

StorageTek Enterprise Library Software

灾难恢复和异地数据管理指南

发行版 7.3

E63463-02

2016 年 9 月

StorageTek Enterprise Library Software

灾难恢复和异地数据管理指南

E63463-02

版权所有 © 2015, 2016, Oracle 和/或其附属公司。保留所有权利。

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

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

如果将本软件或相关文档交付给美国政府，或者交付给以美国政府名义获得许可证的任何机构，则适用以下注意事项：

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

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

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

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

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

目录

| | |
|---|----|
| 前言 | 13 |
| 目标读者 | 13 |
| 先决条件 | 13 |
| 关于本书 | 13 |
| 如何选择满足需求的解决方案? | 14 |
| 文档可访问性 | 15 |
| 新增内容 | 17 |
| 1. 灾难恢复介绍 | 19 |
| 定义恢复时间目标 (Recovery Time Objective, RTO) | 20 |
| 定义恢复点目标 (Recovery Point Objective, RPO) | 20 |
| 处理临时中断 | 21 |
| 关键概念: 同步点恢复 | 21 |
| 将 RPO 与同步点恢复联系起来 | 22 |
| 规划数据高可用性 (Data High Availability, D-HA) | 23 |
| 高可用性物理磁带 | 23 |
| 高可用性虚拟磁带 | 24 |
| D-HA 和同步点恢复 | 26 |
| 执行真正的灾难恢复 | 26 |
| 规划 DR 测试 | 27 |
| DR 测试的数据移动 | 27 |
| 通过物理导出/导入进行 DR 测试 | 28 |
| 通过 CDRT 进行 DR 测试 | 29 |
| 通过 VSM 跨 Tapeplex 复制进行 DR 测试 | 30 |
| 2. 执行物理导出和导入 | 33 |
| 按管理类导出和导入 | 33 |
| 示例: 按管理类从源 VSM 系统导出 | 34 |
| 示例: 按管理类导入目标 VSM 系统 | 35 |
| 按存储类导出和导入 | 36 |
| 示例: 按存储类从源 VSM 系统导出 | 37 |
| 示例: 按存储类导入目标 VSM 系统 | 37 |

| | |
|---|-----------|
| VLE 导出/导入 | 39 |
| 3. 使用 ELS 外部保管功能 | 41 |
| 准备进行 ELS 外部保管 | 41 |
| 为 DR 和 LTR 创建 MVC | 42 |
| 使用保管功能时的 DELSCR 考虑事项 | 43 |
| 保管卷返回到 ACS 时会发生什么? | 43 |
| 保管用于 DR 的 MVC | 43 |
| 含 MVC 的基本 DR 保管 | 43 |
| 第 1 步—创建保管库 VTV/MVC | 44 |
| 第 2 步—导出保管库 MVC | 44 |
| 第 3 步—(可选) 将其他数据集写入清单文件磁带 | 45 |
| 第 4 步—弹出保管库 MVC | 45 |
| 第 5 步—弹出本机卷 (包括清单文件磁带) | 46 |
| 第 6 步—创建要从保管库中返回的卷的拉式列表 | 46 |
| 第 7 步—准备要返回的保管 MVC | 47 |
| 第 8 步—准备要返回的保管本机卷 | 48 |
| 第 9 步—装入返回的卷 | 49 |
| 长达数周的含 MVC 的 DR 保管 | 50 |
| 远程磁带库中的含 MVC 的 DR 保管 | 51 |
| 保管用于 LTR 的 MVC | 52 |
| 将特定的卷弹出至本地 (基础) 保管库 | 53 |
| 4. 在 DR 解决方案中使用跨 TapePlex 复制 | 55 |
| CTR 如何工作? | 55 |
| CTR VTV 只读注意事项 | 56 |
| 配置 CTR | 57 |
| 设置: 配置和启动 CTR | 58 |
| 为 CTR 定义策略 | 60 |
| 发送 TapePlex 的策略 | 60 |
| 接收 TapePlex 的策略 | 62 |
| 当远程站点没有 LPAR 时使用 CTR | 63 |
| 使用 CTR 作为 DR 解决方案 | 65 |
| 使用 CTR 实现业务连续性 | 65 |
| 使用 CTR 实现业务恢复 | 66 |
| 使用跨 Tapeplex 复制进行灾难恢复测试 | 67 |
| 当 DR 站点没有 LPAR 时进行 DR 测试 | 68 |

| | |
|--|-----------|
| 管理使用跨 TapePlex 复制 (Cross-TapePlex Replication, CTR) 复制的 VTV | 69 |
| 5. 配置远程磁带库 | 71 |
| 修改 SMC SCMDS 文件 | 71 |
| 更新 VTCS CONFIG 面板以定义远程磁带库 | 72 |
| MVC 池注意事项 | 72 |
| 6. 使用群集 VTSS 配置 | 75 |
| 什么是群集 VTSS? | 75 |
| 群集 VTSS 要求 | 76 |
| 群集 VTSS 配置的工作原理 | 78 |
| VTSS 调节工作原理 | 79 |
| 单向和双向群集 | 79 |
| 单向群集 | 80 |
| 单向 VTSS 群集工作原理 | 80 |
| 双向群集 | 81 |
| 双向 VTSS 群集工作原理 | 81 |
| 扩展群集 | 82 |
| 同步或异步复制 | 83 |
| 实施同步复制 | 83 |
| 利用作业监视实施异步复制 | 83 |
| 使用 TCP/IP 连接形成群集 | 85 |
| TCP/IP 环境 | 85 |
| 为 TCP/IP CLINK 配置 VTCS | 85 |
| CONFIG CLINK 语句 | 86 |
| 7. 使用 Concurrent Disaster Recovery Test 软件 | 87 |
| 元数据注意事项 | 88 |
| CDRT 从何处获得其 VTV 数据? | 88 |
| 测试结束时数据会发生什么情况? | 89 |
| 管理 CDRT 资源 | 89 |
| 卷资源 | 89 |
| 暂存子池 | 90 |
| MVC 资源 | 90 |
| VTSS 资源 | 91 |
| 非共享 VTSS 资源 | 91 |
| CDRT 中的群集 VTSS | 91 |

