

客户机应用程序用于定位 CIMOM 的目录服务。

SMI-S 旨在成为行业标准，它扩展了 CIM、WBEM 和 SLP 的一般功能以实现存储器网络的互操作性。例如，WBEM 提供有关安全性、资源锁定管理、事件通知以及服务发现的条款。

有关针对 Storwize V7000 的 CIM 代理程序所支持的符合 SMI-S 的概要文件和编写基于标准的实施的更多信息，请参阅以下 Web 站点：

www.snia.org/forums/smi/tech_programs/smis_home/

CIM 代理程序

公共信息模型 (CIM) 代理程序是由分布式管理任务组 (DMTF) 开发的一组标准。

CIM 提供了设计和实施存储系统、应用程序、数据库、网络 and 设备的开放式方法。

CIM 规范提供了描述管理数据的语言和方法。例如，用于管理存储阵列的 CIM 模式 2.7 指定了如何以常用方法配置数据管理的管理环境。 CIM 用于定义公共对象类、关联和方法。成员供应商可以使用这些对象并加以扩展，从而指定如何在特定受管环境中处理和组织数据。

CIM 代理程序组件

该信息描述了公共信息模型 (CIM) 代理程序对象模型的基本术语和功能。

CIM 代理程序规范使用以下概念和术语来描述对象模型:

关联 (association)

具有两个引用的类, 它定义两个引用对象之间的关系。

类 (Class)

在特定层次结构中的某个对象的定义。对象类可以具有属性和方法, 并充当关联的目标。

表示 (Indication)

事件的对象表示。

实例 (instance)

成为类成员的单个对象。在面向对象程序设计中, 通过实例化类创建对象。

受管对象格式 (Managed Object Format, MOF)

定义公共信息模型 (CIM) 模式的语言。

方法 (method)

对类实现某一功能的方式。

名称空间 (Namespace)

CIM 模式应用的作用域。

对象路径 (object path)

由名称空间路径和模型路径组成的对象。名称空间路径提供对 CIM 代理程序管理的 CIM 实施的访问, 而模型路径提供实施内的导航。

属性 (property)

用来表现类实例的特征的属性。

限定符 (qualifier)

一个值, 该值提供了关于类、关联、指示、方法、方法参数、实例、属性或引用的附加信息。

引用 (Reference)

指向另一实例的指针, 它定义了关联中某对象的角色和作用域。

模式 (Schema)

为单独名称空间定义并适用于它的一组对象类。在 CIM 代理程序中, 受支持的模式是通过受管对象格式化 (MOF) 编译器装入的。

CIM 代理程序组件

通过公共信息模型 (CIM) 代理程序, 应用程序员可以使用公共构建块 (而不是专有软件或特定于设备的编程界面) 来管理与 CIM 兼容的设备。应用程序管理存储器方式的标准化使存储器管理更为容易。

组件

CIM 代理程序包含以下组件:

代理程序代码

一种开放系统标准, 用于当 CIM 请求和响应在客户机应用程序和设备之间传送时对它们进行解释。

CIM 对象管理器 (CIMOM)

用于数据管理的公共概念性框架，它接收、确认并认证来自客户机应用程序的 CIM 请求。然后，它引导请求到合适的组件或设备提供程序。Storwize V7000 基于 Open Pegasus V2.9.0 CIMOM。

客户机应用程序

存储管理程序，它为设备发出 CIM 请求到 CIM 代理程序。

设备 存储服务器，它处理并管理客户机应用程序请求。

设备供应程序

特定于设备的处理程序，充当 CIM 的插件。即 CIMOM 使用该处理程序与设备进行交互。

服务位置协议 (SLP)

客户机应用程序调用以定位 CIMOM 的目录服务。

针对 Storwize V7000 的 CIM 代理程序

针对 Storwize V7000 的公共信息模型 (CIM) 代理程序充当 Storwize V7000 的配置界面。

CIM 代理程序包含以下主要组件：

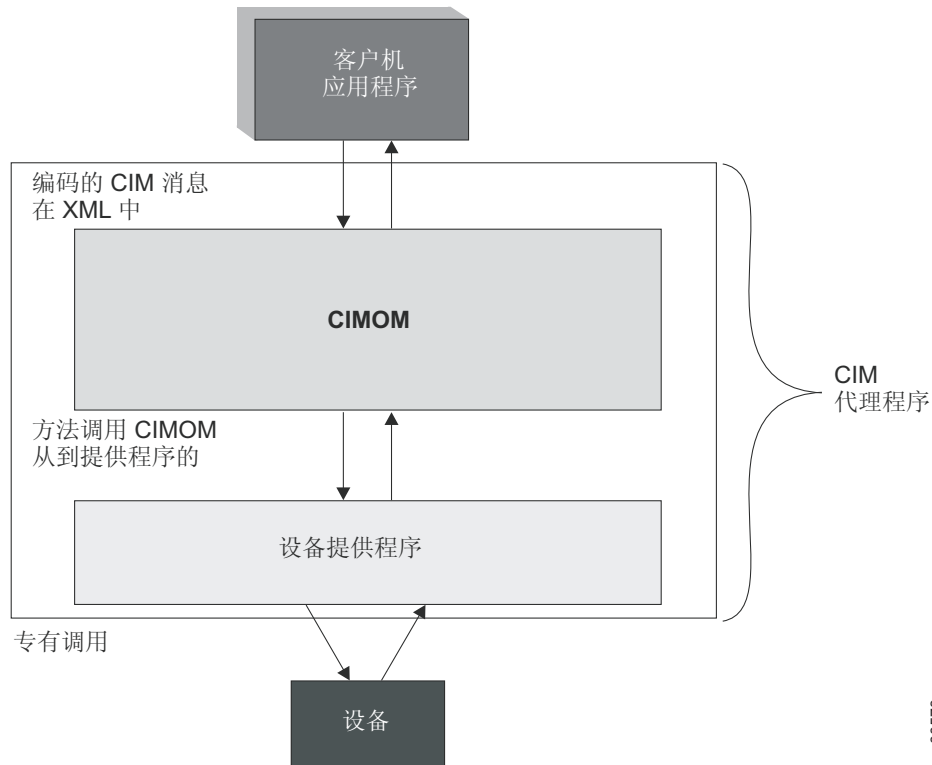
- CIM 对象管理器 (CIMOM)
- 服务位置协议 (SLP) 代理程序
- Storwize V7000 提供程序

CIM 代理程序工作方式示例

公共信息模型 (CIM) 代理程序可用于提供公共构建块，以管理与 CIM 兼容的设备。

典型 CIM 代理程序配置

第 4 页的图 1 显示了典型 CIM 代理程序如何工作。



svc00573

图 1. 工作中的典型 CIM 代理程序

CIM 客户机的典型配置是使用位于 `https://cluster_ip` 端口 5989 并使用 `namespace=/root/ibm` 的 CIMOM。标准安全端口为 5989。您也可以使用 `namespace=interop`，这是标准 SMI-S 名称空间。

客户机应用程序向 CIMOM 发送 CIM 请求。当请求到达 CIMOM 时，CIMOM 将对每一个请求进行验证和授权。然后，它将请求引导到合适的功能性 CIMOM 组件或设备提供程序。要满足客户机应用程序的请求，提供程序会代表 CIMOM 调用针对设备唯一的编程接口。

管理应用程序可以从 CIMOM 获得 `RemoteServiceAccessPoint` 的实例。该实例由管理应用程序用于访问 Web 用户界面。

使用服务位置协议的 CIM 代理程序配置

如果使用服务位置协议 (SLP) 来发现 CIMOM，那么客户机应用程序会通过调用 SLP 目录服务来查找该 CIMOM。当首次调用 CIMOM 时，它会向 SLP 服务代理程序注册自身并提供其位置、IP 地址、端口号以及其提供的服务类型。这样会注册一个描述 CIM 代理程序访问点的字符串。

以下输出提供了一个注册字符串示例：

```
service:wbem:https://<CIM Agent IP>:<port number>
```

SLP 提供以下属性：

```
template-type=wbem
template-version=1.0
template-description=This template describes the attributes used for
advertising WBEM servers.
template-url-syntax=https://9.47.24.91:5989
service-location-tcp=https://9.47.24.91:5989
service-hi-name=IBM Storwize(r)
V7000 CIMOM
```

```

service-hi-description=IBM Storwize
| V7000 CIM Agent Version 6.2.0.xxx
service-id=IBMTSSVC:9.47.24.91
ProtocolVersion=1.2
CommunicationMechanism=cim-xml
FunctionalProfilesSupported=Basic Read, Basic Write, Instance Manipulation,
Association, Traversal, Query Execution, Qualifier Declaration, Indications
AuthenticationMechanismSupported=Basic
Namespace=/root/ibm
| Interop=/root/ibm
MultipleOperationsSupported=false
RegisteredProfilesSupported=SNIA:Storage Virtualizer,SNIA:Storage Virtualizer:
Access Points,SNIA:Storage Virtualizer:Block Services,SNIA:Storage Virtualizer:
Cascading,SNIA:Storage Virtualizer:Copy Services,SNIA:Storage Virtualizer:
FC Initiator Ports,SNIA:Storage Virtualizer:FC Target Ports,SNIA:Storage
Virtualizer:Health,SNIA:Storage Virtualizer:Masking and Mapping,SNIA:Storage
Virtualizer:Multiple Computer System,SNIA:Storage Virtualizer:Physical Package,
| SNIA:Storage Virtualizer:Software,SNIA:Storage Virtualizer:Location,SNIA:
| Storage Virtualizer:Block Server Performance,SNIA:Storage Virtualizer:iSCSI
| Target Ports,SNIA:Storage Virtualizer:Disk Drive Lite,SNIA:Storage Virtualizer:
| Extent Composition,SNIA:Storage Virtualizer:Disk Sparing,SNIA:Server,SNIA:Server:
| Profile Registration,SNIA:Server:Object Manager Adapter,SNIA:Server:Indication,
SNIA:SMI-S

```

根据该信息，客户机应用程序开始与 CIMOM 直接通信。

控制台和集群系统上的 CIM 代理程序配置

- | CIMOM 位于 Storwize V7000 集群系统上。管理 GUI 不使用位于集群系统上的 CIMOM。其他 IBM 存储管理工具继续使用位于集群系统上的 CIMOM。

第 6 页的图 2 显示了如何定义该 CIMOM 配置。

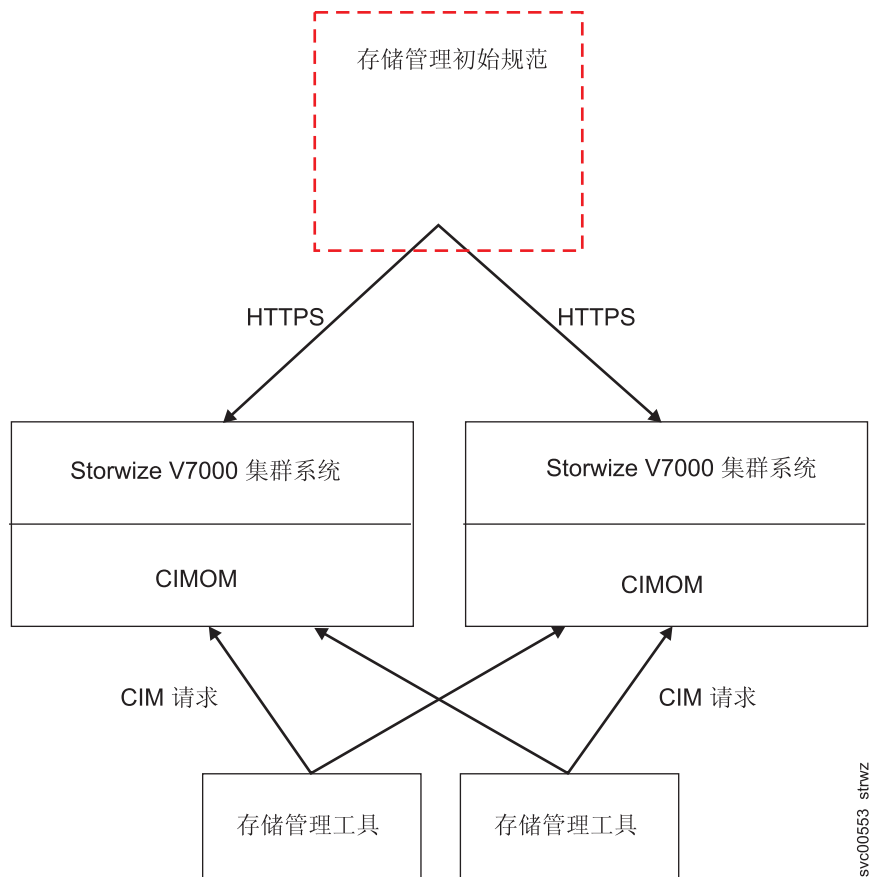


图 2. 集群系统上的 CIMOM

Storwize V7000 CIMOM 编程

CIM 对象管理器 (CIMOM) 编程在维护安全性的同时，还可处理来自多个源的多个连接。CIM 客户机通过用户名和密码连接至 CIMOM，然后调用运行命令的方法。

- | 创建 CIM 客户机需要适当的框架，例如 Java WBEM Service 项目和 SBLIM CIM Client for Java。同时还
- | 有采用其他语言的一些实施，包括 C++ 和 Python 语言。有关更多信息，请参阅 Java WBEM Services Web
- | 站点：

wbemservices.sourceforge.net/

第 7 页的图 3 显示了连接至 Storwize V7000 CIMOM 的一个简单 Java 程序。

```

import java.util.*;

import javax.wbem.cim.*;
import javax.wbem.client.*;

public class ITSOClient {
    public static void main(String[] args)
    {
        String username = args[0];
        String password = args[1];
        String masterConsoleIP = args[2];
        String masterConsoleSecurePort = args[3];
        UserPrincipal user = new UserPrincipal(username);
        PasswordCredential pwd = new PasswordCredential(password);
        CIMNameSpace ns = new CIMNameSpace("https://" +
            masterConsoleIP + ":" +
            masterConsoleSecurePort + "/root/ibm");

        CIMClient client = null;
        try
        {
            System.out.println("Connecting to CIMOM");
            client = new CIMClient(ns,user,pwd);
        }
        catch (CIMException e)
        {
            // Handle the CIM Exception
            e.printStackTrace();
        }
    }
}

```

图 3. 连接至 Storwize V7000 CIMOM 的 Java 程序

- 要查看完整的受管对象格式 (MOF) 文档，包括 CIM 代理程序样式页面，请从以下 Web 站点中选择文档信息：

Storwize V7000 支持 Web 站点：www.ibm.com/storage/support/storwize/v7000

Storwize V7000 概念如何与 CIM 概念对应

要通过 CIM 对象管理器 (CIMOM) 管理 Storwize V7000，了解 Storwize V7000 和公共信息模型 (CIM) 概念之间的关系至关重要。

表 3 显示了这些概念之间的相互关系。

表 3. Storwize V7000 概念与 CIM 概念之间的关系

Storwize V7000 概念	CIM	
	CIM 名称	CIM 概念
系统	IBMTSSVC_Cluster	类
系统名称	ElementName	属性
系统标识	Name	属性
VDisk	IBMTSSVC_StorageVolume	类
VDisk 标识	DeviceID	属性
FlashCopy® 一致性组 (常规)	IBMTSSVC_FlashCopyStorageSynchronizedSet	类
FlashCopy 一致性组 (自动删除)	IBMTSSVC_CloneCopyStorageSynchronizedSet	类
FlashCopy 映射	IBMTSSVC_LocalStorageSynchronized	关联

表 3. Storwize V7000 概念与 CIM 概念之间的关系 (续)

Storwize V7000 概念	CIM	
	CIM 名称	CIM 概念
FlashCopy 映射状态	SyncState	属性
mkfemap	AttachReplica	方法
preparefcmap	ModifySynchronization	方法
startfcmap	ModifySynchronization	方法
远程拷贝一致性组 (全局镜像)	IBMTSSVC_AsyncCopyStorageSynchronizedSet	类
远程拷贝一致性组 (高速镜像)	IBMTSSVC_SyncCopyStorageSynchronizedSet	类
远程拷贝关系	IBMTSSVC_RemoteStorageSynchronized	关联
远程拷贝关系状态	NativeState	属性
mkrcrelationship	AttachReplica	方法
startrcrelationship	ModifySynchronization	方法
mdisk	IBMTSSVC_BackendVolume	类
mdiskgrp	IBMTSSVC_ConcreteStoragePool	类
mkvdisk	CreateOrModifyElementFromStoragePool	方法
mkmdiskgrp	CreateOrModifyStoragePool	方法
rmvdisk	ReturnToStoragePool	方法
rmmdiskgrp	DeleteStoragePool	方法
主机 (与端口相关)	IBMTSSVC_StorageHardwareID	类
主机 (与卷 (VDisk) 映射有关)	IBMTSSVC_ProtocolController	类
mkvdiskhostmap	ExposePaths	方法
rmvdiskhostmap	HidePaths	方法
mkhost	CreateStorageHardwareID	方法
rmhost	DeleteStorageHardwareID	方法
磁盘驱动器	IBMTSSVC_DiskDrive	类
阵列	IBMTSSVC_Array	类
mkarray	CreateOrModifyElementFromElements	方法
charray	CreateOrModifyElementFromElements	方法
rmarray	DeleteArray	方法
机柜	IBMTSSVC_Enclosure	类

公共信息模型代理程序的功能图

公共信息模型 (CIM) 代理程序对象模型的功能图显示了 CIM 代理程序提供的特定功能，包含存储器配置服务、拷贝服务、LUN 屏蔽和安全性。该图还说明了针对 Storwize V7000 的 CIM 代理程序的体系结构。