| | |
|---|------------|
| 为 DR 测试准备 VTSS 群集 | 91 |
| 在 DR 测试前后管理 VTSS 内容 | 92 |
| 共享 VTSS 资源 | 93 |
| ACS 资源 | 93 |
| DR 测试 ACS 限制 | 93 |
| DR 生产主机要求 | 93 |
| VLE 资源 | 94 |
| VTCS 策略 | 94 |
| 为非共享 VTSS 定义 MGMTCLAS/STORCLAS | 95 |
| 为共享 VTSS 定义 MGMTCLAS | 95 |
| 优化对测试资源和生产资源的访问 | 95 |
| 运行 DR 测试 | 95 |
| DR 测试完成后进行清理 | 97 |
| DR 测试完成后进行清理 | 97 |
| 恢复正常操作 | 98 |
| 操作方案 | 98 |
| 方案 1: 生产和测试站点, 其中每个站点均有 ACS, 且测试站点有备 用 VTSS | 99 |
| 方案 1 的示例 JCL | 100 |
| 方案 2: 生产站点和测试站点, 其中每个站点均有 ACS, 且测试站点 有 VTSS 接管 | 101 |
| 方案 2 的示例 JCL | 101 |
| 方案 3: 生产站点和测试站点, 其中每个站点均有 ACS, 无 VTSS | 102 |
| 方案 3 的示例 JCL | 103 |
| 方案 4: 生产站点和 DR 测试站点的群集 VTSS | 103 |
| 方案 4 的示例 JCL | 104 |
| 方案 5: 生产站点和测试站点, 其中每个站点均有 ACS 和 VLE | 105 |
| 方案 5 的示例 JCL | 106 |
| 方案 6: 生产站点和测试站点, 其中每个站点只有 VLE | 107 |
| 方案 6 的示例 JCL | 108 |
| 方案 7: 生产站点和 DR 测试站点的群集 VTSS (无磁带) | 109 |
| 方案 7 的示例 JCL | 109 |
| 8. 在 VSM 环境中创建系统恢复点 | 111 |
| 检查点示例 | 112 |
| 示例 1: 本地 MVC 副本和远程 MVC 副本 | 112 |
| 示例 2: 使用 CONFIG RECLAIM PROTECT | 114 |
| 9. 将 VLE 用于灾难恢复 | 115 |

| | |
|--|------------|
| 正常的生产模式 | 115 |
| 使用 VLE 运行 DR 测试 | 116 |
| 使用 VLE 运行 DR 测试之后清除 | 119 |
| 使用 VLE 实现业务连续性 | 120 |
| A. 群集 VTSS 示例 | 123 |
| 单向群集 VTSS | 123 |
| 配置和管理单向群集 VTSS 系统 | 125 |
| 双向群集 VTSS | 128 |
| 配置和管理双向群集系统 | 130 |
| 扩展群集 | 133 |
| 配置和管理包含 3 个 VTSS 的群集系统 | 134 |
| 带有 TCP/IP CLINK 的 VSM5 到 VSM5 群集 | 137 |
| 带有 TCP/IP CLINK 和交叉连接 VLE 的 VSM5 到 VSM6 群集 | 138 |
| 带有 TCP/IP CLINK 的 VSM6 到 VSM6“无磁带”群集 | 140 |
| 应该使用单向还是双向群集? | 141 |
| 词汇表 | 143 |
| 索引 | 157 |

表格清单

| | |
|---------------------------------|-----|
| 6.1. 群集 VTSS 要求 | 76 |
| 6.2. 同步复制要求 | 77 |
| 6.3. 增强复制要求 | 77 |
| A.1. VSMR1 的 CLINK IPIF 值 | 137 |
| A.2. VSMR2 的 CLINK IPIF 值 | 137 |
| A.3. VSMR1 的 CLINK IPIF 值 | 139 |
| A.4. VSMR1 的 RTD IPIF 值 | 139 |

示例清单

| | |
|-----------------------------------|----|
| 4.1. TapePlex A 的 CONFIG 面板 | 59 |
| 4.2. TapePlex B 的 CONFIG 面板 | 59 |

前言

Oracle StorageTek Enterprise Library Software (ELS) 是包含以下基本软件的解决方案：

- StorageTek Storage Management Component (SMC)
- StorageTek Host Software Component (HSC) StorageTek Virtual Tape Control Software (VTCS)
- StorageTek Concurrent Disaster Recovery Test (CDRT)

此外，以下软件也随 ELS 软件包提供：

- StorageTek Library Content Manager (LCM) (原 ExLM) 。 LCM 包括一个以前称为异地保管库功能 (Offsite Vault Feature) 的产品的增强版本。
- StorageTek Client System Component for MVS Environments (MVS/CSC)
- StorageTek LibraryStation

目标读者

本指南面向负责规划和实施灾难恢复 (Disaster Recovery, DR) 和异地数据管理解决方案的 StorageTek 或客户人员。

先决条件

要执行本指南中介绍的任务，您应该已了解以下内容：

- Z/OS 操作系统
- JES2 或 JES3
- Enterprise Library Software (ELS)

关于本书

《ELS 灾难恢复和异地数据管理指南》包含以下内容：

- [第 1 章 灾难恢复介绍](#)，是对灾难恢复解决方案的介绍和概述。
- 有关异地数据管理，请参见：
 - [第 2 章 执行物理导出和导入](#)，介绍了如何使用 EXPORT 和 IMPORT 工具。EXPORT 可以在源站点创建可移植的 MVC，您随后将其从源 ACS 中弹出，以物理方式将其传送到另一个站点，再使用 IMPORT 将其装入目标站点。

- [第 3 章 使用 ELS 外部保管功能](#)，介绍了如何使用该 DR 功能，进而使用 EXPORT 和 IMPORT 加上额外的命令和实用程序来自动执行异地保管 MVC 的过程。
- [第 5 章 配置远程磁带库](#)，显示了包含远程磁带库的配置样例以及有关如何配置远程磁带库的信息。
- [第 6 章 使用群集 VTSS 配置](#)，提供了您在实施群集 VTSS 方面想要了解的所有内容（而且还远不止于此），包括 ELS 7.0 及更高版本中的新功能—扩展群集（每个群集 3 个以上 VTSS）。严格地说，单独的群集 VTSS 不是 DR 解决方案。但是，由于群集 VTSS 具备冗余/备份/分布式功能，因此很容易将自身带向 DR 解决方案。例如，我设置了管理类和存储类，以便创建两个 MVC 副本，一个在本地站点，另一个在远程站点。有关更多信息，另请参见[附录 A, 群集 VTSS 示例](#)。
- 有关特定 DR 解决方案，请参见：
 - [第 4 章 在 DR 解决方案中使用跨 TapePlex 复制](#)，介绍了如何使用跨 Tapeplex 复制功能为磁带数据设置完整的灾难恢复解决方案。
 - [第 7 章 使用 Concurrent Disaster Recovery Test 软件](#)，介绍了如何使用该功能验证站点的 DR 解决方案。请注意，对于 ELS 7.2，CDRT 支持 Virtual Library Extension (VLE) 1.0 及更高版本。
 - [第 8 章 在 VSM 环境中创建系统恢复点](#)，介绍了如何在 VSM 环境中实施 DR 恢复点。
 - [第 9 章 将 VLE 用于灾难恢复](#)，介绍了如何使用 VLE (Virtual Library Extension) 作为灾难恢复解决方案，用以提供一种简化且无中断的 DR 测试执行方法。