访问点的子概要文件

针对 Storwize V7000 的公共信息模型 (CIM) 代理程序的访问点子概要文件包含一些基本类。

图 4 显示了该模型的基本类 (构建块)。

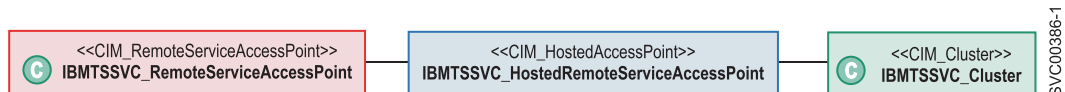
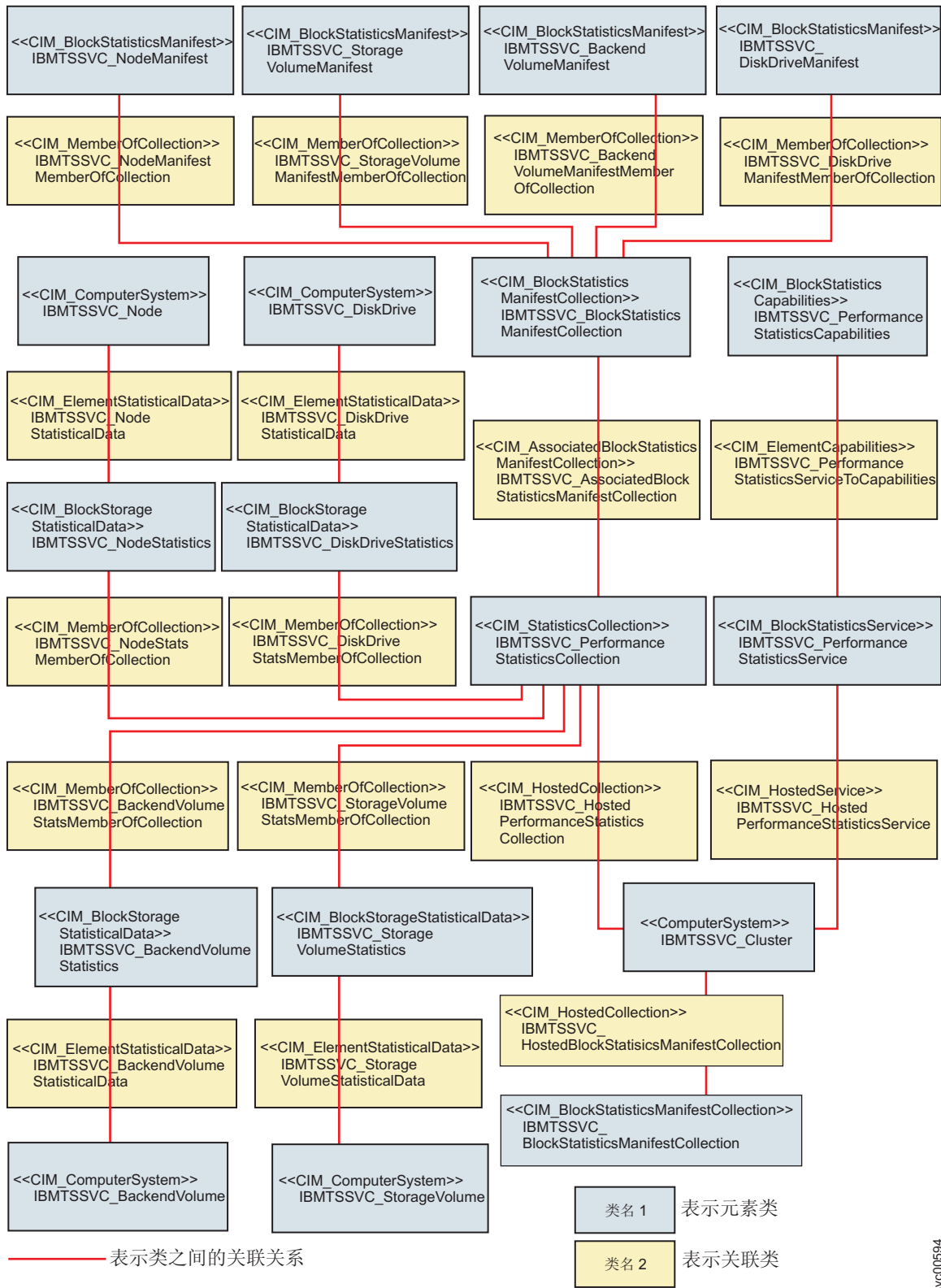


图 4. CIM 代理程序的访问点子概要文件的高级概述

块服务器性能子概要文件

块服务器性能子概要文件描述了性能统计信息。

第 10 页的图 5 描述了 IBMTSSVC_Node、IBMTSSVC_StorageVolume、IBMTSSVC_BackendVolume 和 IBMTSSVC_DiskDrive 的性能统计信息。



svc00594

图 5. 块服务器性能子概要文件

块服务概要文件

您可以使用多个对象类来处理存储池。

块服务概要文件将后端存储卷分配到存储池中，然后再创建存储卷。

第 12 页的图 6 提供了可用于池处理的对象类的高级概述。

块服务的 RAID 支持

RAID 阵列类提供对块服务概要文件的支持。

该图说明了有关块服务概要文件中 RAID（阵列）功能的类及其关系。

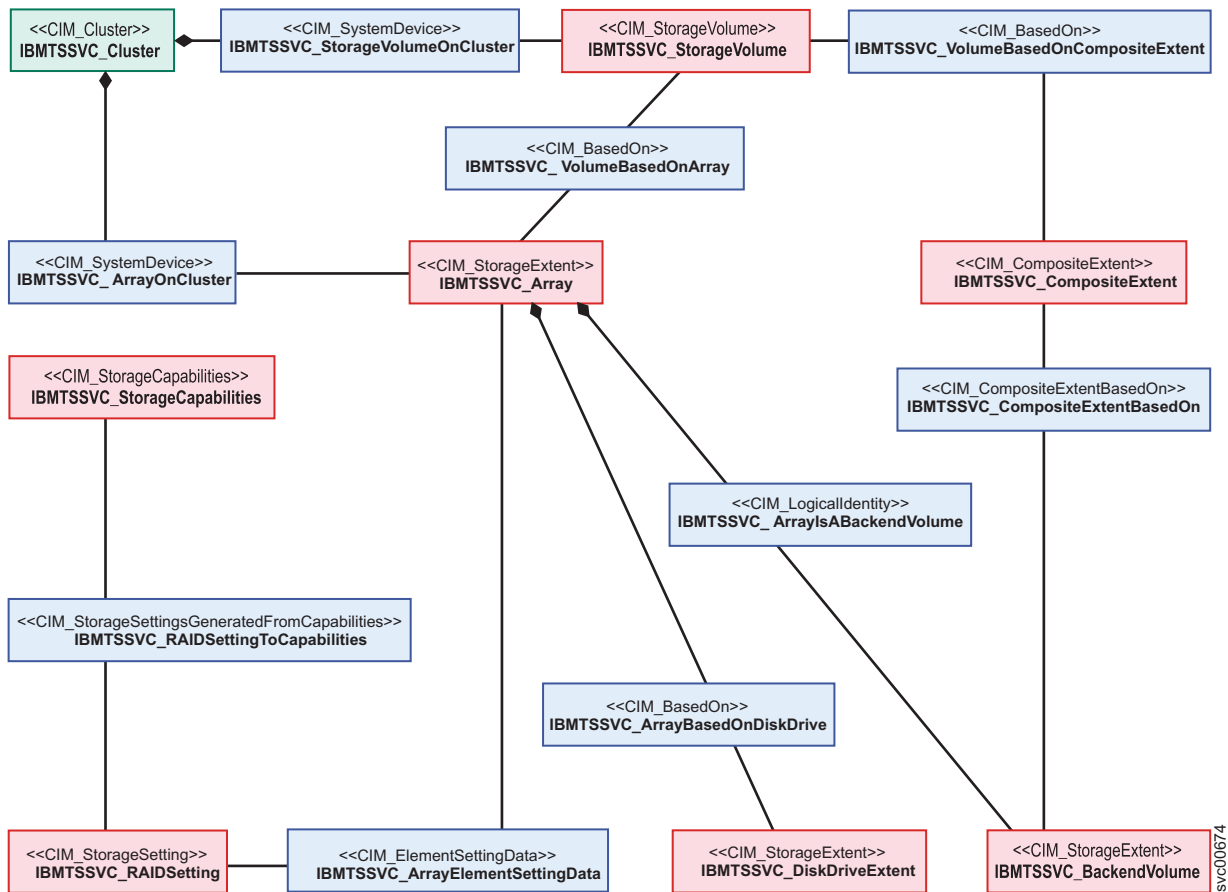


图 7. 块服务的 RAID 支持

集群系统子概要文件

多种类和关联都提供集群系统服务功能。

第 14 页的图 8 显示了提供集群系统服务功能的类和关联。

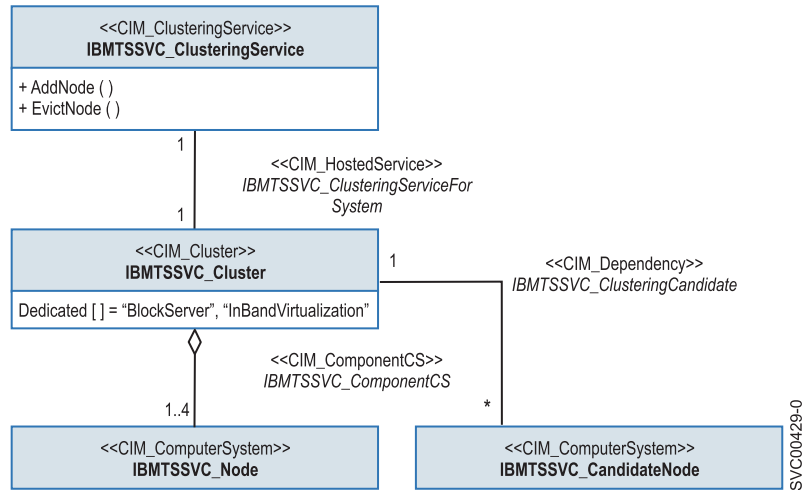


图 8. 集群系统实例的类图

拷贝服务

IBMTSSVC_StorageConfigurationService 类提供用于创建拷贝关系的方法。

第 15 页的图 9 显示了提供 FlashCopy、高速镜像和全局镜像服务的对象类。

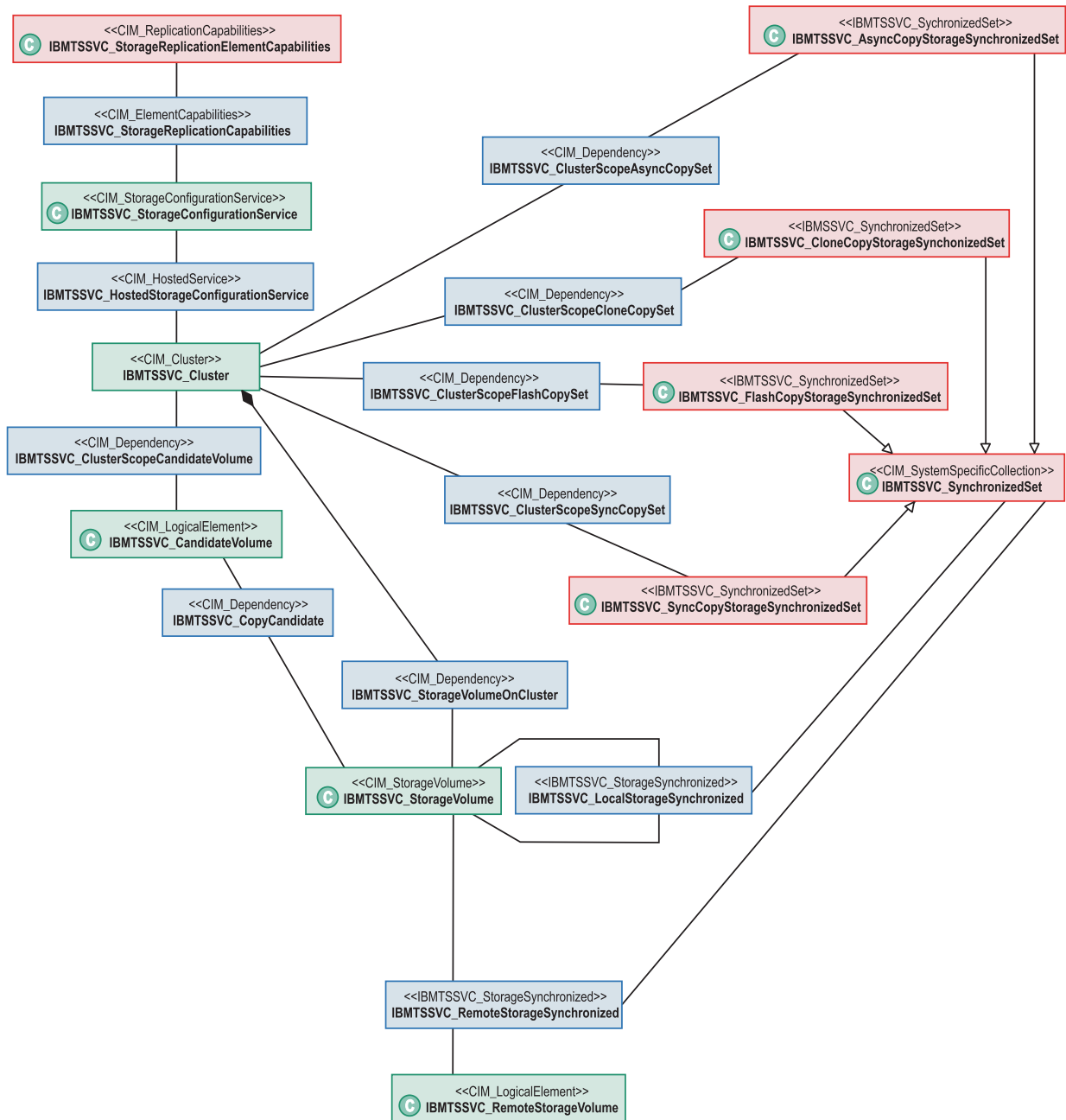


图 9. 拷贝服务实例的类图

磁盘驱动器 Lite 概要文件

磁盘驱动器是磁盘驱动器 Lite 概要文件的主要硬件元素。

第 16 页的图 10 显示了磁盘驱动器 Lite 概要文件的基本元素。

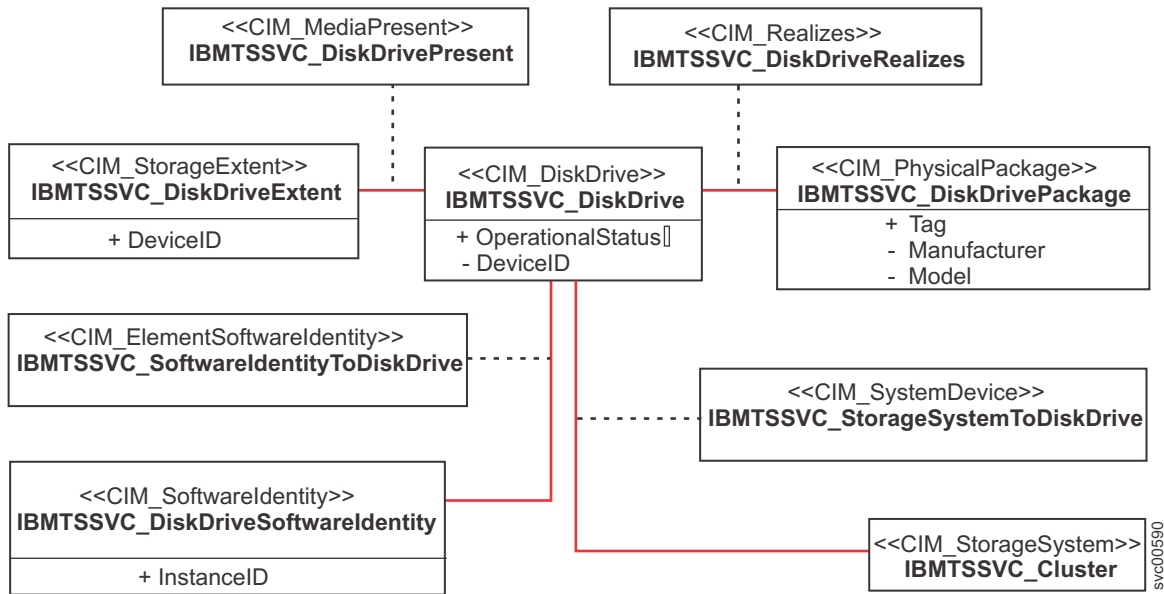


图 10. 磁盘驱动器 Lite 概要文件的统一建模语言元素概述

磁盘驱动器备件

磁盘驱动器备件具有与 RedundancySet 关联的基本结构。

磁盘驱动器备件 RedundancySet 结构由一个实例类和三个关联类组成：

实例类：

- IBMTSSVC_StorageRedundancySet (扩展自 CIM_StorageRedundancySet)

关联类：

- IBMTSSVC_StoragePoolExtentDependency 表示 IBMTSSVC_DiskDriveExtent 和 IBMTSSVC_ConcreteStoragePool 之间的关联。
- IBMTSSVC_HostedRedundancySet 表示 IBMTSSVC_StorageRedundancySet 和 IBMTSSVC_Cluster 之间的关联。
- IBMTSSVC_MemberOfRedundancySet 表示 IBMTSSVC_DiskDriveExtent 和 IBMTSSVC_StorageRedundancySet 之间的关联。 DiskDriveExtents 形成了 StoragePool 的冗余集。

磁盘驱动器备件 CIM 结构

磁盘驱动器备件 CIM 结构主要由 DiskDriveExtent 和 StorageRedundancySet 组成。

第 17 页的图 11 中显示了各种组件之间的相互关系。

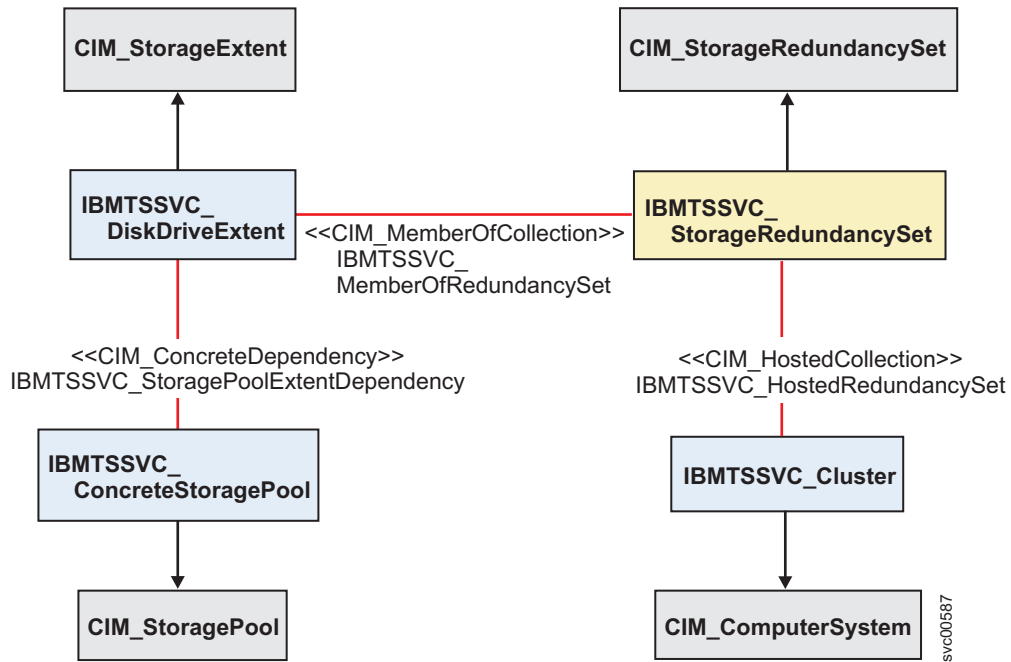


图 11. 磁盘驱动器备件 CIM 结构

备用磁盘驱动器扩展数据块

DiskDriveExtents 用于失败的实现。

用于替换实现失败的 DiskDriveExtent 的 DiskDriveExtents 通过 IsSpare 关联与 RedundancySet 相关联，如图 12 所示。