总体来讲，所有这些章节介绍了将数据从一个位置移至另一个位置以及管理数据的方法。这些过程可与用于实现 DR/业务连续性/业务恢复的其他过程一起使用。借助 ELS 保管库功能，您可以管理异地 DR 和长期保留 (Long Term Retention, LTR) 卷，还能管理手动机架挂载（“基础”）卷。

所以，您拥有全面的 DR 和异地数据管理解决方案。如何从这些产品中为您的站点选择最佳方案呢？

如何选择满足需求的解决方案？

《ELS 灾难恢复和异地数据管理指南》中介绍的解决方案有着不同的成本和附带好处，一般的经验法则是解决方案成本越高，恢复速度越快或数据管理周期越短。例如，MVC 的导出/导入（可以选择使用 ELS 外部保管库功能）成本相对较低，但是恢复需要一些时间，就是因为需要将 MVC 从一个站点移至另一个站点。

不过，跨 TapePlex 复制 (Cross-TapePlex Replication, CTR) 成本相对较高。您需要两个站点、两个站点上多个 VTSS、RTD 和 MVC 等等。CTR 使用 VTSS 群集功能通过 FICON 或 ESCON 群集链路直接将数据从一个 TapePlex 中的 VTSS 复制到另一个 TapePlex 中的 VTSS。数据移动（以及元数据移动）在站点之间以电子方式进行，所以备份和恢复速度可能比使用外部保管库时要快得多。

文档可访问性

有关 Oracle 对可访问性的承诺，请访问 Oracle Accessibility Program 网站 <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>。

获得 Oracle 支持

购买了支持服务的 Oracle 客户可通过 My Oracle Support 获得电子支持。有关信息，请访问 <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info>；如果您听力受损，请访问 <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs>。

新增内容

此修订版包括以下更新：

- 为按管理类的 MVC 导出和导入添加注释。

请参见“[按管理类导出和导入](#)”。

- 为按存储类的 MVC 导出和导入添加注释。

请参见“[按存储类导出和导入](#)”。

- 为 VLE 的导入/导出添加了一个过程。

请参见“[VLE 导出/导入](#)”。

- 为群集 VTSS 说明添加了一个注释，该注释说明同步复制或增强复制与缺少中断处理程序 (Missing Interrupt Handle, MIH) 的关系。

请参见“[什么是群集 VTSS?](#) ”

- 添加了增强复制要求表。

请参见“[群集 VTSS 要求](#)”。

第 1 章 灾难恢复介绍

从根本上说，企业灾难恢复 (Disaster Recovery, DR) 的最佳做法包括设计和实施可以抵御灾难 (“业务连续性”) 并恢复正常运行 (“业务恢复”) 的容错硬件和软件系统，同时干预最少，理想情况下没有数据丢失。构建容错环境来实现企业 DR 目标并满足实际预算约束可能成本高昂而且耗费大量时间，并且需要企业强有力的承诺。

DR 规划通常可以应对下面一种或多种类型的灾难：

- 由于自然灾害 (地震、风暴、洪水等) 或其他原因 (火灾、故意破坏、盗窃等) 导致的大面积或广泛性 IT 设备损坏。
- 广泛失去 IT 设备关键服务，例如断电、无法进行冷却或网络访问。
- 失去关键人员。

DR 规划过程首先是确定业务必须抵御并从其恢复运行的灾难类型并描述其特征。规划过程要确定高层面的业务连续性 (business continuance, BC) 和业务恢复 (business resumption, BR) 要求，包括必需的容错程度。DR 规划的产物是一个恢复体系结构，使容错系统、应用程序和数据能够在既定的约束下满足这些要求。典型的 DR 约束包括恢复时间目标 (recovery time objective, RTO)、恢复点目标 (recovery point objective, RPO) 和可用预算。DR 体系结构加上业务约束使得 DR 过程以真正的“端到端”方式集成所有系统元素，以保证整个 DR 过程产生可预测的结果。

容错系统通常通过冗余实现稳健性和弹性。全冗余系统构建成本非常高昂，这种体系的体系中不会发生单点故障，可以在其限度内可能最严重的灾难发生期间运行并从灾难中恢复运行。航天飞机和航空飞机控制系统就是很好的全冗余系统的例子。不太关键的 IT 应用程序通常使用冗余较低的不太强大的系统。这些系统构建成本较低，灾难过后必然会引发服务中断，在中断期间，企业将努力复原可恢复的系统、应用程序和数据。

最后，企业的性质、客户的要求和 DR 的可用预算构想 DR 要求的关键因素。全面的 DR 解决方案可能成本非常高昂，但是不得不构建。您不能让资金、硬件和软件面临潜在的灾难，盼望着抵御灾难并恢复业务运营。不过，如果您明智地规划和构建，可能不得不遭受更长时间的中断、服务降级或两者，直到全部服务可以恢复，但是您仍然可以拥有一个可靠的有限 DR 解决方案。

然而要明白，或许再多的规划也不能预料和应对全部可能的 DR 情形。例如，起初一个系统上有一个显然微不足道的问题，该问题可能会随着时间的推移蔓延进而以不同方式影响其他系统，全部加起来造成了没有恢复情形的灾难。同样，如果关键的假设不成立 (例如，如果关键部件或服务不可用，或者如果 DR 提供商的交付能力不像宣

传的那样强大)，企业履行服务协议的能力可能会变差。不过，真正关键在于，如果发生的灾难超过您规划的最严重的情形，可能无法进行恢复。

定义恢复时间目标 (Recovery Time Objective, RTO)

RTO 是一个服务级别目标，是指在灾难发生后达到所需运行能力所需的时间。例如，业务要求可能规定这样的 RTO：所有生产系统在将会持续超过一个小时的任何计划外中断发生后的 30 分钟内启动并以灾难前能力的 80% 运行（如果不存在 DR 能力）。发生灾难后 RPO 的处理时间、是否有合格的 IT 人员以及所需手动 IT 流程的复杂性就是可能会使 RTO 确定成形的约束的例子。RTO 不适用于全容错系统，因为这些系统在灾难期间和灾难过后隐式恢复，不会造成服务中断。