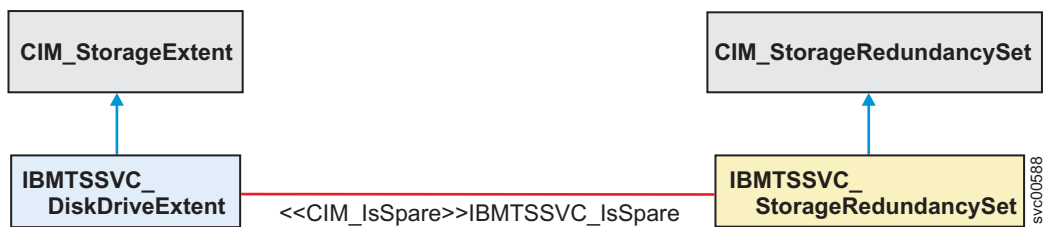


图 12. IsSpare 关联

类 IBMTSSVC_IsSpare 表示用作备件的 IBMTSSVC_DiskDriveExtent 与 IBMTSSVC_StorageRedundancySet 之间的关联。

磁盘驱动器故障关联

可为发生故障的 DiskDriveExtent 提供备件。

关联的 DiskDriveExtents 可作为备件或更换其他可能发生故障的 DiskDriveExtents，如第 18 页的图 13 所示。

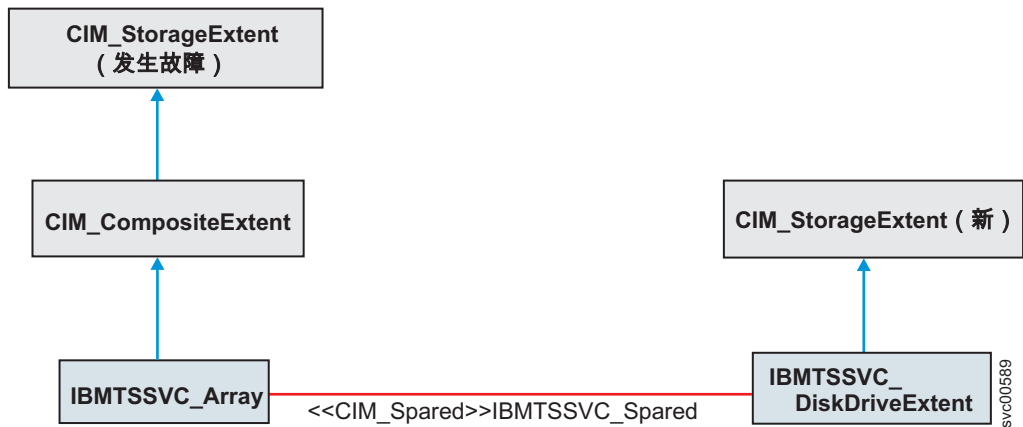


图 13. 发生故障期间的关联

在新的 DiskDriveExtent 和要重新构建的阵列之间的该关系中，将使用新的 DiskDriveExtent 作为发生故障的 DiskDriveExtent 的备件。 IBMTSSVC_Spared 关联将保持不变，直到备用过程完成。

机柜概要文件

机柜是机柜概要文件的主要硬件元素。

图 14 显示了机柜概要文件的基本元素。

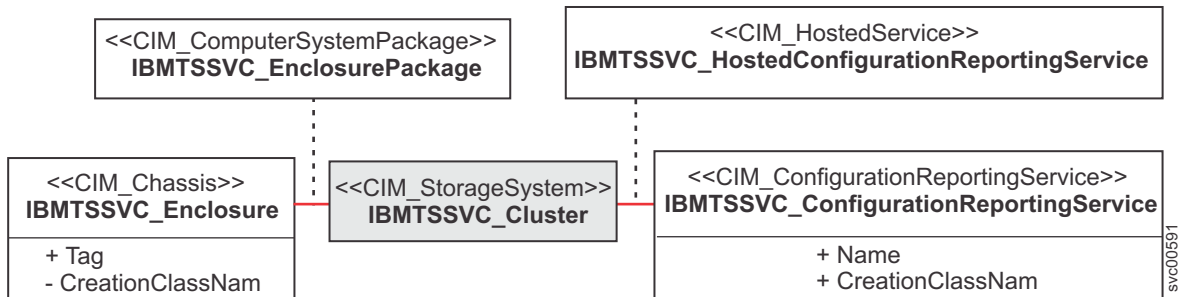


图 14. 机柜概要文件的元素

扩展数据块组合子概要文件

扩展数据块组合子概要文件链接 CompositeExtents 和 StorageExtents。

扩展数据块组合子概要文件由块服务软件包选择用于虚拟化来自底层原始存储池的可呈现块存储元素。抽象表示为扩展数据块的代表性层次结构。第 19 页的图 15 表示了磁盘驱动器 Lite、扩展数据块组合和阵列子概要文件之间的关系。

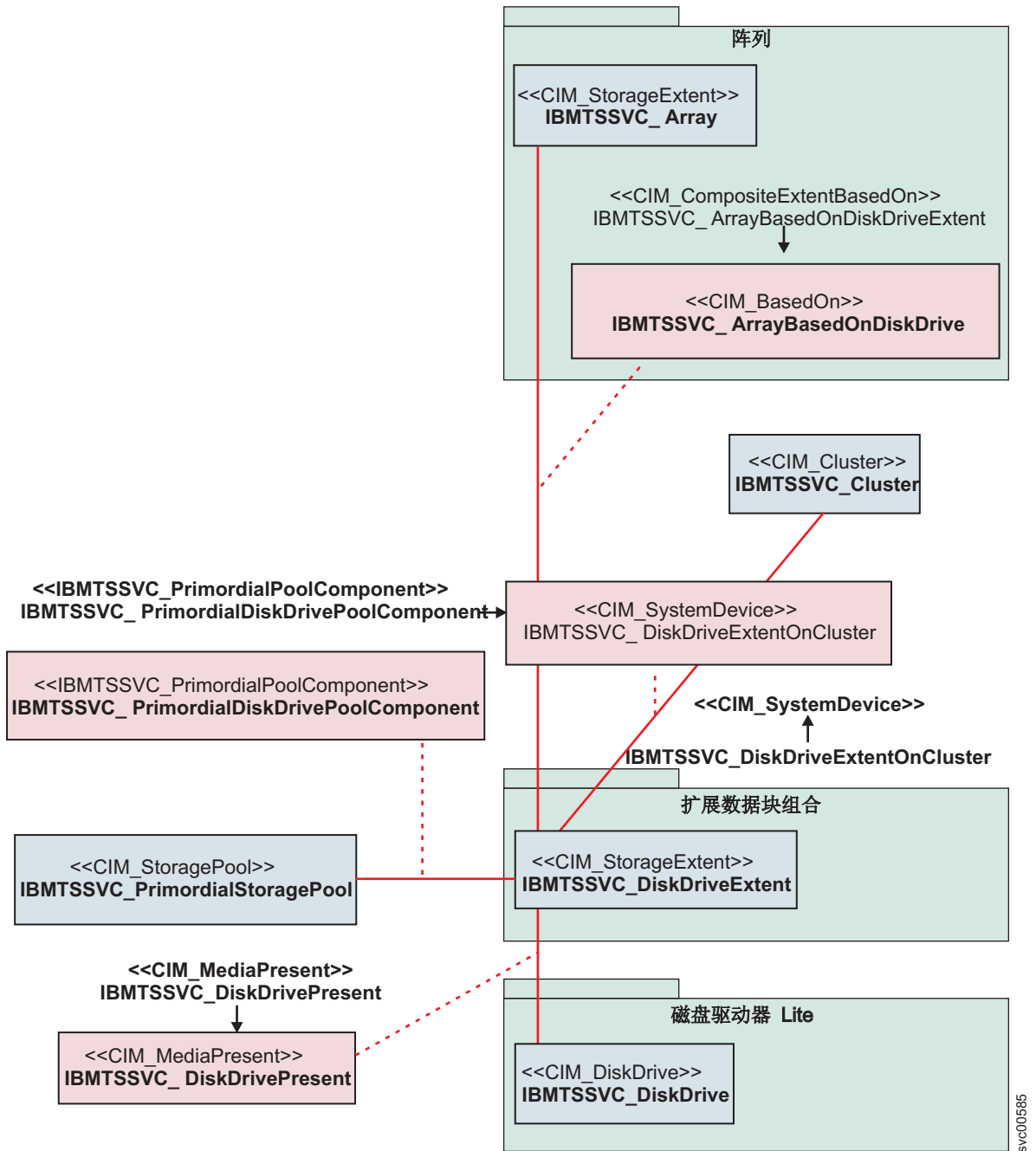


图 15. 扩展数据块组合子概要文件

这些扩展数据块是通过 CompositeExtentBasedOn 和 BasedOn 关联的组合而链接的 CompositeExtents 和 StorageExtents 的实例。在图 15 中，蓝色的块表示类，红色的块表示关联。

FC 端口概要文件

FC 端口概要文件用于对 Storwize V7000 和 Storwize V7000 虚拟化的后端存储器之间的光纤通道连接关系进行建模。

第 20 页的图 16 显示了 FC 端口概要文件的类和关联。

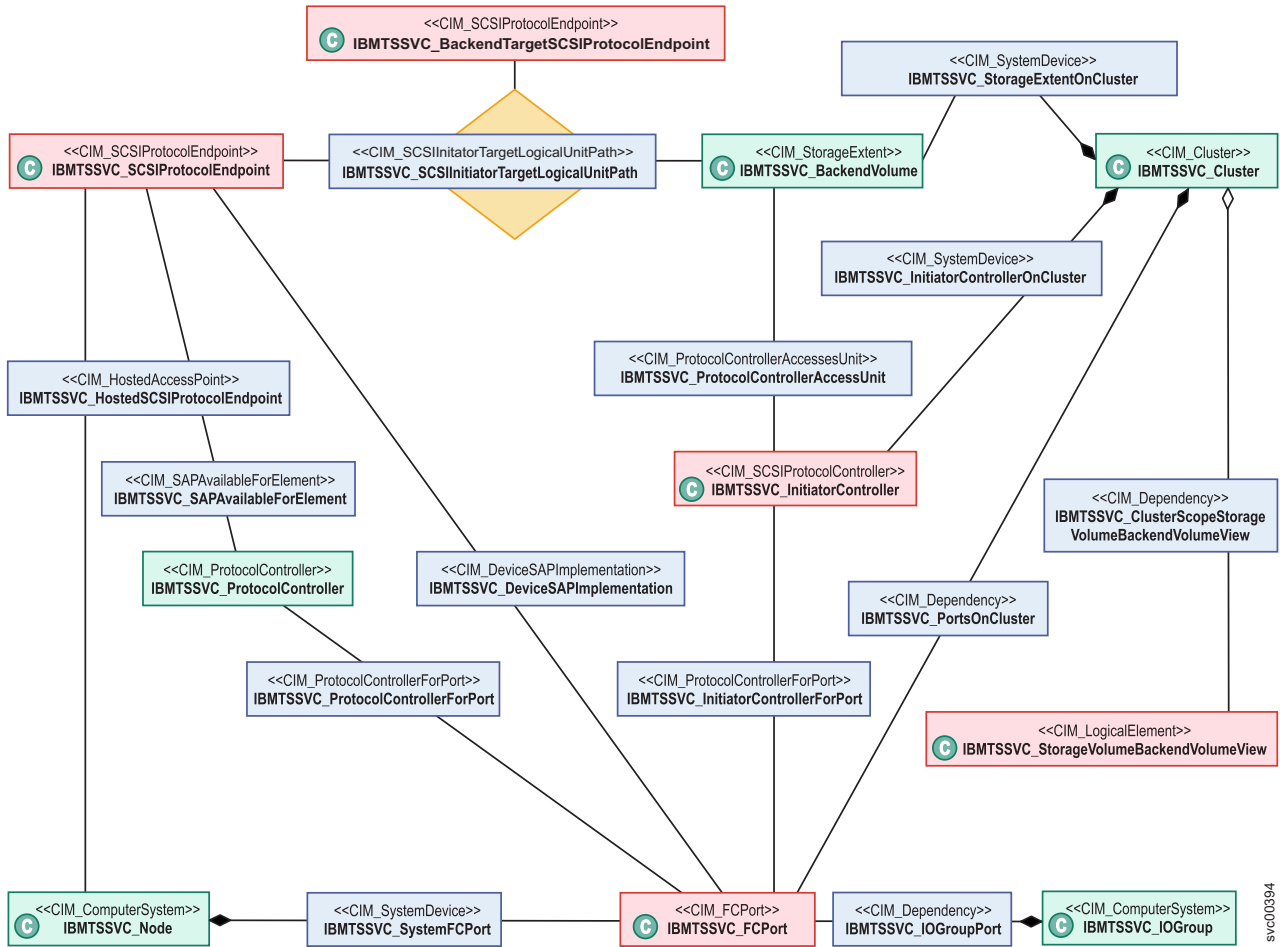


图 16. FC 端口实例的类图

1 Interop 概要文件

1 interop 概要文件包含的类可用于查找来自 interop 名称空间的注册概要文件。

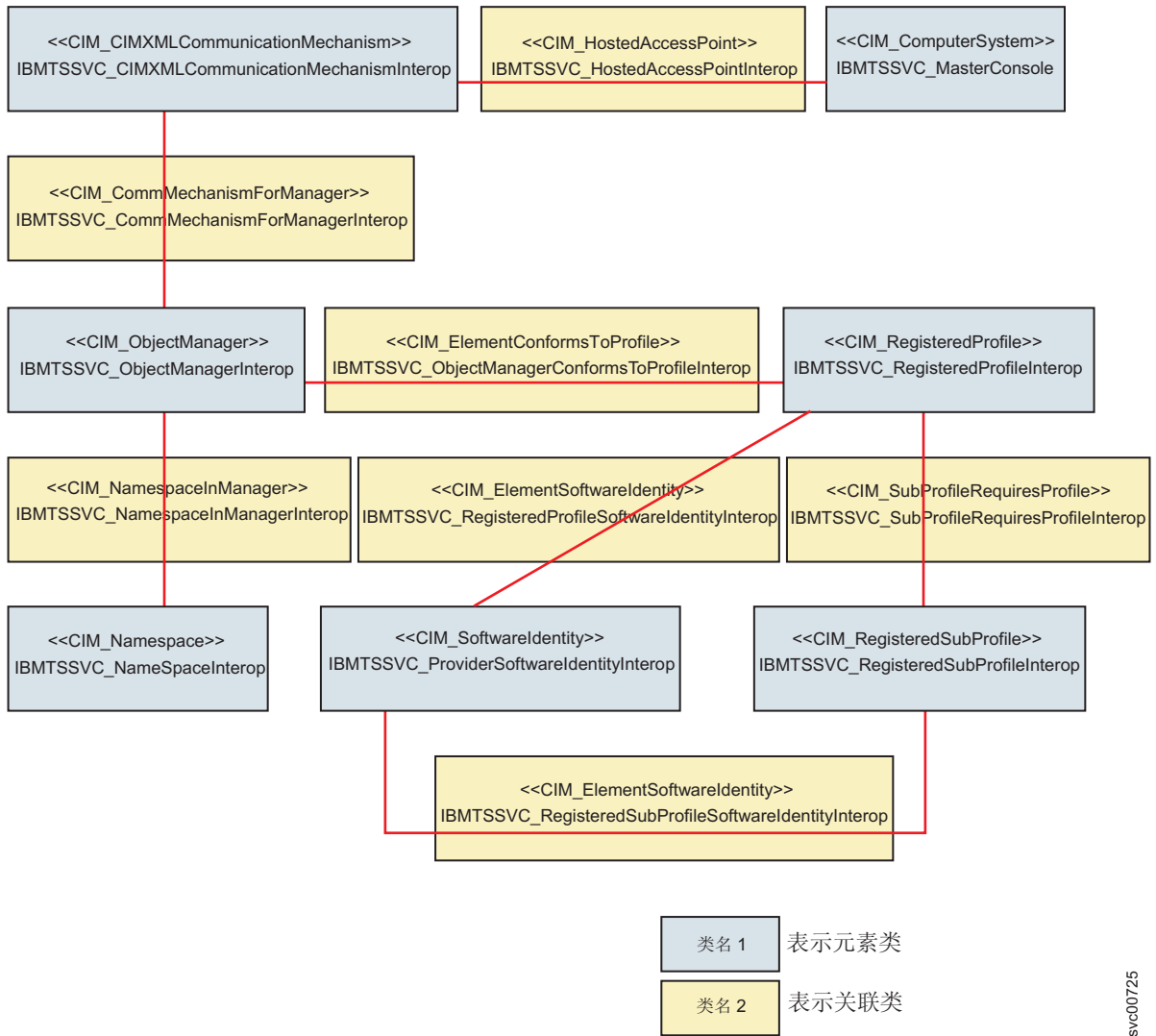
1 第 21 页的图 17 显示了 interop 概要文件的类和关联。在该概要文件中，IBMTSSVC_HostedAccessPointInterop

1 表示 IBMTSSVC_CIMXMLCommunicationMechanismInterop 和 IBMTSSVC_MasterConsole 的关系。

1 IBMTSSVC_CIMXMLCommunicationMechanismInterop 是在 interop 名称空间中注册的。

1 IBMTSSVC_MasterConsole 是在 root/ibm 名称空间中注册的。

1



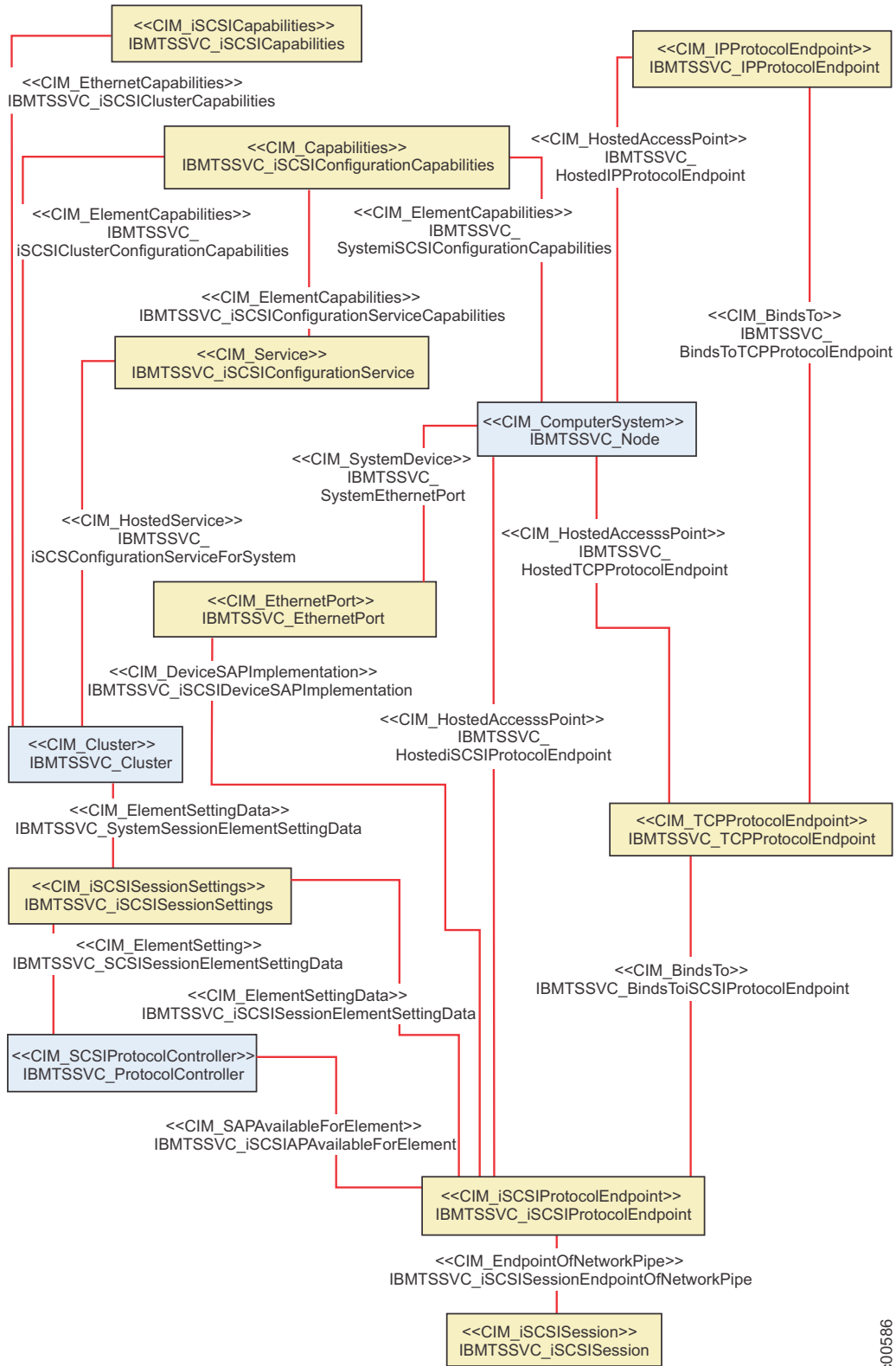
svc00725

图 17. interop 实例的类图

iSCSI 关系

目标设备特定于 iSCSI 的方面包括实例和关联类。

第 22 页的图 18 显示了目标设备特定于 iSCSI 的方面。



svc00586

图 18. 目标设备特定于 iSCSI 的方面概述

在图 18 中，显示为黄色的类和相关关联现已受支持。

作业控制概要文件

作业控制概要文件包含的类可用于监控对设备执行格式化、迁移或运行拷贝操作的异步命令。

图 19 显示了作业控制概要文件的类和关联。

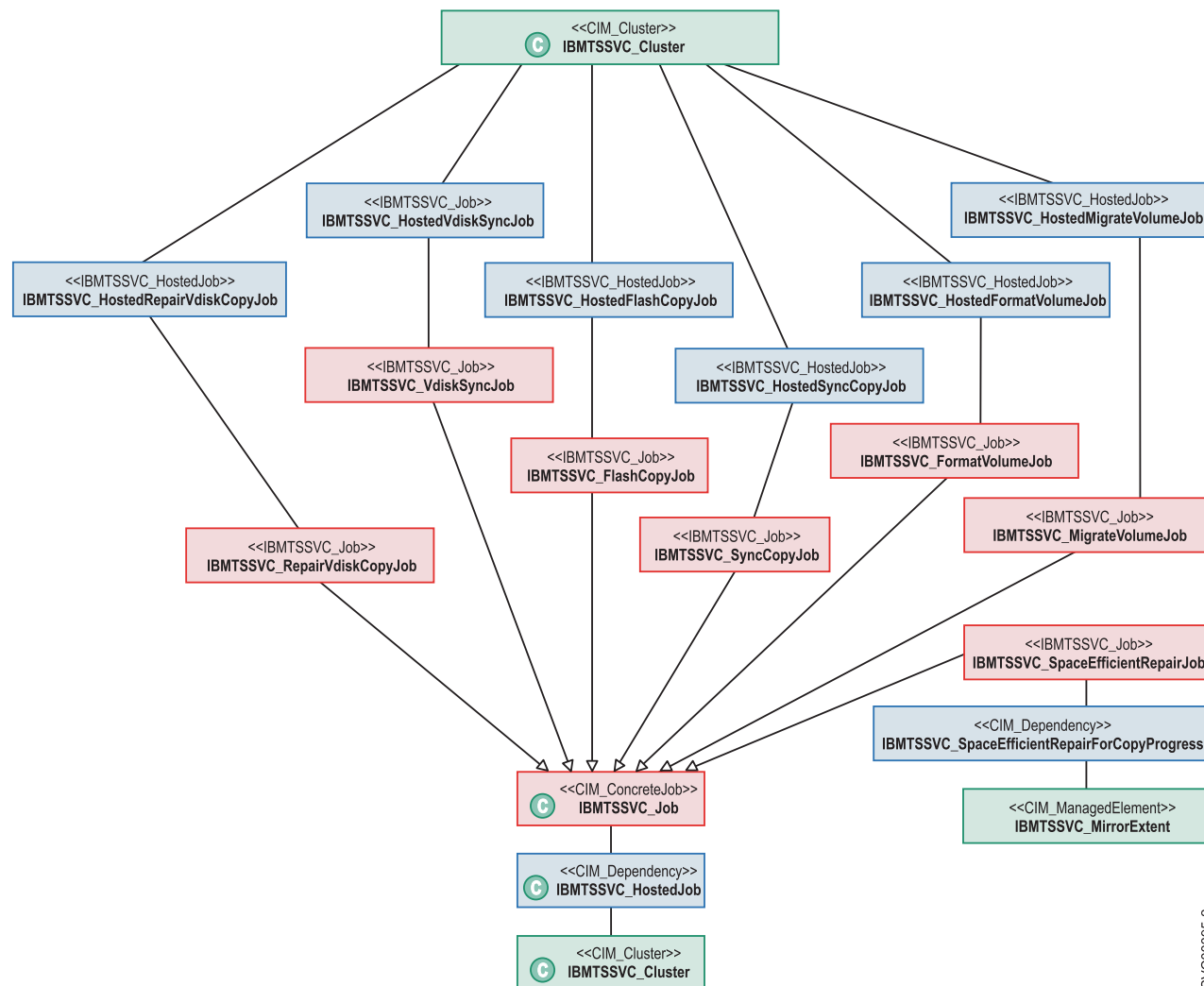
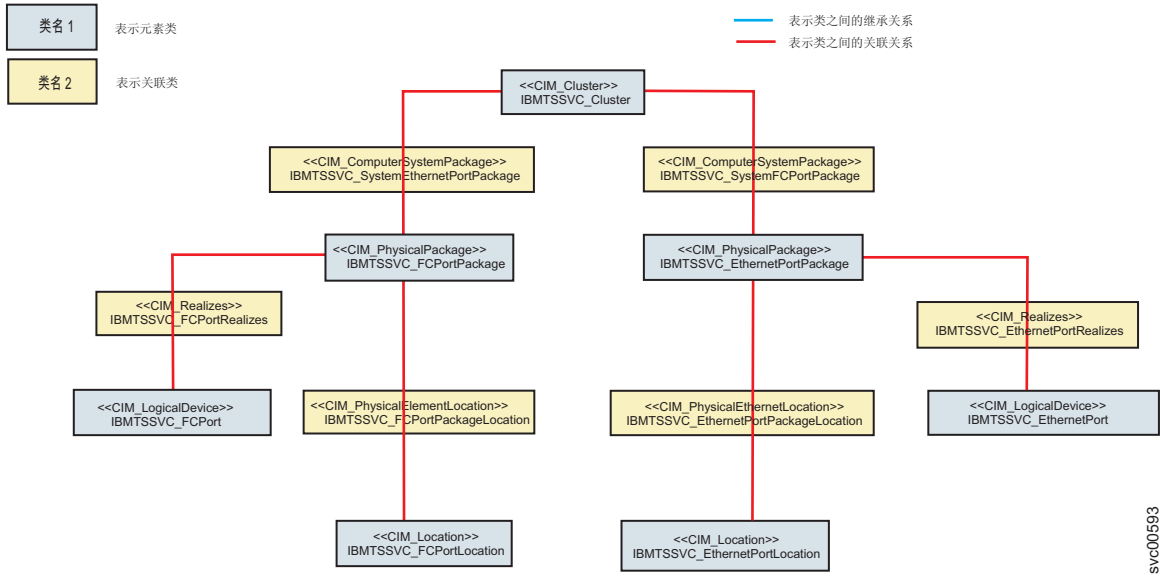


图 19. 作业控制实例的类图

位置子概要文件

位置子概要文件表示位置和物理软件包之间的关系。

第 24 页的图 20 显示了位置子概要文件的基本元素。



svc00593

图 20. 位置和物理软件包之间的关系概述

在图 20 中，黄色块中的类表示关联类。例如，IBMTSSVC_FCPortRealizes 是 IBMTSSVC_FCPort 和 IBMTSSVC_FCPortPackage 之间的关联。蓝色块中的类表示元素类。

屏蔽和映射概要文件

屏蔽和映射概要文件提供了一个创建、修改、删除和屏蔽主机的接口。

第 25 页的图 21 显示了屏蔽和映射概要文件的类和关联。

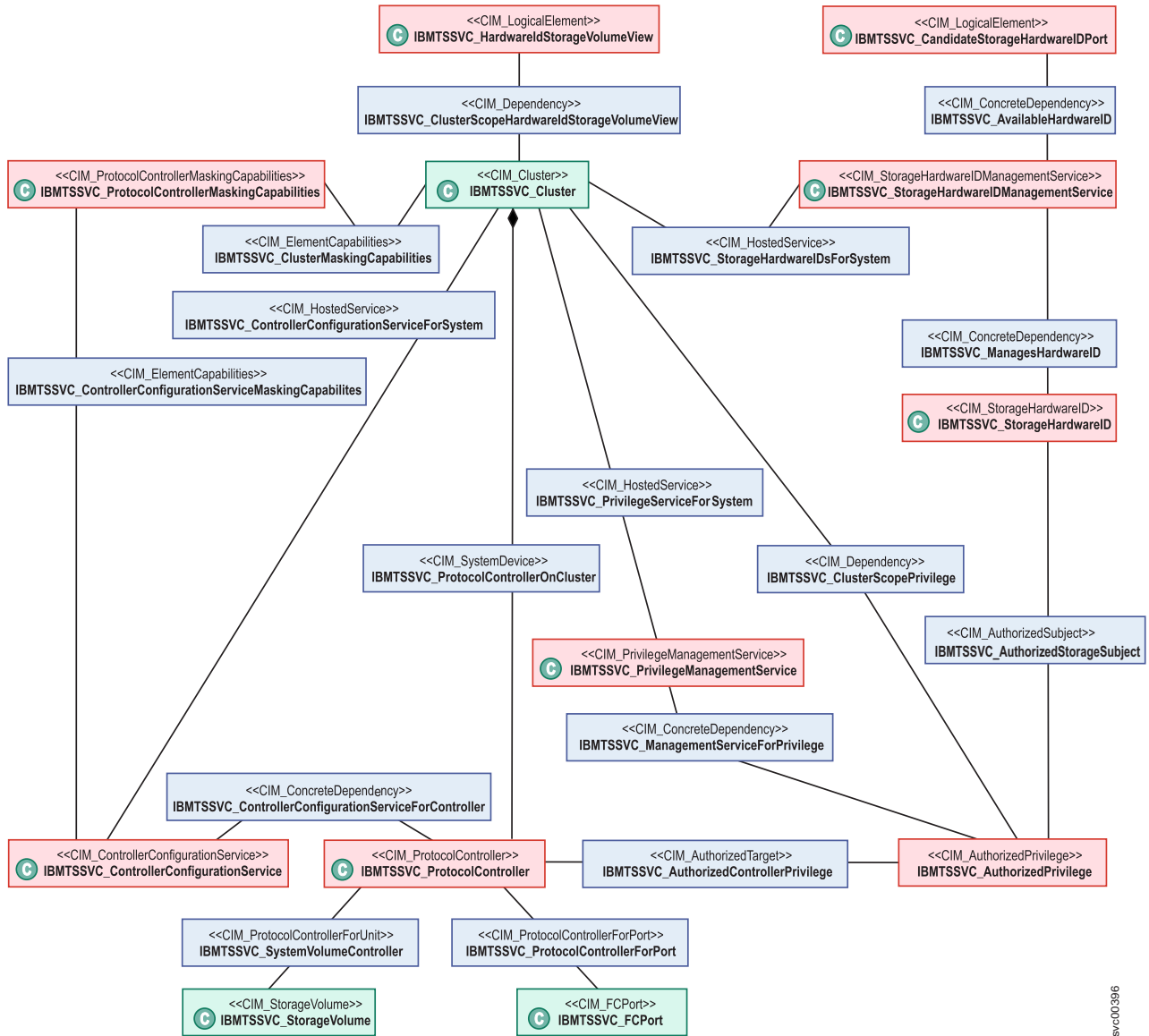


图 21. 屏蔽和映射实例的类图

多计算机系统概要文件

多计算机系统概要文件可使用多个系统呈现虚拟计算机系统。

第 26 页的图 22 显示了多计算机系统概要文件的类和关联。

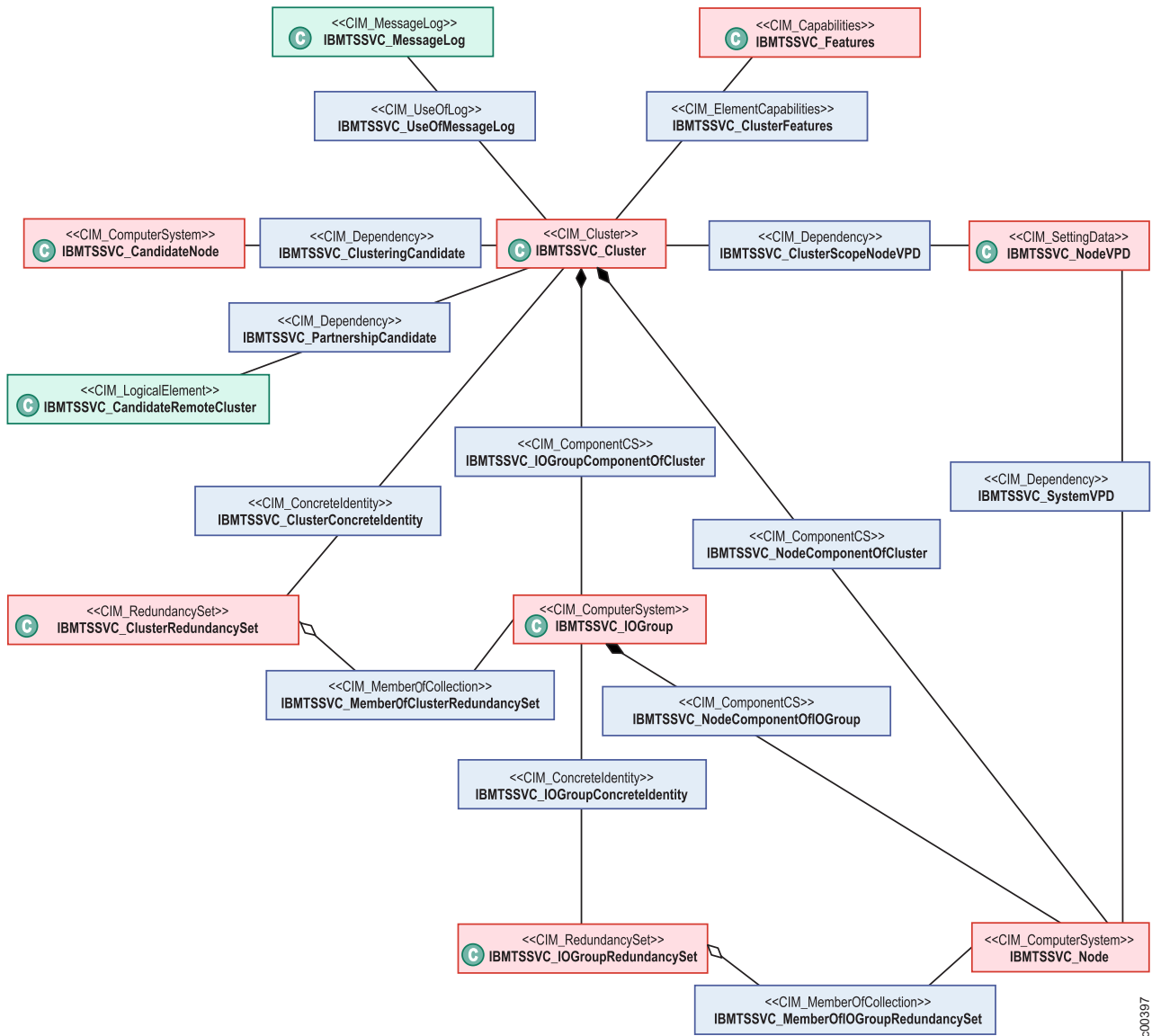
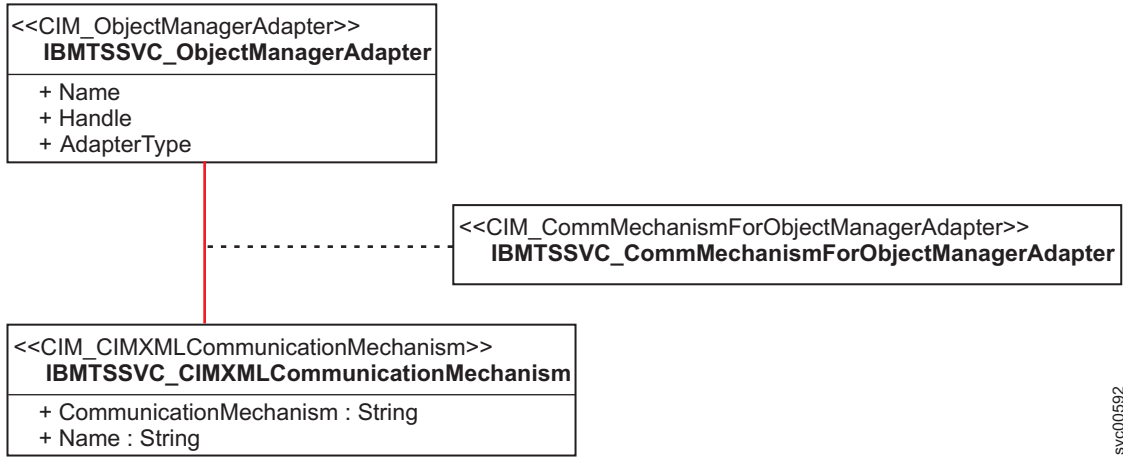