DR 规划者可能会根据定义的部分或全部 BC 要求设置不同的 RTO。不同类型的业务运营可能需要不同的 RTO，例如联机系统与批处理窗口需要不同的 RTO。进一步说，可能会在一个分阶段 DR 规划的各个阶段应用不同的 RTO，其中每个阶段都有一个定义的 RTO。更进一步说，对于可恢复的应用程序，各个服务级别中的每一个级别可能有不同的 RTO。

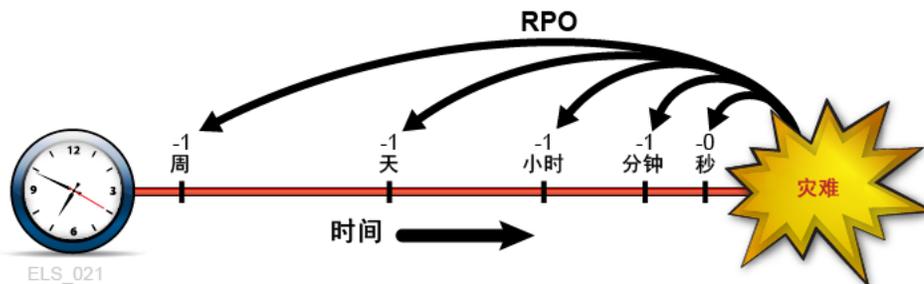
BC 数据可用性要求对于 RTO 规划极其重要。当灾难恢复站点中不存在必须输入 DR 恢复过程的数据时，现场检索这些数据所需的时间将会延迟 RTO。例如，位于异地保管库中的数据需要一些时间来检索。如果开始灾难恢复操作之前在恢复站点复制了最新的输入数据，恢复便可快速进行。

定义恢复点目标 (Recovery Point Objective, RPO)

RPO 是一个业务连续性目标，规定灾难恢复过程复原所有可恢复系统后达到的业务状态或业务状况。从概念上讲，RPO 是灾难发生前的已知“回滚”或同步目标状态。也就是说，RPO 是中断的可恢复应用程序可从其恢复处理的灾难恢复后的点。在 RPO 与发生灾难的时间之间的间隔内发生的所有事务都不可恢复。RPO 不适用于全容错系统，因为灾难并不影响这些系统的业务连续性。

图 1.1 “恢复点目标”通过建议多个恢复点供 DR 规划者考虑来阐明 RPO 的概念。规划必须确保所需的 RPO 相对于选择的 RTO 可行，反之亦然。一般来说，与比较远的 RPO 相比，要求 RPO 比较接近灾难发生时间的灾难恢复规划需要更高的容错并且实施成本更高。与 RTO 一样，DR 规划者可能会根据不同的 BC 要求、DR 规划阶段或应用程序服务级别设置不同的 RPO。

图 1.1. 恢复点目标



更广泛地说，RPO 规划必须确定要复原每个可恢复系统而必须存在的所有支持元素，包括数据、元数据、应用程序、平台、设备和人员。规划还必须确保这些元素在所需的业务状况级别可用以实现恢复。BC 数据状况要求对于 RPO 规划尤为重要。例如，如果 BC 要求规定了一小时的 RPO，则输入恢复过程的任何数据或元数据都必须在 RPO 之前保持最新，否则将无法实现 RPO。组织的 DR 过程将指定在规定的 RTO 内实现定义的所有 RPO 的步骤。

RPO 恢复需要的系统元数据包括 OS 目录结构和磁带管理系统信息。这些项目必须在灾难恢复过程中进行更新才能启用选择的所有 RPO。例如，要确保 DR 恢复过程的各种元数据输入之间的一致性，必须为将在 RPO 时重新创建的现有数据集取消目录编制；必须将在 RPO 与灾难发生时间之间更新的数据集恢复到 RPO 时或之前存在的版本；必须将与磁带相关的任何目录更改与磁带管理系统同步。

处理临时中断

对于会使生产站点长期不可使用的很长时间的中断，灾难恢复提供了补救措施。虽然本章剩余部分讲解灾难恢复做法，但是制定过程来减轻相对较短的中断（如果任其发展，可能会对生产造成负面影响）可能同样重要。例如，考虑一下这样的服务中断：某些硬件或网络设备在一两个小时内不可用，但是通过一些快速的临时调整，此中断期间可以在“降级模式”下继续生产。临时中断过程将记录如何隔离问题、做出哪些更改、通知哪些人员以及在恢复服务后如何恢复到正常的操作环境。

关键概念：同步点恢复

在定义的 RPO 重新启动生产应用程序是在真正的灾难恢复期间和 DR 测试期间执行的关键操作。最具弹性的 DR 环境将确保每个可恢复的应用程序（无论是来源于他还是企业内部开发）都实施核心 DR 要求：也就是说，将应用程序设计成从计划的时间点（称为同步点）重新启动，以减轻应用程序执行期间计划外中断产生的影响。在同步点重新启动中断的应用程序时，结果就像应用程序未曾中断一样。

可恢复应用程序的重新启动过程取决于应用程序及其输入的性质。很多时候，真正的灾难恢复或 DR 测试期间的应用程序重新启动过程与正常生产运行期间发生故障时用于重新启动应用程序的过程一样。在可能的情况下，在真正的灾难恢复或 DR 测试期间重用生产重新启动过程可以简化 DR 过程的创建和维护，并利用这些久经验证的过程。在最简单的情况下，一个可恢复的应用程序是只有一个同步点的单个作业步骤，该同步点是该步骤调用的程序的开端。在这种情况下，恢复过程可能像重新提交中断的作业那样简单。再稍微复杂的重新启动过程可能涉及取消应用程序在上次运行期间生成的所有输出数据集的目录编制，然后重新启动应用程序。

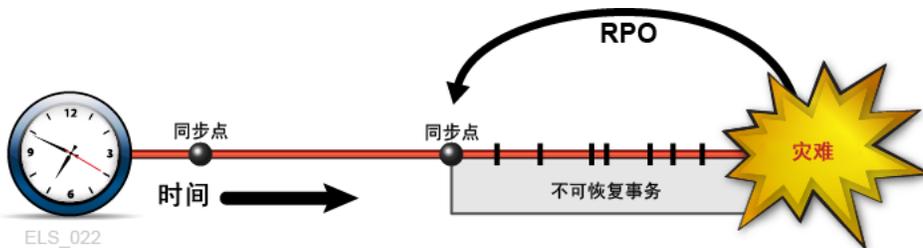
如果应用程序具有多个可供选择的内部同步点，其重新启动过程可能并不那么简单。使用检查点/重新启动技术实施这些同步点的应用程序定期记录其进度，例如可以使用记录的检查点信息在中断之前记录的最后一个内部同步点重新启动。重新启动过程将符合每个同步点的要求。如果正在使用检查点，当检查点对应用程序恢复仍然有效时，与检查点关联的数据集不能过期、已取消目录编制或暂存。要为修改其现有输入数据集的作业步骤建立同步点，一种简单的方法是在运行该步骤之前为每个可

修改的数据集创建备份副本。通过在 DD 语句中或在动态分配请求中搜索 JCL 属性 DISP=MOD 可以轻松识别这些可修改的输入数据集。如果一个作业步骤失败或中断，只需丢弃所有修改的输入数据集、从备份副本恢复这些输入数据集，并从恢复的副本重新启动该步骤。这些备份副本对于重新启动失败或中断的作业步骤（原数据已过期、已取消目录编制或暂存）也很有用。

将 RPO 与同步点恢复联系起来

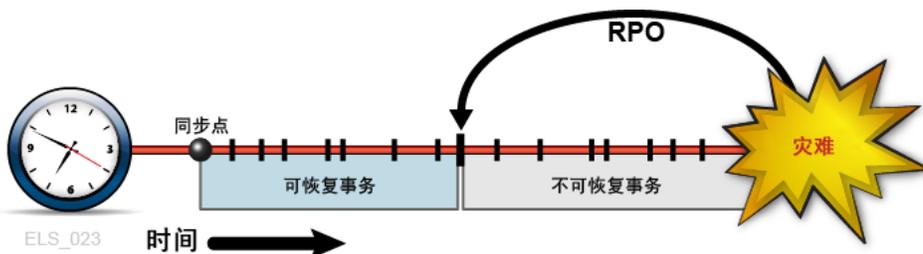
当 RPO 与某个同步点保持一致时，执行为该同步点制定的应用程序重新启动过程将从该起点恢复应用程序，就像未发生中断一样（图 1.2 “同步点处的 RPO”）。在该 RPO 后直到发生灾难时处理的所有事务都假定为不可恢复。

图 1.2. 同步点处的 RPO



在其他时候，BC 要求可能证明将 RPO 置于同步点之间是合理的。在这些情况下，同步点之间的恢复依赖于描述任何关键应用程序状态更改或建立最新同步点后所发生事件的补充数据。例如，考虑一下灾难发生前一分钟的 RPO。假设将一个可恢复的应用程序设计成使用检查点记录其进度，但是假设无法承受在一分钟的时间间隔内创建这些检查点的开销。一个解决方案是降低创建检查点的频率并记录在检查点之间提交的所有事务。该事务日志随后将成为检查点恢复过程用于从最新同步点之后的某个 RPO 重新启动的补充输入数据。在此示例中，应用程序重新启动过程将访问最新的检查点数据并应用补充事务日志来复原在检查点之后且在 RPO 之前处理的所有已提交事务（图 1.3 “同步点之间的 RPO”）。这样一来，同步点恢复便可使用来自多个来源的输入数据实现目标 RPO。在 RPO 后直到发生灾难时处理的所有事务都假定为不可恢复。

图 1.3. 同步点之间的 RPO



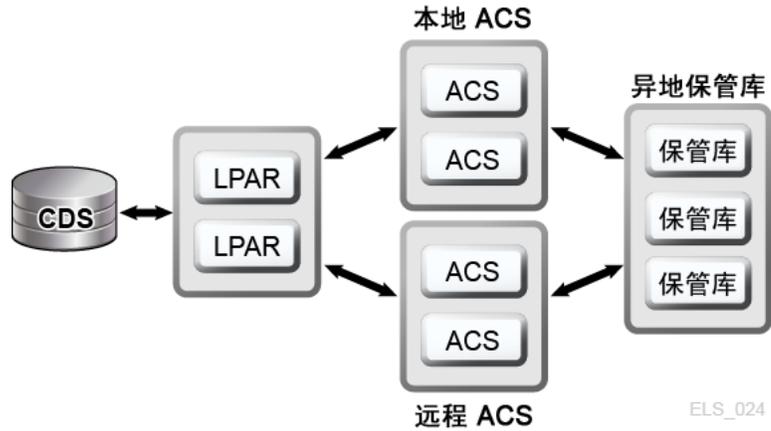
规划数据高可用性 (Data High Availability, D-HA)

数据通常是企业拥有的最宝贵的资产之一。许多公司格外注重并进行额外的投资来预防业务关键型数据丢失，并确保数据在需要时可用于预期目的。如果一家公司无法应付关键数据丢失，很有可能要遭受灾难性的后果。防止数据丢失最常见的方法或许就是将关键数据的副本存储在不同的存储介质或子系统上，并将其中某些副本存储在不同的物理位置。存储在可移除存储介质（包括盒式磁带、CD-ROM 和 DVD）上的副本通常保管在异地存储位置。通常还会在 IT 设备现场存储额外的副本，应用程序可以在 IT 设备中处理该数据。创建和存储关键数据副本可以提高数据冗余并改进数据容错。对于可移除介质，尤其是对于盒式磁带来说，仅仅提高数据冗余通常不足以确保数据对于将要使用它的应用程序而言也具有高可用性。例如，Oracle 的大型机虚拟磁带 VSM 系统将数据存储称为 MVC 的物理磁带卷上。VSM 可以自动创建 MVC 副本以改进数据冗余并降低由于介质故障或错放盒式磁带而引起的风险。生产 VSM 系统利用多个专门的硬件组件来检索存储在 MVC 上的数据，包括 VTSS 缓冲区设备、自动化磁带库和磁带库连接的磁带机（称为 RTD，它们也连接到 VTSS 缓冲区设备）。主机应用程序依靠所有这些 VSM 组件一起运行来从 MVC 中检索数据。尽管大多数人不会将单个组件故障视为与在地震中失去整个数据中心同样严重的灾难，但是如果单个 VSM 关键组件在没有备份的情况下发生故障，无论存在多少个冗余 MVC 副本，当然可能会无法检索任何 MVC 数据。因此，虽然创建 MVC 副本是减轻漏洞和风险的久经验证的最佳做法，但并不总是足以保证在发生故障时具备数据高可用性 (data high availability, D-HA)。对于 DR 规划来说，D-HA 要求是关键业务连续性要求。通常，通过提高冗余以消除会阻止应用程序在存储系统发生故障期间访问数据的单点故障来实现 D-HA。例如，包括冗余组件的 VSM 系统可以改进 VSM 系统容错。安装多个 VTSS 设备、冗余 SL8500 机械手和多个 RTD 目的是消除从应用程序到存储在 MVC 上的关键数据的数据路径上的 VSM 单点故障。VSM 体系结构自始至终设计成支持添加冗余组件来提高冗余并提升 D-HA。