图 22. 多计算机系统实例的类图

对象管理器适配器概要文件

对象管理器适配器概要文件的主要功能是开启或关闭指示服务。

第 27 页的图 23 显示了对对象管理器适配器概要文件的基本元素。



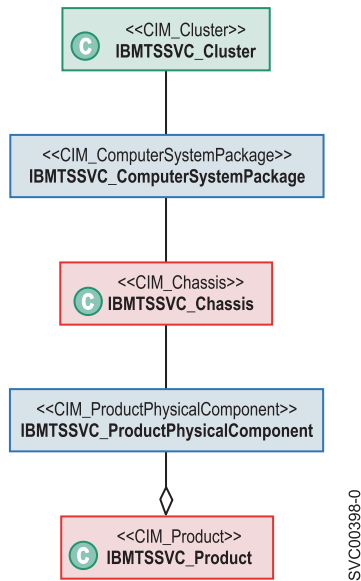
syc00592

图 23. 对象管理器适配器概要文件的统一建模语言元素概述

物理软件包

针对 Storwize V7000 的公共信息模型 (CIM) 代理程序的物理软件包由两个类和两个关联类组成。

图 24 显示了模型的基本类 (构建块)。



SVC00398-0

图 24. CIM 代理程序(针对 Storwize V7000) 的物理软件包的高级概述。

服务器概要文件

针对 Storwize V7000 的公共信息模型 (CIM) 代理程序的服务器概要文件由多个基本类组成。

第 28 页的图 25 显示了模型的基本类 (构建块)。

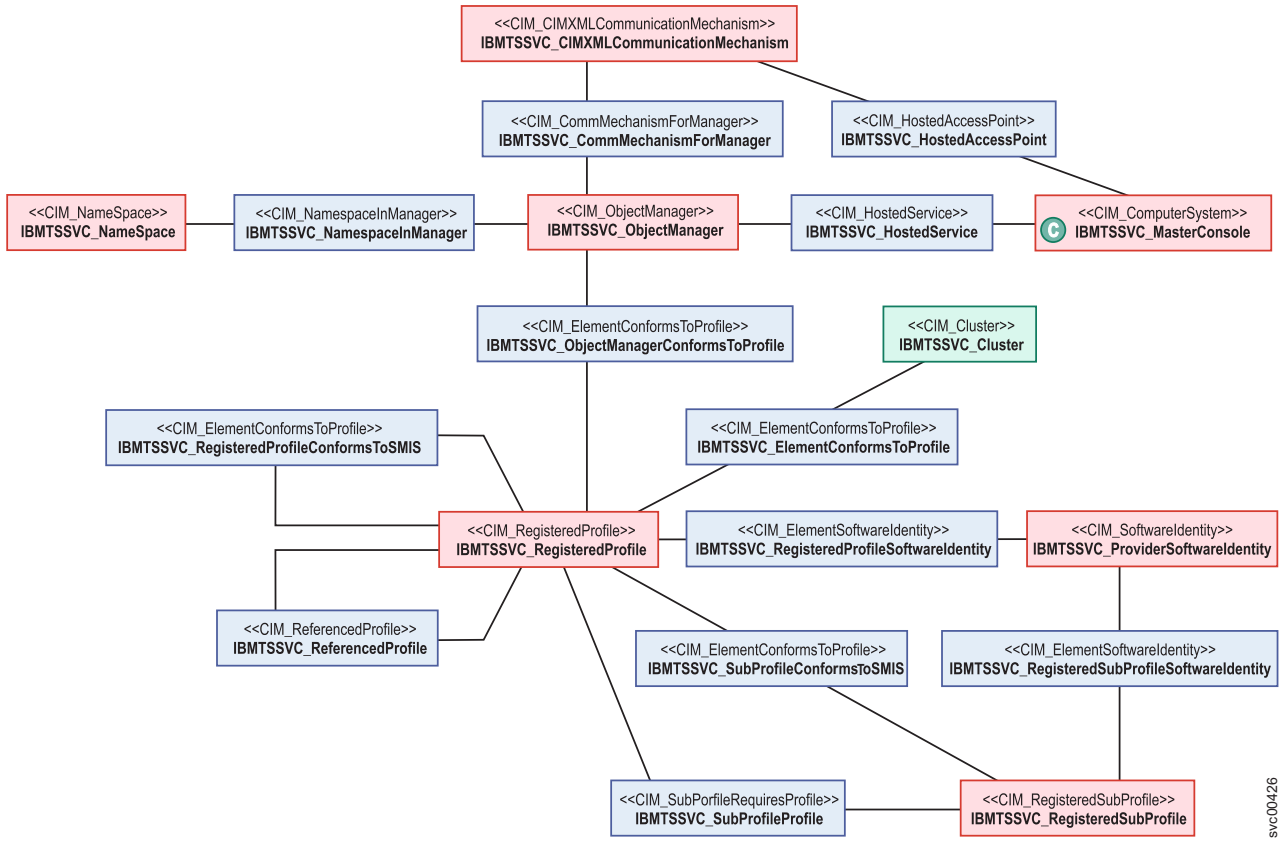


图 25. CIM 代理程序（针对 Storwize V7000）的服务器概要文件的高级概述。

软件概要文件

CIM 代理程序使用软件概要文件对 Storwize V7000 集群系统和 CIM 代理程序的软件进行建模。

图 26 显示软件概要文件的类和关联。

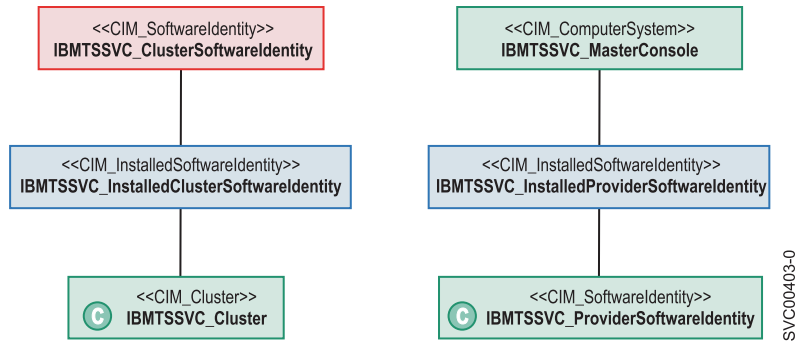


图 26. 软件实例的类图

存储卷镜像

您可以创建两个存储卷镜像拷贝以提高数据可用性。只要其中一个拷贝可用，那么卷将保持联机并可用。

图 27 显示了镜像存储卷中调用的类和关联。

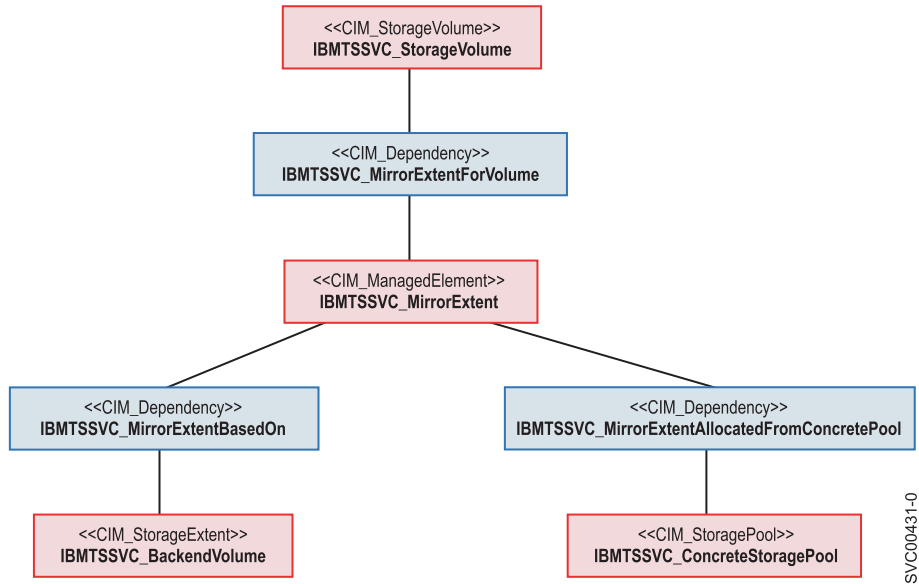


图 27. 镜像存储卷中的类和关联

第 2 章 执行存储器配置任务

存储器配置是后端存储器到存储池的映射和来自这些池的卷的分配。在完成 Storwize V7000 的初始设置之后，可以使用公共信息模型 (CIM) 代理程序对象类实例来完成基本存储器配置任务。

在针对 Storwize V7000 的 CIM 代理程序中，存储器配置会调用三层对象：后端、中间和前端。后端层中的对象包含后端控制器和卷，中间层中的对象包含存储池，前端层中的对象包含向主机呈现的存储卷。

`IBMTSSVC_StorageConfigurationService` 类提供 `CreateOrModifyStoragePool()` 和 `CreateOrModifyElementFromStoragePool()` 方法，用于执行基本存储配置。

您可以使用 `CreateOrModifyStoragePool()` 方法来创建 `IBMTSSVC_StoragePool` 并添加或删除 `IBMTSSVC_BackendVolume`。您可以使用 `CreateOrModifyElementFromStoragePool()` 方法通过 `IBMTSSVC_ConcreteStoragePool` 分配、扩展或缩小 `IBMTSSVC_StorageVolume`。

要完成基本存储器配置，请执行以下任务：

1. 向集群系统添加节点。
2. 向系统添加候选控制机柜。
3. 通过磁盘驱动器创建阵列。
4. 创建存储池。
5. 修改存储池。
6. 创建存储卷。
7. 修改存储卷。

创建新的存储池

`IBMTSSVC_StorageConfigurationService` 类提供用于创建新的 `IBMTSSVC_ConcreteStoragePool` 的方法。

要创建新的 `IBMTSSVC_ConcreteStoragePool` 实例，请遵循以下步骤：

1. 通过以下两种方法中的一种方法，获取与要在其中创建新存储池的 `IBMTSSVC_Cluster` 关联的 `IBMTSSVC_StorageConfigurationService` 实例的引用 (CIMObjectPath):
 - 直接获取 `IBMTSSVC_StorageConfigurationService` 实例的引用。由于 CIMOM 嵌入在集群系统中，因此仅有一个实例通过枚举获得。
 - 通过 `IBMTSSVC_HostedStorageConfigurationService` 关联来获取 `IBMTSSVC_StorageConfigurationService` 实例的引用。
2. 调用 `IBMTSSVC_StorageConfigurationService.CreateOrModifyStoragePool` 方法，同时通过 `IBMTSSVC_BackendVolume` 实例列表指定 `InExtents[]` 参数。

`InExtents[]` 参数是一个字符串数组，包含指向 `IBMTSSVC_BackendVolume` 的 `CIMObjectPath` 的表示。

您也可以只指定大小，CIM 代理程序会执行最佳试探以与其匹配。您可以通过使用 `ElementName` 参数指定池的名称，使用 `BlockSize` 参数指定块的大小。

修改存储池

通过更改池名称和向池中添加或从中删除 `IBMTSSVC_BackendVolume` 实例，可以修改 `IBMTSSVC_ConcreteStoragePool` 实例。

要修改 `IBMTSSVC_ConcreteStoragePool` 实例，请遵循以下步骤：

1. 从 `IBMTSSVC_Cluster` 中选择要修改的 `IBMTSSVC_ConcreteStoragePool` 实例。
2. 识别包含 `IBMTSSVC_ConcreteStoragePool` 设置实例的参数设置的 `IBMTSSVC_StoragePoolSetting` 实例。您可以通过调用 `IBMTSSVC_StorageCapabilities.CreateSetting()` 方法或枚举通过 `IBMTSSVC_StorageSettingsGeneratedFromCapabilities` 与 `IBMTSSVC_StorageCapabilities`（与要修改的 `IBMTSSVC_ConcreteStoragePool` 关联）关联的 `IBMTSSVC_StoragePoolSetting` 来完成该操作。
3. 通过调用 `IBMTSSVC_StorageConfigurationService.CreateOrModifyPool`，使用新的元素名称修改所选 `IBMTSSVC_ConcreteStoragePool` 的名称。
4. 如果需要，可通过向池中添加或从中除去 `IBMTSSVC_BackendVolume` 实例来进一步修改 `IBMTSSVC_ConcreteStoragePool`。

创建新的存储卷

在针对 Storwize V7000 的公共信息模型 (CIM) 代理程序中，`IBMTSSVC_StorageConfigurationService` 类提供用于创建、修改和删除 `IBMTSSVC_StorageVolume` 实例所需的所有方法。

要创建新的 `IBMTSSVC_StorageVolume` 实例，请遵循以下步骤：

在要向其分配新卷的 `IBMTSSVC_Cluster` 上使用 `CreateOrModifyElementFromStoragePool()` 方法，通过以下参数规范来创建新的 `IBMTSSVC_StorageVolume`：

- 虚拟化类型是使用 `VirtualizationType` 参数（0、1 和 2 分别代表条带、顺序或映像）设置的。
- `Format` 标志可用于指定卷是在创建还是扩展时格式化的。
- 您可以指定 `BackendVolumes` 以使用 `BackendVolume REF` 参数来放置卷的扩展数据块。
- `PreferredNode` 参数用于设置卷的首选节点。
- `UnitDeviceID` 参数用于在运行 4.1.0 或更高级别软件的集群系统上设置卷的单元设备标识。
- `ElementName` 参数用于在创建卷时设置卷名称。
- `Autoexpand` 参数

1. 将 `ElementType` 设置为 2。
2. 将 `Size` 设置为首选卷大小（字节）。
3. 获取要从中分配 `IBMTSSVC_StorageVolume` 的 `IBMTSSVC_ConcreteStoragePool` 实例的引用 (`CIMObjectPath`)。
4. 将 `InPool` 设置为要从中分配卷的池的引用（在前面步骤中获取的）。

以下是可用作卷镜像功能或自动精简配置卷功能的一部分的其他参数示例。

Autoexpand

用于尝试保留卷上固定数量的、未使用的实际容量，称为应急容量。该容量最初设置为实际容量，是在创建卷时分配的。如果修改该实际容量，那么会将应急容量重新设置为已用容量和实际容量间的差值。

Copies

指示要创建的镜像拷贝数。指定 1 或 2。

Grainsize

设置自动精简配置卷的颗粒度。仅在设置了 `RealSize` 时，该设置才有效。有效选项包括 32、64、128 或 256。

Import

用于请求 Storwize V7000 从存储池中导入自动精简配置存储卷。

IsSpaceEfficient

用于指示创建的卷为自动精简配置卷的标志。该标志不应用于修改操作。

RealSizeBytes 和 RealSizePercent

指示自动精简配置拷贝的实际大小。这些参数不能一起使用。

WarningSizeBytes

指示将触发警告的已用容量设置点。该参数不能与 `WarningSizeBytes` 参数一起使用。

WarningSizePercent

指示已用容量与触发警告的卷虚拟容量的比率。该参数不能与 `WarningSizeBytes` 参数一起使用。

创建 RAID 阵列

您可以使用本地驱动器创建 RAID 阵列。

要创建 RAID 阵列，请执行以下步骤：

1. 通过以下两种方法中的一种方法，获取与要在其中创建新 RAID 阵列的 `IBMTSSVC_Cluster` 关联的 `IBMTSSVC_StorageConfigurationService` 实例的引用 (`CIMObjectPath`):
 - 直接获取 `IBMTSSVC_StorageConfigurationService` 实例的引用。由于 CIMOM 嵌入在集群系统中，因此仅有一个实例通过枚举获得。
 - 通过 `IBMTSSVC_HostedStorageConfigurationService` 关联来获取 `IBMTSSVC_StorageConfigurationService` 实例的引用。
2. 调用 `IBMTSSVC_StorageConfigurationService CreateOrModifyElementFromElements` 方法。

必须指定以下参数：

- `IBMTSSVC_DiskDriveExtent` 实例列表中的 `InElements []` 参数。 `Extent []` 参数是一个字符串数组，包含指向 `IBMTSSVC_DiskDriveExtent` 的 `CIMObjectPath` 的表示。
- `IBMTSSVC_ConcreteStoragePool` 实例中的 `StoragePool` 参数，用于标识要将创建的 RAID 阵列添加至的 `StoragePool`。
- `IBMTSSVC_RAIDSetting` 实例中的 `Goal` 参数，包含 RAID 级信息。

您也可指定 `StripSize`、`WriteVerify` 和 `SpareGoal`。另请参阅 `BlockServices.mof` 文件，了解更多详细信息。要查看完整的受管对象格式 (MOF) 文档，请从以下 Web 站点中选择文档信息：

Storwize V7000 支持 Web 站点: www.ibm.com/storage/support/storwize/v7000

第 3 章 确定拷贝服务关系

您可以使用公共信息模型代理程序对象类实例来确定新的拷贝服务关系。

拷贝服务

Storwize V7000 提供的拷贝服务包括 FlashCopy、全局镜像和高速镜像。

这些拷贝服务可用于连接到 Storwize V7000 的所有受支持的主机。

FlashCopy 服务能够使您针对目标 IBMTSSVC_StorageVolume 实例创建源 IBMTSSVC_StorageVolume 实例的即时时间点拷贝。

同步拷贝服务（高速镜像）在目标 IBMTSSVC_StorageVolume 上提供一致的源 IBMTSSVC_StorageVolume 拷贝。当数据写入源卷后，会被同步写入到目标卷，源卷和目标卷可属于相同的 IBMTSSVC_Cluster 实例，也可属于不同的 IBMTSSVC_Cluster 实例。

异步拷贝服务（全局镜像）在目标 IBMTSSVC_StorageVolume 上提供源 IBMTSSVC_StorageVolume 的拷贝。当数据写入源卷后，会被异步写入到目标卷，源卷和目标卷可属于相同的 IBMTSSVC_Cluster 实例，也可属于不同的 IBMTSSVC_Cluster 实例。

在存储卷之间创建新的 FlashCopy 关系

IBMTSSVC_StorageConfigurationService 类提供的一些方法可用于确定两个大小相同且属于相同 IBMTSSVC_Cluster 实例的 IBMTSSVC_StorageVolume 实例之间的 FlashCopy 关系。

要在两个 IBMTSSVC_StorageVolume 实例之间创建 FlashCopy 关系，请执行以下步骤：

1. 选择作为首选 FlashCopy 关系的源卷的 IBMTSSVC_StorageVolume 实例。
2. 选择作为目标的有效 IBMTSSVC_StorageVolume。

确保源 IBMTSSVC_StorageVolume 和目标 IBMTSSVC_StorageVolume 实例属于相同的 IBMTSSVC_Cluster 实例。

3. 检索与所选 IBMTSSVC_StorageVolume 实例所属的 IBMTSSVC_Cluster 实例关联的 IBMTSSVC_StorageConfigurationService 实例。
4. 用以下参数规范调用 IBMTSSVC_StorageConfigurationService.AttachReplica() 方法：
 - a. 将 SourceElement 设置为源 IBMTSSVC_StorageVolume 的引用 (CIMObjectPath)。
 - b. 将 TargetElement 设置为目标 IBMTSSVC_StorageVolume 的引用 (CIMObjectPath)。
 - c. 将 ElementName 设置为同步名称（可选）。
 - d. 将 BackgroundCopyRate 设置为后台拷贝速率 (0 - 100) 的首选优先级（可选）。
 - e. 指定 Set 以将新创建的 FlashCopySynchronization 添加到集合中（可选）。如果您指定了 null 值，此新创建的 FlashCopySynchronization 就不会成为同步集合的成员。
 - f. 将 CopyType 设置为 5（可选）。这会将 AutoDelete 设置为 True，该设置会在后台拷贝完成之后自动删除 FlashCopy 映射。
 - g. 如果不希望自动删除 FlashCopy 映射，那么将 CopyType 设置为 4。

源 `IBMTSSVC_StorageVolume` 和目标 `IBMTSSVC_StorageVolume` 实例现已通过 `IBMTSSVC_LocalStorageSynchronized` 关联连接起来。

针对同步集创建 FlashCopy 关系

`IBMTSSVC_StorageConfigurationService` 类提供的一些方法可用于确定两个 `IBMTSSVC_StorageVolume` 实例之间的 FlashCopy 关系，然后将该关系添加到 `IBMTSSVC_FlashCopyStorageSynchronizedSet` 实例。