高可用性物理磁带

Oracle 的大型机磁带自动化解决方案通过将数据的冗余副本存储在 TapePlex 内（即，由单个 CDS 映射的复合磁带内）的不同 ACS 中，可使物理磁带应用程序实现 D-HA。例如，在包含单个 TapePlex 的 IT 设备中运行的应用程序可以轻松地将磁带数据集的重复副本存储在该 TapePlex 内的一个或多个 ACS 中。这种技术可通过添加冗余介质、磁带传送装置和自动化磁带库来改进 D-HA。举个简单的例子，应用程序将关键数据集的冗余副本存储在单个 SL8500 磁带库中的两个不同盒式磁带上，该磁带库包含冗余电子部件、每个滑轨上两个机械手，并且每个滑轨上两个或多个与数据集介质兼容的磁带库连接的磁带传送装置。为了消除 SL8500 磁带库作为潜在的单点故障，又向 ACS 中添加了一个 SL8500 来存储关键数据集的更多冗余副本。为了消除 IT 设备本身作为单点故障，可以异地保管冗余数据集副本或在带有通道扩展磁带传送装置的远程 ACS 创建冗余数据集副本（[图 1.4 “FD-HA 物理磁带配置”](#)）。

图 1.4. FD-HA 物理磁带配置

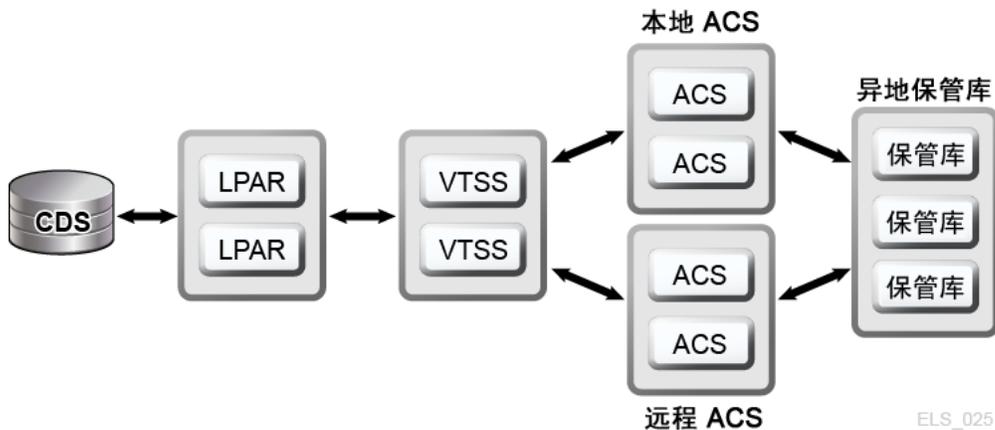


如果每个物理位置都有自己独立的 CDS，即如果每个位置的硬件代表单独的 TapePlex，您也可以在不同的物理位置创建两个或多个物理磁带副本。通过使用 SMC 客户机/服务器功能并定义用于使数据集副本指向远程 TapePlex 的策略，作业可以在另一个 TapePlex 中的 ACS 中创建磁带副本而 JCL 没有变化。

高可用性虚拟磁带

VSM 提供了 MVC 多路复用和群集技术，以使大型机虚拟磁带实现 D-HA。VSM 多路复用涉及在一个或多个 ACS 中创建多个 MVC 副本（例如双工、四工）以提高冗余（图 1.5 “D-HA VSM 多路复用配置”）。接收多路复用副本的 ACS 可以是本地磁带库或带有通道扩展磁带传送装置的远程 ACS。VSM 迁移策略控制驻留在 VTSS 缓冲区中的 VTV 到本地或远程 MVC 的移动，这种移动可能会循环到异地保管库。

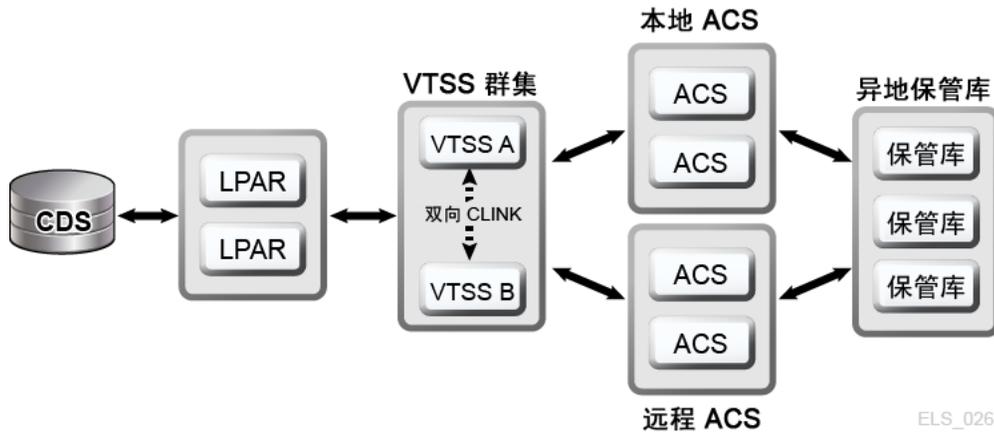
图 1.5. D-HA VSM 多路复用配置



VSM 群集由两个或多个为了通过通信链路 (CLINK) 进行数据交换而联网的 VTSS 设备 (节点) 组成。CLINK 是单向或双向通道。最简单的 VSM 群集配置包括两个 VTSS 节点，它们在使用单向 CLINK 连接的同一个人 TapePlex 中，但是一般部署双向

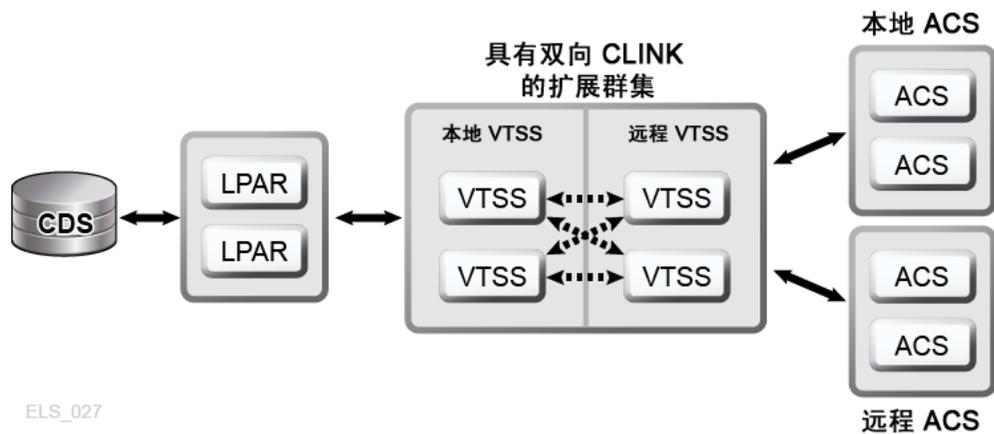
CLINK (图 1.6 “D-HA VSM 群集配置”)。每个群集节点可能位于不同的站点。VSM 单向存储策略控制通过单向 CLINK 从 VTSS A 到 VTSS B 的虚拟磁带卷 (virtual tape volume, VTV) 自动复制。双向存储策略和双向 CLINK 允许从 VTSS A 复制到 VTSS B，反之亦然。

图 1.6. D-HA VSM 群集配置



借助 VSM 扩展群集，可以在一个 TapePlex 中的三个或更多 VTSS 设备之间建立多对多连接，从而实现更高级别的数据可用性 (图 1.7 “D-HA 扩展群集配置 (未显示异地保管库)”)。如图所示在一个 TapePlex 内的两个或多个站点中安装 VTSS 群集设备可通过消除每个站点作为单点故障来提高冗余。

图 1.7. D-HA 扩展群集配置 (未显示异地保管库)

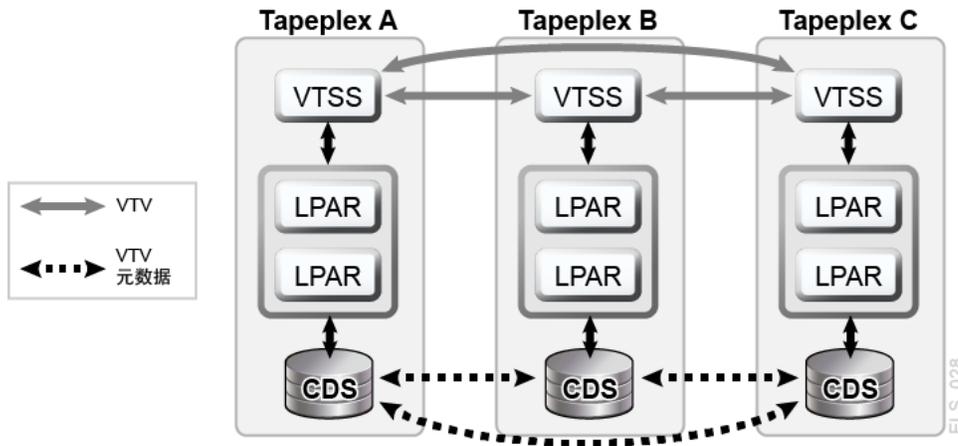


Oracle 的 LCM 产品可通过管理保管库与生产磁带库之间的回收过程来简化 MVC 卷的异地保管过程。当过期数据量超过指定的阈值时，LCM 保管功能将计划退还保管的 MVC 卷。

VSM 跨 Tapeplex 复制 (Cross-Tapeplex Replication, CTR) 群集允许 VTSS 群集设备位于不同的 Tapeplex 中，并提供了将 VTV 从一个 Tapeplex 复制到一个或多个其

他 Tapeplex 的功能，从而通过单向或双向 CLINK 启用多对多群集复制模型（图 1.8 “D-HA VSM 跨 Tapeplex 复制配置”）。发送和接收 Tapeplex 可能位于不同的站点。复制的 VTV 作为只读卷装入到接收 Tapeplex 的 CDS 中。这样提供了强大的数据保护，以防在 Tapeplex 中运行的应用程序对其进行更改。接收 Tapeplex 的 CDS 还会指明 CTR 复制的 VTV 副本由发送 Tapeplex 拥有，作为附加保护措施，CTR 将确保 Tapeplex 无法修改它不拥有的任何 VTV。

图 1.8. D-HA VSM 跨 Tapeplex 复制配置



D-HA 和同步点恢复

创建物理卷的多个副本（MVC 或非 MVC）可以改进数据冗余，但是这些副本对同步点恢复提出了特殊注意事项。对于同步点恢复来说，最重要的一方面就是确保在同步点创建的数据保持只读状态，同时对灾难恢复用途仍然有效。这意味着可用于灾难恢复的物理磁带卷副本必须保持只读状态。要做到这一点，一种方法是将这些副本发送到不存在磁带处理功能的异地保管库位置。请注意，任何经过更改的未受保护的副本都不可用于同步点恢复，因为更新的内容不再反映关联的同步点。虚拟磁带环境添加了一个额外的维度用于管理同步点恢复的多个卷副本。VTV 副本可同时存在于多个 VSM 缓冲区中以及多个 MVC 上。即使当给定 VTV 的所有 MVC 都异地保管时，也可修改留在 VSM 缓冲区现场的 VTV 副本。更新的驻留在缓冲区中的 VTV 副本不得用于同步点恢复，除非该 VTV 属于一个新的同步点，而该同步点使异地保管的副本对灾难恢复用途无效。

执行真正的灾难恢复

真正的灾难恢复操作的成功依赖于具备适当的 DR 站点、训练有素的人员、久经验证的 DR 过程和可恢复的生产工作负荷，生产工作负荷的同步点要满足定义的 RPO，并且所有输入数据和必要的系统元数据能够达到这些 RPO。输入数据和系统元数据必须在需要时可在 DR 站点访问，并且必须在所需的状况级别可用。经过仔细的规划、充分的准备和精心编排的执行，真正的灾难恢复操作可以按规划顺利开展以实现定义的 RPO 和 RTO。当 DR 站点充当生产站点时，在 DR 站点生成的生产数据必须得到充分保护。例如，假设 D-HA 体系结构要求生产工作负荷在三个远程站点复制冗余数