要在两个 `IBMTSSVC_StorageVolume` 实例之间创建 FlashCopy 关系并将该关系添加到 `IBMTSSVC_FlashCopyStorageSynchronizedSet` 实例，请执行以下步骤：

1. 选择作为首选 FlashCopy 关系的源卷的 `IBMTSSVC_StorageVolume` 实例。
2. 选择作为目标的有效 `IBMTSSVC_StorageVolume`。可使用 `IBMTSSVC_CandidateVolume` 确定有效卷。

确保源 `IBMTSSVC_StorageVolume` 和目标 `IBMTSSVC_StorageVolume` 实例属于相同的 `IBMTSSVC_Cluster` 实例。

3. 检索与所选 `IBMTSSVC_StorageVolume` 实例所属的 `IBMTSSVC_Cluster` 实例关联的 `IBMTSSVC_StorageConfigurationService` 实例。
4. 用以下参数规范调用 `IBMTSSVC_StorageConfigurationService.AttachReplica()` 方法：
 - a. 将 `SourceElement` 设置为源 `IBMTSSVC_StorageVolume` 的引用 (`CIMObjectPath`)。
 - b. 将 `TargetElement` 设置为目标 `IBMTSSVC_StorageVolume` 的引用 (`CIMObjectPath`)。
 - c. 将 `ElementName` 设置为同步名称 (可选)。
 - d. 将 `BackgroundCopyRate` 设置为以百分比 (0 - 100%) 表示的后台拷贝速率的首选优先级 (可选)。
 - e. 指定 `Set` 以将新创建的 `FlashCopySynchronization` 添加到集合中 (可选)。如果您指定了 `null` 值，此新创建的 `FlashCopySynchronization` 就不会成为同步集合的成员。
 - f. 将 `CopyType` 设置为 5 (可选)。这会将 `AutoDelete` 设置为 `True`，该设置会在后台拷贝完成之后自动删除 FlashCopy 映射。
 - g. 如果不希望自动删除 FlashCopy 映射，那么将 `CopyType` 设置为 4。
5. 通过用以下参数规范调用 `IBMTSSVC_StorageConfigurationService.CreateSynchronizedSet()` 方法来创建 `IBMTSSVC_FlashCopyStorageSynchronizedSet` 实例：
 - a. 将 `CopyType` 设置为 4 (闪烁)。
 - b. 将 `ElementName` 设置为最新创建的 `IBMTSSVC_FlashCopyStorageSynchronizedSet` 实例的名称 (可选)。
6. 通过将 `Operation` 参数设置为 0 (添加) 以调用 `IBMTSSVC_StorageConfigurationService.ModifySynchronizedSet()` 方法，从而将 `IBMTSSVC_StorageSynchronized` 实例添加到 `IBMTSSVC_FlashCopySynchronizedSet` 实例。如果添加到集中的 FlashCopy 为 `CopyType` 5，那么该集将变为 `CopyType` 5，添加到该集的其他任何 FlashCopy 映射必须具有 `CopyType` 5，否则添加将失败。同样，如果集包含的 FlashCopy 为 `CopyType` 4，那么该集中所有其他 FlashCopy 映射必须为 `CopyType` 4，否则修改操作将失败。

同步必须与托管服务属于相同的集群系统。

在位于相同集群系统中的卷之间创建同步拷贝关系

`IBMTSSVC_StorageConfigurationService` 类提供的一些方法可用于在源 `IBMTSSVC_StorageVolume` 和目标 `IBMTSSVC_StorageVolume` 之间创建同步拷贝关系。

执行以下步骤来创建同步拷贝关系：

1. 选择作为首选源卷的源卷的 `IBMTSSVC_StorageVolume` 实例。

2. 选择作为目标卷的 IBMTSSVC_StorageVolume 实例。
3. 获取与所选卷所属的 IBMTSSVC_Cluster 实例关联的 IBMTSSVC_StorageConfigurationService 实例的引用 (CIMObjectPath)。
4. 用以下参数规范调用 IBMTSSVC_StorageConfigurationService.AttachReplica() 方法:
 - a. 将 SourceElement 设置为源 IBMTSSVC_StorageVolume 实例的引用 (CIMObjectPath)。
 - b. 将 TargetElement 设置为目标 IBMTSSVC_StorageVolume 的引用 (CIMObjectPath)。
 - c. 将 ElementName 设置为同步名称 (可选)。
 - d. 针对同步, 将 CopyType 设置为 3; 针对异步, 将其设置为 2。

源 IBMTSSVC_StorageVolume 实例和目标 IBMTSSVC_StorageVolume 实例现已通过 RemoteStorageSynchronized 关联连接起来。

在位于不同集群系统中的卷之间创建同步拷贝关系

IBMTSSVC_StorageConfigurationService 类提供的一些方法可用于在属于不同 IBMTSSVC_Cluster 实例的源 IBMTSSVC_StorageVolume 实例和目标 IBMTSSVC_RemoteStorageVolume 实例之间创建同步拷贝关系。

要在源位于本地集群系统而目标位于远程集群系统的两个卷之间创建同步拷贝关系, 请执行以下步骤:

1. 将 IBMTSSVC_Cluster 实例标识为首选同步拷贝关系的源集群系统。
2. 获取与源集群系统关联的 IBMTSSVC_StorageConfigurationService 实例的引用 (CIMObjectPath)。
3. 通过 IBMTSSVC_RemotePartnership 关联标识要放置同步拷贝的 IBMTSSVC_RemoteCluster。
4. 通过用以下参数规范调用 IBMTSSVC_CascadingAllocationService.Allocate() 方法来创建远程集群系统伙伴关系:
 - a. 将 RemoteCluster 设置为 IBMTSSVC_RemoteCluster 实例的引用 (CIMObjectPath)。
 - b. 将 Bandwidth 设置为首选带宽 (MB) (可选)。

确保同时从源和候选集群系统中发布该方法, 以确定完全配置的伙伴关系; 否则, 将无法确定同步拷贝关系。

5. 从源 IBMTSSVC_Cluster 选择一个 IBMTSSVC_StorageVolume 作为源卷。
6. 从 IBMTSSVC_RemoteCluster 选择一个 IBMTSSVC_CandidateVolume 作为目标卷。要确定候选卷, 请使用来自源卷的 IBMTSSVC_CopyCandidate 关联。

注: 远程集群系统上的 IBMTSSVC_StorageVolume 实例在本地集群系统上显示为 IBMTSSVC_CandidateVolume 实例。

7. 用以下参数规范调用 IBMTSSVC_StorageConfigurationService.AttachReplica() 方法:
 - a. 将 SourceElement 设置为源 IBMTSSVC_StorageVolume 实例的引用 (CIMObjectPath)。
 - b. 将 TargetElement 设置为目标 IBMTSSVC_CandidateVolume 的引用 (CIMObjectPath)。
 - c. 将 ElementName 设置为同步名称 (可选)。
 - d. 将 CopyType 设置为 3。

源 IBMTSSVC_StorageVolume 实例和目标 IBMTSSVC_StorageVolume 或 IBMTSSVC_RemoteStorageVolume 实例现已通过 IBMTSSVC_MasterRemoteStorageSynchronized 关联连接起来。

创建和启动 FlashCopy 映射的代码样本

该信息演示 CIMOM 方法如何控制集群系统。代码样本包含来自 Java 类的旨在创建和启动 FlashCopy 映射的主方法和其他从主方法中调用的方法。

在本主题中，术语方法 (*method*) 指 Java 方法。术语方法 (*Method*, 首字母大写) 指 CIM 方法。

Java 主方法

本示例显示用于创建和启动 FlashCopy 映射的 Java 主方法。本示例假设您的 Java 程序每次旨在控制相同的集群系统。使之变得更加灵活是一个相对简单的过程，但具体设计由您来完成。

```
/*
 * FC Mapping states
 */
private static UnsignedInt16 INITIALIZED = new UnsignedInt16(2);
private static UnsignedInt16 PREPARING = new UnsignedInt16(3);
private static UnsignedInt16 PREPARED = new UnsignedInt16(4);

public static void main(String[] args) throws CIMException
{
    /*
     * First step is to connect to the CIMOM
     */
    UserPrincipal user = new UserPrincipal("superuser");
    PasswordCredential pwd = new PasswordCredential("itsol3sj");
    CIMNameSpace ns = new CIMNameSpace("https://9.43.86.115:5989/root/ibm");

    CIMClient client = null;

    client = new CIMClient(ns,user,pwd);

    /*
     * Next, select the clustered system that we are interested in
     */
    CIMInstance chosenCluster = getCluster("ITSOCL1",client);

    /*
     * At this point, the relevant clustered system has been selected
     * and 'chosenCluster' is a CIMInstance of this clustered system
     *
     * Get the Config Service of this clustered system
     */
    CIMObjectPath cService = getConfigService(chosenCluster, client);

    /*
     * Now, get all of the Volumes in this clustered system
     */
    Map<Integer,CIMObjectPath> volumesById = getVolumes(chosenCluster,client);

    /*
     * Select the FlashCopy Source
     *
     * In this case, Volume 10 is our source
     * Volume 11 is our target
     */
    CIMObjectPath fcSrc = volumesById.get(new Integer(10));
    CIMObjectPath fcTgt = volumesById.get(new Integer(11));/*

    /*
     * Now create FC Mapping
     */
    CIMValue rc = makeFlashCopyMapping("CIMOMTestMap", fcSrc, fcTgt, cService,
        client,false);
}
```

```

/*
 * Now that this has been created, we need to get an
 * Object Path to the newly created Association
 */
List<CIMObjectPath> fcMaps = getFCMappings(fcSrc, client);
CIMObjectPath fcMapping = fcMaps.get(0);

/*
 * Now we prepare the FC Mapping
 */
CIMArgument[] outArgs = new CIMArgument[2];
rc = prepareFCMapping(cService, fcMapping, client, outArgs);
System.out.println("Got value:"+
    Integer.toHexString(Integer.parseInt(rc.toString())));

/*
 * Loop until it is prepared
 */
CIMValue fcMapState = new CIMValue(PREPARING);
while(fcMapState.equals(new CIMValue(PREPARING)))
{
    CIMInstance fcMapInfo = client.getInstance(fcMapping);
    fcMapState = fcMapInfo.getProperty("SyncState").getValue();
}

/*
 * Now start the FC Mapping
 */
rc = startFCMapping(cService, fcMapping, client, outArgs);
System.out.println("Got value:"+
    Integer.toHexString(Integer.parseInt(rc.toString())));
}

```

getCluster 方法

getCluster 方法返回与具有所提供名称的集群系统对应的 CIM 实例。它通过枚举类 IBMTSSVC_Cluster 的所有实例并检查每个实例的名称来完成该操作。当找到与所提供名称匹配的实例时，将返回到该实例的对象路径。

```

static private CIMInstance getCluster(String clusterName, CIMClient client) throws
CIMException
{
    CIMInstance chosenCluster = null;
    Enumeration<CIMInstance> clusters =
        client.EnumerateInstances(new CIMObjectPath("/root/ibm:IBMTSSVC_Cluster"));

    while(clusters.hasMoreElements())
    {
        CIMInstance possibleCluster = clusters.nextElement();
        String possibleName =
            possibleCluster.getProperty("ElementName").getValue().toString();

        if(possibleName.equals("\"+clusterName+"\"))
        {
            chosenCluster = possibleCluster;
        }
    }
    return chosenCluster;
}

```

getConfigService 方法

CIM_StorageConfigurationService 类在 SVC 中没有直接对等项，但需要一个该类的实例来针对其调用方法。

在本方法中，将请求与所提供集群系统关联的所有实例。将集群系统与其配置服务连接起来的关联为 CIM_HostedService。由于集群系统只具有与其关联的配置服务，因此选择了枚举中的第一个对象路径。

```
static private CIMObjectPath getConfigService(CIMInstance cluster, CIMClient
client) throws CIMException
{
    Enumeration<CIMObjectPath> configServices = null;
    configServices = client.associatorNames(
        cluster.getObjectPath(),
        "CIM_HostedService",
        "CIM_StorageConfigurationService",
        null,
        null);
    return configServices.nextElement();
}
```

getVolumes 方法

该方法返回将卷标识（整数）与 IBMTSSVC_StorageVolume 对象路径关联的映射。该方法请求与所提供集群系统实例关联的所有 IBMTSSVC_StorageVolume 实例。

```
static private Map<Integer,CIMObjectPath> getVolumes(CIMInstance cluster, CIMClient
client) throws CIMException
{
    Enumeration<CIMObjectPath> volumes = client.associatorNames(
        cluster.getObjectPath(),
        null,
        "IBMTSSVC_StorageVolume",
        null,
        null);

    Map<Integer,CIMObjectPath> volumesById = new HashMap<Integer, CIMObjectPath>();

    while(volumes.hasMoreElements())
    {
        CIMObjectPath volumeOP = volumes.nextElement();
        CIMValue volumesId = volumeOP.getKey("DeviceID").getValue();
        String idAsString = volumeId.toString();
        String idNoQuotes = idAsString.substring(1, idAsString.length()-1);
        volumesById.put(Integer.parseInt(idNoQuotes), volumeOP);
    }
    return volumesById;
}
```

makeFlashCopyMapping 方法

该示例根据集群系统配置服务调用 AttachReplica。CIM 方法使用分类的参数。在本方法中，可以看到使用了 argRef、argString 和 argUint16 方法。这些方法充当生成 CIM 方法所需参数的快捷方式。AttachReplica 方法用于 FlashCopy、高速镜像和全局镜像。CopyType 参数指示所需的类型。

```
static private CIMValue makeFlashCopyMapping(
    String name,
    CIMObjectPath source,
    CIMObjectPath target,
    CIMObjectPath configService,
    CIMClient client,
    boolean autodelete) throws CIMException
{
    CIMArgument src = argRef("SourceElement", source, "IBMTSSVC_StorageVolume");
    CIMArgument tgt = argRef("TargetElement", target, "IBMTSSVC_StorageVolume");
    CIMArgument fcName = argString("ElementName",name);
    CIMArgument type = argUint16("CopyType",autodelete?5:4);
    CIMArgument[] inArgs = {src,tgt,fcName,type};
    CIMArgument[] outArgs = new CIMArgument[1];
```



```

    CIMValue rc = client.invokeMethod(configService,
        "AttachReplica",
        inArgs,
        outArgs);
    return rc;
}

```

getFCMappings 方法

getFCMappings 方法返回与所提供 VDisk 关联的所有 FCMappings 的列表。本方法请求引用所提供 IBMTSSVC_StorageVolume 的所有关联的列表。目前，本类型的所有 Java WBEM 服务方法都返回枚举。该方法会将此枚举转化为列表以易于使用。

```

static private List<CIMObjectPath> getFCMappings(CIMObjectPath volume, CIMClient
client) throws CIMException
{
    Enumeration<CIMObjectPath> assocs = client.referenceNames(
        volume,
        "IBMTSSVC_LocalStorageSynchronized",
        null);
    return Collections.list(assocs);
}

```

prepareFCMapping 方法

prepareFCMapping 方法准备 FlashCopy 映射。与 AttachReplica 方法非常相似，ModifySynchronization 方法用于控制 FlashCopy、高速镜像和全局镜像。operation 参数指示您实际要执行的操作。

```

private static CIMValue prepareFCMapping(
    CIMObjectPath configService,
    CIMObjectPath fcMapping,
    CIMClient client,
    CIMArgument[] outArgs) throws CIMException
{
    CIMArgument operation = argUint16("Operation", 6);
    CIMArgument synch = argRef("Synchronization",
    fcMapping, "IBMTSSVC_FlashCopyStorageSynchronized");

    CIMArgument[] inArgs = new CIMArgument[]{operation, synch};
    outArgs = new CIMArgument[2];

    return client.invokeMethod(configService,
        "ModifySynchronization",
        inArgs,
        outArgs);
}

```

startFCMapping 方法

startFCMapping 方法启动 FlashCopy 映射。本方法与在『prepareFCMapping 方法』中一样调用 ModifySynchronization 方法，但使用不同的 Operation 参数。

```

private static CIMValue startFCMapping(
    CIMObjectPath configService,
    CIMObjectPath fcMapping,
    CIMClient client,
    CIMArgument[] outArgs) throws CIMException
{
    CIMArgument operation = argUint16("Operation", 4);
    CIMArgument synch = argRef("Synchronization",
    fcMapping, "IBMTSSVC_FlashCopyStorageSynchronized");

    CIMArgument[] inArgs = new CIMArgument[]{operation, synch};
    outArgs = new CIMArgument[2];
}

```

```

return client.invokeMethod(configService,
    "ModifySynchronization",
    inArgs,
    outArgs);
}

```

自变量生成器类

该类使用以下自变量生成器:

- **argUInt16** 方法返回一个无符号 16 位整数类型的自变量。

```

static private CIMArgument argUInt16(String name, int arg)
{
    return new CIMArgument(
        name,
        new CIMValue(
            new UnsignedInt16(arg),
            new CIMDataType(CIMDataType.UINT16)
        )
    );
}

```

- **argString** 方法返回一个字符串类型的自变量。

```

static private CIMArgument argString(String name, String str )
{
    return new CIMArgument(
        name,
        new CIMValue(
            str,
            new CIMDataType(CIMDataType.STRING)
        )
    );
}

```

- **argRef** 方法返回一个引用类型的自变量。它是对提供的对象路径指示的实例的引用。

```

static private CIMArgument argRef(
    String name,
    CIMObjectPath path,
    String className )
{
    return new CIMArgument(
        name,
        new CIMValue(
            path,
            new CIMDataType(className)
        )
    );
}

```

第 4 章 CIM 代理程序网络注意事项

您可以手动设置针对远程连接访问的 CIM 代理程序服务或用户界面连接信息。

RemoteServiceAccessPoint 实例

在具有多个网络适配器的环境中，可能需要手动设置 RemoteServiceAccessPoint (RSAP) 实例的连接数据。

IBMTSSVC_RemoteServiceAccessPoint 类托管连接到 Web 用户界面时所需的信息。管理应用程序从 CIMOM 获得 RSAP 的实例，以通过 Web 启动用户界面。

您可以手工设置 RSAP 的连接数据。在使用多块网卡的环境中这将很有帮助。

要设置连接数据，请执行以下步骤：

1. 获得 IBMTSSVC_Cluster 实例。
2. 通过集群系统上的“修改”实例，使用包含已更改属性的属性列表修改 ConsoleIP 和 ConsolePort 属性。

CIM 代理程序会自动更新 RSAP。

第 5 章 维护和诊断任务

执行维护、参考和诊断任务。

可以执行提供有关系统信息的任务。

获取类列表

使用 `IBMTSSVC_ConfigurationReportingService` 获取有关机柜类的信息。

要检索 `IBMTSSVC_Enclosure` 支持或已安装的类列表，请执行以下步骤：

1. 选择作为容器的 `IBMTSSVC_Enclosure` 实例。
2. 检索 `IBMTSSVC_ConfigurationReportingService` 实例。
3. 采用以下参数规范调用 `IBMTSSVC_ConfigurationReportingService.GetClassTypes()` 方法：
 - a. 将 `InquiryType` 设置为 2 或 3。
 - b. 将 `CIM_ManagedElement` 设置为目标 `IBMTSSVC_Enclosure` 的引用 (`CIMObjectPath`)。

返回的 `ClassTypes[]` 包含 `IBMTSSVC_Enclosure` 支持或已安装的类的类名。 `IBMTSSVC_Enclosure` 当前支持类 `IBMTSSVC_DiskDrive` 类型。

获取 UnitTypes 列表

使用 `IBMTSSVC_ConfigurationReportingService` 来获取有关机柜 `UnitTypes` 的信息。

要检索 `IBMTSSVC_Enclosure` 支持或已安装的单元类型的列表，请执行以下步骤：

1. 选择作为容器的 `IBMTSSVC_Enclosure` 实例。
2. 直接获取 `IBMTSSVC_ConfigurationReportingService` 实例。
3. 采用以下参数规范调用 `IBMTSSVC_ConfigurationReportingService.GetClassTypes()` 方法：
 - a. 将 `InquiryType` 设置为 2 或 3。
 - b. 将 `ClassType` 设置为在 `GetClassTypes.ClassTypes` 属性中返回的一个条目。
 - c. 将 `CIM_ManagedElement` 设置为目标 `IBMTSSVC_Enclosure` 的引用 (`CIMObjectPath`)。

返回的 `UnitTypes[]` 包含在 `IBMTSSVC_Enclosure` 中受支持或已安装的 `ClassTypes` 的单元类型。

获取当前已安装单元的数目

使用 `IBMTSSVC_ConfigurationReportingService` 来获取有关给定 `UnitType`（给定 `ClassType`）的最大受支持单元数或当前已安装单元数的信息。

要检索有关给定 `UnitType`（给定 `ClassType`）的最大受支持单元数或当前已安装单元数的信息，请执行以下步骤：

1. 选择作为容器的 `IBMTSSVC_Enclosure` 实例。
2. 检索与所选 `IBMTSSVC_Enclosure` 实例所属的 `IBMTSSVC_Cluster` 实例关联的 `IBMTSSVC_ConfigurationReportingService` 实例。

3. 采用以下参数规范调用 `IBMTSSVC_ConfigurationReportingService.GetClassTypes()` 方法:
 - a. 如果要返回受支持的数目, 那么将 `InquiryType` 设置为 2; 如果要返回当前已安装的数目, 那么将其设置为 3。
 - b. 将 `ClassType` 设置为在 `GetClassTypes.ClassTypes` 属性中返回的一个条目。
 - c. 将 `CIM_ManagedElement` 设置为目标 `IBMTSSVC_Enclosure` 的引用 (`CIMObjectPath`)。

返回的 `NumberOfUnits` 是 `IBMTSSVC_Enclosure` 中支持或已安装的 `ClassTypes` 的最大受支持单元数或当前已安装单元数。

使用指示服务

您可以开启或关闭指示服务。

`IBMTSSVC_ObjectManagerAdapter` 类提供用于访问指示服务的方法。要开启或关闭指示服务, 请执行以下步骤:

1. 检索与 `IBMTSSVC_Cluster` 实例关联的 `IBMTSSVC_ObjectManagerAdapter` 实例。
2. 调用 `IBMTSSVC_ObjectManagerAdapter.StartService()` 以启动指示服务, 调用 `IBMTSSVC_ObjectManagerAdapter.StopService()` 方法以停止该指示服务。

获取性能统计信息

`IBMTSSVC_PerformanceStatisticsService` 类提供的一些方法可用于获取 `IBMTSSVC_Node`、`IBMTSSVC_DiskDrive`、`IBMTSSVC_StorageVolume` 和 `IBMTSSVC_BackendVolume` 的性能统计信息。

要获取性能统计信息实例, 请执行以下步骤:

1. 获取与 `IBMTSSVC_Cluster` 关联的 `IBMTSSVC_PerformanceStatisticsService` 实例的引用 (`CIMObjectPath`), 您将通过 `IBMTSSVC_HostedPerformanceStatisticsService` 关联在该 `IBMTSSVC_Cluster` 中获取性能统计信息。
2. 调用 `IBMTSSVC_PerformanceStatisticsService.GetStatisticsCollection` 方法, 以获取性能统计信息。使用以下参数规范:
 - `ElementTypes` 参数用于设置将根据首选源返回的首选统计信息:
 - 4 返回 `IBMTSSVC_StorageVolume` 的性能统计信息
 - 8 返回 `IBMTSSVC_DiskDrive` 的性能统计信息
 - 32768 返回 `IBMTSSVC_BackendVolume` 的性能统计信息
 - 32769 返回 `IBMTSSVC_Node` 的性能统计信息
 - 将 `ManifestCollection` 设置为对 `CIM_BlockStatisticsManifestCollection` 的引用 (`CIMObjectPath`)。
 - 将 `StatisticsFormat` 设置为 2 (将返回一个以逗号分隔的值 [CSV])。

`GetStatisticsCollection` 方法返回的统计信息属性的顺序与受管对象格式 (MOF) 文件中属性定义的顺序相同。

收集 CIM 日志文件

您可以为针对 IBM Support Center 的诊断报告生成 CIM 日志文件。

CIMOM 日志是从管理 GUI 中收集的。

要收集所有集群系统日志和 CIMOM 日志, 请完成以下步骤:

1. 启动管理 GUI。

2. 单击故障诊断 > 支持 > 下载支持包。
3. 选择所需的日志类型，然后单击下载。

注：在 CIMOM 上设置最高级别的日志记录会严重降低 CIMOM 的性能。

第 6 章 CIM 代理程序类、方法和支持的 SMI-S 概要文件

CIM 代理程序类是公共信息模型 (CIM) 代理程序的构建块，使用诸如存储器配置、拷贝服务和逻辑单元号 (LUN) 屏蔽之类的功能。存储管理计划规范 (SMI-S) 基于多种现有的技术，包括 CIM。

要查看这些类和方法的完整受管对象格式 (MOF) 文档，请从以下 Web 站点中选择文档信息：

Storwize V7000 支持 Web 站点：www.ibm.com/storage/support/storwize/v7000

表 4 指出了针对 Storwize V7000 的 CIM 代理程序支持的 SMI-S 概要文件。

表 4. CIM 支持的 SMI-S 概要文件和子概要文件

概要文件	子概要文件
<p>SNIA:Storage Virtualizer</p> <p>存储虚拟化器以类似 RAID 阵列的方式运作，但可以使用位于存储虚拟化器和本地磁盘外部的系统提供的存储器。存储虚拟化器系统将远程和本地存储器组合起来以创建无缝的池。虚拟化系统为主机系统分配池中的卷以供使用。</p> <p>基本的虚拟化器系统概要文件提供系统的只读视图。各种子概要文件对该描述进行扩展，同时还启用配置。</p>	SNIA:Storage Virtualizer:Access Points
	SNIA:Storage Virtualizer:Block Server Performance
	SNIA:Storage Virtualizer:Block Services
	SNIA:Storage Virtualizer:Cascading
	SNIA:Storage Virtualizer:Copy Services
	SNIA:Storage Virtualizer:Disk Drive Lite
	SNIA:Storage Virtualizer:Disk Sparing
	SNIA:Storage Virtualizer:Extent Composition
	SNIA:Storage Virtualizer:FC Initiator Ports
	SNIA:Storage Virtualizer:FC Target Ports
	SNIA:Storage Virtualizer:Health
	SNIA:Storage Virtualizer:iSCSI
	SNIA:Storage Virtualizer:Location
	SNIA:Storage Virtualizer:Masking and Mapping
	SNIA:Storage Virtualizer:Multiple Computer System
SNIA:Storage Virtualizer:Physical Package	
SNIA:Storage Virtualizer:RAID	
SNIA:Storage Virtualizer:Software	
<p>SNIA:Server</p> <p>所有兼容的 SMI-S 服务器都需要服务器概要文件。模型的对象管理器部分根据其支持的通信机制定义 CIM 对象管理器的功能。</p>	SNIA:Server:Profile Registration
	SNIA:Server:Indication Object Manager Adapter

第 7 章 返回码

公共信息模型 (CIM) 返回码提供了有关 CIM 代理程序操作状态的信息。

公共信息模型

表 5 是可能的 CIM 返回码的简明列表。

表 5. 公共信息模型返回码

CIM 返回码	描述	方法	解释
0	成功	GetFreeExtends() ListConfiguration Backups()	参数有效。方法已成功完成。
		AddNode()	已成功添加节点。
		CheckValidity()	已成功获取有关证书的信息。
		DeleteAccount()	已成功删除帐户。
		GenerateCIMOM Certificate() EnableAuto Generation() DisableAuto Generation()	已成功删除证书。
		CreateCode()	已成功创建新帐户。
		SetDefault Validity()	已成功设置有效性。
		GrantGlobal Access() GrantSystem Access()	已成功更改角色。
		CreateGatewayID() AddHardwareIDs ToCollection()	已成功创建集合。
		DeleteStorage HardwareID()	已成功删除 StorageHardwareID。
		CreateStorage HardwareID()	已成功创建 StorageHardwareID。

表 5. 公共信息模型返回码 (续)

CIM 返回码	描述	方法	解释
0	成功 (续)	AttachDevice()	已成功连接卷。
		DeleteProtocolController()	已成功删除控制器。
		CreateProtocolControllerWithPorts()	已成功创建克隆。
		DeleteRemoteClusterPartnership()	已成功删除集群系统关系。
		CreateRemoteClusterPartnership()	已成功确定集群系统关系。
		DeleteHardwareIDCollection()	已成功创建集合。
		DeleteCertificate()	已成功删除证书。
		DeleteSynchronizedSet()	已成功删除 SynchronizedSet。
		ModifySynchronisation()	方法已成功运行。
		EvictNode()	已成功除去节点。

表 5. 公共信息模型返回码 (续)

CIM 返回码	描述	方法	解释
0	成功 (续)	RestartService()	重新引导 CIMOM。
		Shutdown()	已成功启动节点/集群系统的关机操作。
		SetLocale()	已设置语言环境。
		SetTimezone()	已成功设置集群系统的时区。
		SetPasswords() ModifyReset Password ChangeFeature()	已更改密码。
		GetResetPassword ChangeFeature Status()	已成功检索功能部件状态。
		StartStatistics Collection()	已启动统计信息收集。
		DetachDevice()	已成功分离卷。
		StopStatistics Collection()	已停止统计信息收集。
		Backup Configuration()	已成功创建备份。
		Reload Configuration()	已重新装入配置。
		Restore Configuration() Delete Configuration Backup()	已成功恢复。
		AttachReplica()	已成功确定拷贝关系。
		CreateSynchronized Set()	已成功创建 SynchronizedSet。
		SetPassword()	已成功更改密码。

表 5. 公共信息模型返回码 (续)

CIM 返回码	描述	方法	解释
0	作业已完成且无错误	CreateOrModifyStoragePool()	已成功创建池。
		CreateOrModifyElementFromStoragePool()	已成功创建卷。已成功修改池。
		DeleteStoragePool()	已成功删除池。
		ReturnToStoragePool()	已成功删除卷。
		RequestDiscovery()	BackendVolume 发现已成功。
		SetIOGroup()	更改已成功。
		SetQuorum()	方法已成功。
		IncludeBackendVolume()	已成功包含卷。
		ModifySynchronizedSet()	已成功运行 CLI 命令。
0	作业已成功启动	MigrateVolume() MigrateVolumeToImageMode()	已启动迁移作业。
1	不支持	SetLocales()	集群系统不支持语言环境。
		SetPasswords()	集群系统不支持密码更改 (CISCO)。
		ModifyResetPasswordChangeFeature() GetResetPasswordChangeFeatureStatus()	集群系统不支持重置密码更改功能。
		Upgrade()	在 2145 集群系统配置服务上调用该方法。

表 5. 公共信息模型返回码 (续)

CIM 返回码	描述	方法	解释
2	失败	Dump() Clean() Enter() Exit() Clean() GetDump() ClearLog() GetRecord() GetAllRecords() FixRecord() UnfixRecord() ModifyError Settings() Create2062 Cluster()	出现意外的错误。CLI 命令失败。
		GetDump()	处理命令期间失败。找不到该文件。
		Reload Configuration()	无法重新装入配置。
		CreateCode()	无法创建帐户。
		GrantGlobal Access() GrantSystem Access()	无法更改角色。
		SetPassword()	无法更改密码。
		DeleteAccount()	无法删除帐户。
2	未知错误	GenerateCIMOM Certificate() DeleteCertificate()	由于提供程序内部原因，无法生成新的证书。
		EnableAuto Generation() DisableAuto Generation()	由于提供程序内部原因而失败。
		SetDefault Validity()	无法设置有效性。
		CheckValidity()	无法获取有关证书的信息。
4	失败	GenerateCIMOM Certificate() DeleteCertificate()	发生了意外错误。由于信任库问题，无法生成新的证书。

表 5. 公共信息模型返回码 (续)

CIM 返回码	描述	方法	解释
5	错误的参数集	Delete Configuration Backup()	传递的参数数目或类型错误。找不到给定的备份。
		Modify Synchronisation()	传递的参数数目或类型错误，或者其他参数检查失败。
		CreateCode() GrantGlobal Access() SetPassword() GrantSystem Access() DeleteAccount() DeleteCertificate() SetDefault Validity() CheckValidity()	其中一个参数无效。
5	参数无效	Dump(), GetDump() PositionToFirst RecordRoot() GetRecord() FixRecord() UnfixRecord()	其中一个必需参数缺失。
		ModifyIP Address() Create2062 Cluster() Add2062Cluster() Add2145Cluster() Reset2062Node() Reload2062Node()	其中一个必需参数缺失或无效。
		CreateOrModify StoragePool() CreateOrModify ElementFrom StoragePool() Delete StoragePool() ReturnTo StoragePool()	至少其中一个参数无效。
		CreateGatewayID() GenerateCIMOM Certificate()	其中一个参数无效。
		PositionAtRecord()	其中一个必需参数缺失，或不支持记录数为负。

表 5. 公共信息模型返回码 (续)

CIM 返回码	描述	方法	解释
6	不支持 CopyType	ModifySynchronizedSet()	传入的拷贝类型不同于 3 或 4。
6	不支持操作	ModifySynchronizedSet()	提交的操作码对于同步集的拷贝类型无效。
6	SynchronizedSet 不为空	DeleteSynchronizedSet()	集中仍有 StorageSynchronized 关联。必须先除去所有 StorageSynchronized 关联才可运行删除集操作, 或者必须设置 Force 标志。
6	用户标识已存在	CreateCode()	提交的用户标识已存在于另一个帐户中。
6	正在使用	GenerateCIMOMCertificate()	无法生成新的证书。现有证书仍有效且正在使用。
7	StorageSynchronized 不在集中		集中不存在同步的存储器。
8	StorageSynchronized 已在集中		集中已存在同步的存储器, 无法进行添加。
9	StorageSynchronized 与集不兼容		同步的存储器与集不兼容。例如, 使 FlashCopy 与同步拷贝集同步。
0x1000	参数已检查 - 作业已启动		已运行 CLI 拷贝命令, 并已返回作业对象。
0x1000	未删除与其他 ProtocolControllers 关联的 LogicalDevices	DeleteProtocolController()	所连接的存储卷至少有一个与其他控制器连接, 因此无法删除。
0x1000	LogicalDevice 实例无效	AttachDevice()	该设备不是控制器 RedundancyGroup 的卷。
0x1000	LogicalDevice 与控制器无关联	DetachDevice()	设备与该控制器无 ProtocolControllerForUnit 关联。
0x1000	标识已创建	CreateStorageHardwareID()	已将 WWPN 分配给现有存储器硬件标识。
0x1000	找不到指定的实例	DeleteStorageHardwareID()	找不到存储器硬件标识。
0x1000	HardwareID 实例无效	CreateGatewayID() AddHardwareIDsToCollection()	找不到存储器硬件标识, 或者该标识已成为另一个集合的成员。
0x1001	不支持大小	CreateOrModifyElementFromStoragePool()	最初的池不支持请求的大小。大小参数包含的最接近的支持大小比所请求大小更大。请求的大小不是 512 的倍数。返回可以请求的最接近的支持大小。
0x1001	设备号冲突	AttachDevice()	指定的设备号已被占用。

表 5. 公共信息模型返回码 (续)

CIM 返回码	描述	方法	解释
0x1001	硬件实施不支持指定的 IDType	CreateStorageHardwareID()	标识类型不同于 2。
0x8000	ComputerSystem 无效	AddNode()	提交的 ComputerSystem 不是 IBMTSSVC_CandidateNode。
		EvictNode()	提交的 ComputerSystem 不是 IBMTSSVC_Node。
		Shutdown()	提交的 ComputerSystem 不是 IBMTSSVC_Cluster 的 IBMTSSVC_Node。
0x8000	语言环境无效	SetLocale()	提交的语言环境大于 9。
0x8000	类型无效	Dump()	传入的类型大于 2。
0x8000	连接已被拒绝	GetDump()	与集群系统的连接已丢失或无法连接至节点（仅限 CISCO）。
0x8000	找不到备份	RestoreConfiguration()	找不到指定的备份。
0x8000	删除已失败	DeleteConfigurationBackup()	删除备份目录失败。失败可能由共享违例所引起。
0x8000	IOGroup 必须已聚集节点	SetIOGroup()	I/O 组无任何节点。
0x8000	标识无效	SetQuorum()	定额标识为大于 2 的数字。
0x8000	卷无效	IncludeBackendVolume()	未除去卷。
0x8000	不支持 CopyType	AttachReplica()	传入的拷贝类型不同于 2 或 3。
		CreateReplica()	传入的拷贝类型不同于 3 或 4。
0x8000	端口来自多个 IO 组	CreateProtocolControllerWithPorts()	要求所有端口都属于相同的 I/O 组。
0x8000	HardwareID 仍与 AuthorizationSubject 绑定。必需 Force 标志	DeleteStorageHardwareID()	硬件标识授予存储卷访问权，未指定 Force 标志。
0x8000	主机属于 LUN 映射	DeleteHardwareIDCollection()	要删除该主机，请使用该主机以针对每个特权和该主机所连接的控制器运行 RemoveAccess 方法，或者将“Force”设置为“True”。
0x8000	找不到记录	GetRecord() GetAllRecords()	找不到任何记录。
0x8000	无法连接至集群系统	Create2062Cluster() Add2062Cluster()	无法连接至集群系统。
0x8000	已拒绝连接至集群系统	Add2145Cluster()	已拒绝连接至集群系统。

表 5. 公共信息模型返回码 (续)

CIM 返回码	描述	方法	解释
0x8000	已拒绝连接至交换机	Reset2062Node() Reload2062Node()	已拒绝连接至交换机。
0x8000	找不到管理 IP	RemoveCluster()	找不到集群系统的 IP。
0x8001	已超过集群系统的最大节点数	AddNode()	已向所有 I/O 组分配了两个节点。
0x8001	前缀无效	Dump()	同时传入文件前缀和功能日志类型。
0x8001	找不到文件	GetDump()	找不到给定的文件路径 (CISCO)。
0x8001	备份脚本失败	Backup Configuration()	备份脚本返回了一个错误。
0x8001	恢复脚本失败	Restore Configuration()	备份脚本返回了一个错误。
0x8001	当前状态不支持操作	Modify Configuration()	在同步存储器的当前状态下不支持提交的操作。例如，不能对处于“已同步”状态的同步存储器执行“准备”操作。
0x8001	当前同步状态不支持操作	Modify Synchronized Set()	集的当前同步状态不支持该操作。
0x8001	不支持的协议	CreateProtocol ControllerWith Ports()	协议 != 2。
0x8001	ClusterName 中的语法错误	Create2062 Cluster() Add2062Cluster() Reset2062Node() Reload2062Node()	由于语法错误，集群系统名称无效。
0x8002	ExtraCapacitySet 无效	AddNode()	提交的 ExtraCapacitySet 不是 IBMTSSVC_ IOGroupSet。
0x8002	安全拷贝失败	Backup Configuration()	使用安全拷贝下载备份文件失败。
0x8002	安全拷贝失败	Upload Configuration()	使用安全拷贝上载备份文件失败。
		CreateStorage HardwareID()	元素的名称和设置要求为 NULL。
0x8002	节点中的语法错误或节点无效	Create2062 Cluster() Add2062Cluster() Reset2062Node() Reload2062 Node() Node()	节点包含语法错误，或者指定的节点无效。
0x8003	已超过 IO 组的最大节点数	AddNode()	已向提交的 I/O 组分配了两个节点。

表 5. 公共信息模型返回码 (续)

CIM 返回码	描述	方法	解释
0x8003	备份目录创建失败	Backup Configuration()	无法创建备份目录。
0x8003	清除命令失败	Upload Configuration()	无法清除集群系统 /tmp/ 目录。
0x8003	用户名或密码无效 (仅限 ResetNode)	Add2062Cluster() Reset2062Node() Reload2062Node()	用户名或密码无效。
0x8004	删除/重命名旧的备份文件失败	不适用	无法重命名或删除备份目录。
0x8004	SwitchIP 错误/无法连接至交换机	Create2062 Cluster() Add2062Cluster()	交换机的 IP 不正确, 因此无法连接至交换机。
0x8004	未配置 SwitchIP	Reset2062Node() Reload2062Node()	未配置交换机的 IP。
0x8005	集群系统的 IP 中的语法错误	不适用	集群系统的 IP 包含语法错误。
0x8006	插槽无效	不适用	插槽无效。
0x8007	无法将公用密钥上载至交换机	不适用	无法将公用密钥上载至交换机。
0x8100	集群系统范围违例	不适用	一个或多个参数超出集群系统范围。
0x8200	不适用	不适用	已成功运行方法, 但忽略了一个或多个参数。

附录. 辅助功能选项

辅助功能选项可以帮助身有残疾（如行动不便或视力障碍）的用户顺利地使用软件产品。

功能部件

该列表包含管理 GUI 中的主要辅助功能:

- 可使用屏幕阅读器软件和数字语音合成器听取屏幕上显示的内容。以下屏幕阅读器已经过测试: JAWS 11。
- 使用键盘可访问大部分 GUI 功能。对于不可访问的功能, 可通过命令行界面 (CLI) 来使用同类功能。

通过键盘进行浏览

您可以使用按键或组合键来执行也可通过鼠标操作完成的操作和许多菜单操作。您可以在键盘上使用以下组合键来浏览管理 GUI 和帮助系统:

- 要在不同的 GUI 面板间浏览, 请选择 GUI 登录面板上的低分辨率图形方式选项。您可使用该选项来浏览至所有面板, 而无需手动输入 Web 地址。
- 要转至下一个框架, 请按 **Ctrl+Tab** 键。
- 要移至上一个框架, 请按 **Shift+Ctrl+Tab** 键。
- 要浏览至面板中的下一个链接、按钮或主题, 请在框架 (页面) 内按 **Tab** 键。
- 要移动到面板中的上一个链接、按钮或主题, 请按 **Shift+Tab** 键。
- 要选择 GUI 对象, 请按 **Enter** 键。
- 要打印当前页面或活动框架, 请按 **Ctrl+P** 键。
- 要展开树节点, 请按右方向键。要折叠树节点, 请按左方向键。
- 要滚动到页面顶部, 请按 **Home** 键; 要滚动到页面底部, 请按 **End** 键。
- 要后退, 请按 **Alt+左方向键**。
- 要前进, 请按 **Alt+右方向键**。
- 对于操作菜单:
 - 按 **Tab** 键浏览至网格标题。
 - 按左方向键或右方向键进入下拉字段。
 - 按 **Enter** 键打开下拉菜单。
 - 按向上方向键或向下方向键选择菜单项。
 - 按 **Enter** 键启动操作。
- 对于过滤器窗格:
 - 按 **Tab** 键浏览至过滤器窗格。
 - 按向上方向键或向下方向键更改过滤器或导航未选择内容。
 - 按 **Tab** 键浏览至过滤器窗格中的放大镜图标, 并按 **Enter** 键。
 - 输入过滤文本。
 - 按 **Tab** 键浏览至红色 X 图标, 并按 **Enter** 键以重置过滤器。
- 对于信息区域:
 - 按 **Tab** 键浏览至信息区域。

- 按 Tab 键浏览至可编辑的字段。
- 输入您的编辑内容，并按 Enter 键以发出更改命令。

访问出版物

您可在以下 Web 站点找到 IBM Storwize V7000 信息的 HTML 版本:

publib.boulder.ibm.com/infocenter/storwize/ic/index.jsp

您可以通过使用屏幕阅读器软件访问该信息，并使用数字语音合成器听取屏幕上显示的内容。已使用以下屏幕阅读器测试了此信息: JAWS V10 或更高版本。

声明

本信息是为在美国提供的产品和服务编写的。

在其他国家或地区，IBM 可能不提供本文档中所讨论的产品、服务或功能。有关您当前所在区域的产品和服务的信息，请向您当地的 IBM 代表咨询。任何对 IBM 产品、程序或服务的引用并非意在明示或暗示只能使用 IBM 的产品、程序或服务。只要不侵犯 IBM 的知识产权，任何同等功能的产品、程序或服务，都可以代替 IBM 产品、程序或服务。但是，评估和验证任何非 IBM 产品、程序或服务，则由用户自行负责。

IBM 公司可能已拥有或正在申请与本文档内容有关的各项专利。提供本文档并未授予用户使用这些专利的任何许可。您可以用书面方式将许可查询寄往：

*IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.*

有关双字节 (DBCS) 信息的许可查询，请与您所在国家或地区的 IBM 知识产权部门联系，或用书面方式将查询寄往：

*Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan*

本条款不适用英国或任何这样的条款与当地法律不一致的国家或地区：INTERNATIONAL BUSINESS MACHINES CORPORATION“按现状”提供本出版物，不附有任何种类的（无论是明示的还是暗含的）保证，包括但不限于暗含的有关非侵权、适销和适用于某种特定用途的保证。某些国家或地区在某些交易中不允许免除明示或暗含的保证。因此本条款可能不适用于您。

本信息中可能包含技术方面不够准确的地方或印刷错误。此处的信息将定期更改；这些更改将编入本资料的新版本中。IBM 可以随时对本资料中描述的产品和/或程序进行改进和/或更改，而不另行通知。

本信息中对非 IBM Web 站点的任何引用都只是为了方便起见才提供的，不以任何方式充当对那些 Web 站点的保证。那些 Web 站点中的资料不是 IBM 产品资料的一部分，使用那些 Web 站点带来的风险将由您自行承担。

IBM 可以按它认为适当的任何方式使用或分发您所提供的任何信息而无须对您承担任何责任。

本程序的被许可方如果要了解有关程序的信息以达到如下目的：(i) 允许在独立创建的程序和其他程序（包括本程序）之间进行信息交换，以及 (ii) 允许对已经交换的信息进行相互使用，请与下列地址联系：

*IBM Corporation
Almaden Research
650 Harry Road*

Bldg 80, D3-304, Department 277
San Jose, CA 95120-6099
U.S.A.

只要遵守适当的条件和条款，包括某些情形下的一定数量的付费，都可获得这方面的信息。

本资料中描述的许可程序及其所有可用的许可资料均由 IBM 依据 IBM 客户协议、IBM 国际软件许可协议或任何同等协议中的条款提供。

此处包含的任何性能数据都是在受控环境中测得的。因此，在其他操作环境中获得的数据可能会有明显的不同。有些测量可能是在开发级的系统上进行的，因此不保证与一般可用系统上进行的测量结果相同。此外，有些测量是通过推算而估计的，实际结果可能会有差异。本文档的用户应当验证其特定环境的适用数据。

涉及非 IBM 产品的信息可从这些产品的供应商、其出版说明或其他可公开获得的资料中获取。IBM 没有对这些产品进行测试，也无法确认其性能的精确性、兼容性或任何其他关于非 IBM 产品的声明。有关非 IBM 产品性能的问题可以向这些产品的供应商提出。

所有关于 IBM 未来方向或意向的声明都可随时更改或收回，而不另行通知，它们仅仅表示了目标和意愿而已。

本信息仅用于规划的目的。在所描述的产品上市之前，此处的信息会有更改。

本信息包含在日常业务操作中使用的数据和报告的示例。为了尽可能完整地说明这些示例，示例中包含了个人、公司、品牌和产品的名称。所有这些名称纯属虚构，如与实际商业企业使用的名称及地址雷同，纯属巧合。

版权许可：

本信息包括源语言形式的样本应用程序，这些样本说明不同操作平台上的编程方法。如果是为按照在编写样本程序的操作平台上的应用程序编程接口 (API) 进行应用程序的开发、使用、经销或分发为目的，您可以任何形式对这些样本程序进行复制、修改、分发，而无须向 IBM 付费。这些示例并未在所有条件下作全面测试。因此，IBM 不能担保或暗示这些程序的可靠性、可维护性或功能。样本程序是“按现状”提供的，不附有任何种类的保证。对于因使用样本程序而引起的任何损害赔偿，IBM 不承担责任。

如果您正在查看本信息的软拷贝，图片和彩色图例可能无法显示。

商标

IBM、IBM 徽标和 ibm.com 是 International Business Machine Corp.，在全球许多管辖区域注册的商标或注册商标。其他产品和服务名称可能是 IBM 或其他公司的商标。Web 站点 www.ibm.com/legal/copytrade.shtml 上的“Copyright and trademark information”部分中提供了 IBM 商标的最新列表。

Adobe 和 Adobe 徽标是 Adobe Systems Incorporated 在美国和/或其他国家或地区的注册商标或商标。

Intel、Intel 徽标、Intel Xeon 和 Pentium 是 Intel Corporation 或其分公司在美国和其他国家或地区的商标或注册商标。

Linux 是 Linus Torvalds 在美国和/或其他国家或地区的注册商标。

Microsoft、Windows、Windows NT 和 Windows 徽标是 Microsoft Corporation 在美国和/或其他国家或地区的商标。

UNIX 是 The Open Group 在美国和其他国家或地区的注册商标。

Java 和所有基于 Java 的商标和徽标是 Oracle 和/或其子公司的商标或注册商标。
其他产品和服务名称可能是 IBM 或其他公司的商标。

索引

[A]

安全套接字层 (SSL) 43

[C]

出版物

访问 61

磁盘驱动器

备件 16

备件 CIM 结构 16

备用扩展数据块 17

故障关联 17

磁盘驱动器 Lite 概要文件 15

存储池

创建 31

修改 32

存储管理计划规范 (SMI-S) 1

支持的概要文件 49

存储卷

参数 32

创建 32

创建 FlashCopy 35

在不同集群系统中的卷之间创建同步关系 37

在位于相同集群系统中的卷之间创建同步关系 36

针对同步集创建 FlashCopy 36

存储卷镜像图 29

存储器配置任务 31

错误码 51

[D]

代理程序代码, 已定义 2

单元类型, 列表 45

读者反馈, 发送 xii

对象管理器适配器概要文件 26

对象路径, 已定义 2

多计算机系统概要文件 25

[F]

发送

意见 xii

法律声明

商标 64

声明 63

返回码 51

方法, 受支持 49

方法, 已定义 2

访问

出版物 61

访问点的子概要文件 9

服务

高速镜像 (同步) 35

全局镜像 (异步) 35

FlashCopy (即时) 35

服务器概要文件 27

服务位置协议 (SLP) 3

已定义的 1, 2

辅助功能选项

重复速度

向上和向下按钮 61

键盘 61

快捷键 61

[G]

概念 7

比较 CIM 和 Storwize V7000 7

已定义的 2

CIM 代理程序 2

概述 1, 3

概要文件和子概要文件

磁盘驱动器 Lite 概要文件 15

对象管理器适配器概要文件 26

多计算机系统概要文件 25

访问点的子概要文件 9

服务器概要文件 27

机柜概要文件 18

集群系统子概要文件 13

拷贝服务 14

块服务概要文件 11

块服务器性能子概要文件 9

扩展数据块组合子概要文件 18

屏蔽和映射概要文件 24

软件概要文件 28

位置子概要文件 23

作业控制概要文件 23

FC 端口概要文件 19

interop 概要文件 20

RAID 支持 13

SMI-S, 受支持 49

高速镜像 (同步) 35

公共信息模型 (CIM), 已定义 1

功能图 9

磁盘驱动器备件 CIM 结构 16

磁盘驱动器故障关联 17

磁盘驱动器 Lite 概要文件 15

存储卷镜像 29

功能图 (续)

对象管理器适配器概要文件 26

多计算机系统概要文件 25

访问点的子概要文件 9

服务器概要文件 27

机柜概要文件 18

集群系统子概要文件 13

拷贝服务 14

块服务概要文件 11

块服务器性能子概要文件 9

扩展数据块组合子概要文件 18

屏蔽和映射概要文件 24

软件概要文件 28

位置子概要文件 23

物理软件包概述 27

作业控制概要文件 23

FC 端口概要文件 19

interop 概要文件 20

iSCSI 关系 21

IsSpare 关联 17

RAID 支持 13

关联, 已定义 2

关于本文档

发送意见 xii

[J]

机柜

类列表 45

已安装单元, 数 45

基于 Web 的企业管理 (WBEM), 已定义 1

集群系统

映射 38

在不同集群系统中的卷之间创建同步关系 37

在位于相同集群系统中的卷之间创建同步关系 36

子概要文件 13

简介 1

键盘

辅助功能选项 61

[K]

拷贝服务

高速镜像 (同步) 35

关系 35

全局镜像 (异步) 35

图 14

拷贝服务 (续)

- 在不同集群系统中的卷之间创建同步关系 37
- 在存储卷之间创建 FlashCopy 35
- 在位于相同集群系统中的卷之间创建同步关系 36
- 针对同步集创建 FlashCopy 36
- FlashCopy (即时) 35
- 客户机应用程序, 已定义 2
- 块服务概要文件 11
- 块服务器性能子概要文件 9
- 快捷键
 - 辅助功能选项 61
 - 键盘 61
- 扩展数据块组合子概要文件 18

[L]

- 类
 - 列表 45
 - 受支持 49
- 类, 已定义 2
- 连接数据, 手动设置 43

[M]

- 名称空间, 已定义 2
- 模式, 已定义 2

[P]

- 配置
 - 创建存储池 31
 - 创建存储卷 32
 - 创建 RAID 阵列 33
 - 修改存储池 32
- 屏蔽和映射概要文件 24

[Q]

- 全局镜像 (异步) 35

[R]

- 任务, 存储器配置 31
- 日志文件 46
- 软件概要文件 28

[S]

- 商标 64
- 设备供应商, 已定义 2
- 设备, 已定义 2
- 实例, 已定义 2

示例

- 典型配置 3
- 手动设置连接数据 43
- 受管对象格式 (MOF) 文档 6, 49
- 受管对象格式 (MOF), 已定义 2
- 属性, 已定义 2

[T]

- 体系结构图 9
- 统计信息, 性能 46
- 图 9
 - 磁盘驱动器备件 CIM 结构 16
 - 磁盘驱动器故障关联 17
 - 磁盘驱动器 Lite 概要文件 15
 - 存储卷镜像 29
 - 对象管理器适配器概要文件 26
 - 多计算机系统概要文件 25
 - 访问点的子概要文件 9
 - 服务器概要文件 27
 - 工作中的典型 CIM 代理程序 3
- 机柜概要文件 18
- 集群系统上的 CIMOM 3
- 集群系统子概要文件 13
- 拷贝服务 14
- 块服务概要文件 11
- 块服务器性能子概要文件 9
- 扩展数据块组合子概要文件 18
- 屏蔽和映射概要文件 24
- 软件概要文件 28
- 位置子概要文件 23
- 物理软件包概述 27
- 作业控制概要文件 23
- FC 端口概要文件 19
- interop 概要文件 20
- iSCSI 关系 21
- IsSpare 关联 17
- RAID 支持 13

[W]

- 网络注意事项 43
- 维护任务 45
- 位置子概要文件 23
- 文档
 - 改进 xii
- 物理软件包概述图 27

[X]

- 限定符, 已定义 2
- 相关信息 x
- 信息
 - 中心 x
- 性能统计信息 46

[Y]

- 已安装单元, 数 45
- 意见, 发送 xii
- 引用, 已定义 2
- 映射 7
 - 代码样本, FlashCopy
 - 自变量生成器类 38
 - getCluster 方法 38
 - getConfigService 方法 38
 - getFCMappings 方法 38
 - getVolumes 方法 38
 - Java 主方法 38
 - makeFlashCopyMapping 方法 38
 - prepareFCMapping 方法 38
 - startFCMapping 方法 38
 - CIM 概念与 Storwize V7000 概念 7
- 远程连接访问 43

[Z]

- 诊断任务 45
 - 收集日志文件 46
- 阵列创建
 - RAID 33
- 指示服务 26, 46
- 指示, 已定义 2
- 主要组件 2, 3
- 子概要文件
 - 访问点 9
 - 集群系统 13
 - 块服务器性能 9
 - 扩展数据块组合 18
 - 位置 23
- 自变量生成器类 38
- 作业控制概要文件 23

A

- Aperi 6

C

- CIM 对象管理器 (CIMOM), 已定义 2
- CIM 客户机框架 6
- CIMOM 编程 6

D

- DiskDriveExtent 故障 17

E

- enclosure
 - 单元类型列表 45

enclosure (续)
概要文件 18

F

FC 端口概要文件 19

FlashCopy

代码样本

自变量生成器类 38

getCluster 方法 38

getConfigService 方法 38

getFCMappings 方法 38

getVolumes 方法 38

Java 主方法 38

makeFlashCopyMapping 方法 38

prepareFCMapping 方法 38

startFCMapping 方法 38

关于 35

在存储卷之间创建 35

针对同步集创建 36

G

getCluster 方法 38

getConfigService 方法 38

getFCMappings 方法 38

getVolumes 方法 38

I

interop 概要文件 20

iSCSI 关系图 21

IsSpare 关联 17

J

Java 主方法 38

Java WBEM Service 项目 6

M

makeFlashCopyMapping 方法 38

P

prepareFCMapping 方法 38

R

RAID

阵列创建 33

支持图 13

RemoteServiceAccessPoint (RSAP) 实例

43

S

SBLIM CIM Client for Java 6

SMI-S 1

SMI-S 概要文件 49

startFCMapping 方法 38

Storwize V7000 库

相关出版物 x



Printed in China

G151-1562-00