据副本，并且假设 DR 站点在发生灾难之前是这些远程复制站点之一。当生产站点发生灾难而其工作负荷移至 DR 站点后，DR 站点无法再充当目前在该站点本地运行的生产工作负荷的远程复制站点。要满足三个远程复制站点的 D-HA 要求，只要生产工作负荷仍在 DR 站点上运行，就必须使新的第三个远程复制站点联机。此示例说明了全面分析 D-HA 要求如何帮助 DR 规划者明确生产移至 DR 站点后必须满足的所有关键 D-HA 要求。全面的 DR 规划不仅包括在 DR 站点复原生产的活动，而且还包括当生产站点得到修复且为业务做好准备时腾出 DR 站点的过程（假定 DR 站点只是临时的替代生产站点）。例如，当生产站点准备好恢复运行时，必须在该站点复原生产数据。方法包括在 DR 站点与生产站点之间建立双向群集，为在 DR 站点运行的生产工作留出足够的时间，让其通过数据复制重新填充以前的生产站点。但是，可能有必要直接将物理 MVC 传返回复原的生产站点，这也是更及时、更高效的方法。选择的方法取决于灾难恢复后的要求。

规划 DR 测试

真正的灾难恢复就绪性通过测试在指定的 DR 测试站点恢复生产工作负荷的 DR 系统和过程的效率和效用进行评估。DR 测试环境可能是一个专用的 DR 测试平台，但通常在生产和 DR 测试系统之间共享资源更经济。与生产并行执行且使用与生产共享的资源的 DR 测试称为并发 DR 测试。如果某个应用程序必须在生产和 DR 测试系统上并行执行，则 DR 规划者应该确保这两个应用程序实例在并发运行时不会相互干扰。在单独的 LPAR 上隔离生产和 DR 测试系统并限制从 DR 测试系统访问生产数据通常可以提供足够的隔离。DR 测试通常逐渐执行，以允许在不同时间对不同应用程序进行针对性测试，而不是一次测试整个生产环境的恢复。针对性测试是减少 DR 测试系统所需专用硬件量的关键所在。例如，如果某个可恢复应用程序的 DR 测试只需要一小部分 VSM 资源，则这些资源可以在生产和 DR 测试系统之间共享，并在 DR 测试周期内重新分配给 DR 测试系统。这种方法可以减少 DR 测试系统硬件费用，但是冒着在运行 DR 测试时会影响生产系统性能的风险。但是，通常 DR 测试周期只将很小比例的共享资源专用于 DR 测试系统，缩小的生产环境不会受到并行 DR 测试很大的影响。然而，某些组织有针对更改或影响生产的策略来促进 DR 测试。审计员可能要求 DR 测试结果与生产结果精确匹配以证明 DR 恢复过程。满足这项要求的一种方法是，就在计划的生产运行前建立一个同步点、保存生产结果的副本、在 DR 测试站点恢复该同步点的生产运行，并将输出与保存的生产结果进行比较。结果之间的任何差异突出了必须调查的差距。如果未能及时弥补差距，会使组织的真正灾难恢复能力面临风险。无论 DR 测试设计成恢复复杂的工作负荷还是单个应用程序，都必须使用用于真正灾难恢复的步骤执行 DR 测试过程。这是证明 DR 测试成功的唯一一种可靠方法。

DR 测试的数据移动

在 DR 测试站点为 DR 测试暂存应用程序数据有两种方法：物理数据移动和电子数据移动。物理数据移动涉及在下述称为“物理导出/导入”的过程中将物理盒式磁带传送到 DR 测试站点。电子数据移动使用远程磁带机、远程 RTD 或 VSM 群集技术在 DR 测试站点创建应用程序数据副本。采用这两种数据移动方法均可进行 DR 测试，但是电子数据移动可避免物理数据传输以及与丢失磁带有关的任何潜在问题等等。电子传输通过将数据放在真正的灾难恢复需要的位置，或者通过在 DR 测试周期之前将

数据暂存在 VSM 缓冲区中，还能缩短访问数据的时间。虚拟卷的电子数据移动可以使用 VSM 扩展群集在单个 Tapeplex 中完成，也可以使用跨 Tapeplex 复制在两个 Tapeplex 之间完成。对于单个 Tapeplex 内的数据，Oracle 的 Concurrent Disaster Recovery Test (CDRT) 软件可简化 DR 测试。

通过物理导出/导入进行 DR 测试

假设您要为一个使用虚拟和物理磁带的生产应用程序执行 DR 测试。您的目标是通过重复最近的生产运行并验证测试输出是否与最近的生产输出匹配来在 DR 测试站点测试该应用程序。在准备过程中，您需要保存生产运行使用的任何输入数据集的副本以及生产输出的副本以便进行比较。假设 DR 测试站点已隔离，它与生产站点不共享任何设备。您可以采用以下物理导出/导入过程来执行 DR 测试。

生产站点：

1. 创建必需的 VTV 和物理卷的副本。
2. 导出这些 VTV 副本。
3. 从生产 ACS 中弹出关联的 MVC 副本和物理卷副本。
4. 将弹出的 MVC 和物理卷传送到 DR 测试站点。

DR 测试站点：

1. 将传送的卷装入 DR ACS。
2. 将 OS 目录和磁带管理系统与装入的卷同步。
3. 导入 VTV/MVC 数据。
4. 运行应用程序。
5. 比较结果。
6. 弹出为该测试装入的所有卷。
7. 将弹出的卷传送回生产站点。

生产站点：

1. 将传送的卷装回生产 ACS。

该过程允许 DR 测试安全地与生产并行进行，因为 DR 测试系统与生产系统隔离。DR 测试系统有自己的 CDS，在上述 DR 测试过程中，会将卷信息输入 DR 测试 CDS 以便为 DR 测试做好准备。这样允许恢复的应用程序以在生产系统中使用的卷和数据集名称进行测试。对于虚拟磁带数据集，Oracle 的 LCM 软件保管功能可简化将 VTV 放置到 MVC 上的过程，并简化上述往返步骤，即在生产站点导出和弹出卷、在 DR 测试站点导入这些卷以及弹出这些卷以便移回生产站点。物理导出/导入会使站点产生物理磁带处理费用以及在生产 and DR 测试站点之间传送盒式磁带的快递费用。由快递移动的敏感数据应该在加密的盒式磁带中传送。DR 测试时效性受在站点之间传送盒式磁带以及处理移动的盒式磁带所需时间的影响。

通过 CDRT 进行 DR 测试

经过规划并且在生产和 DR 站点配备足够的硬件后，CDRT 结合电子数据移动可避免将物理盒式磁带传送到 DR 站点，并使并发 DR 测试较之维护隔离的专用 DR 测试站点更经济。借助 CDRT，几乎可以对可想象的任何生产工作负荷、配置、RPO 或 RTO 进行 DR 测试。DR 测试过程将包括一些额外的步骤来启动 CDRT 以及在 DR 测试后进行清除。在借助 CDRT 运行 DR 测试之前，您应该以电子方式将测试所需的所有应用程序数据和系统元数据（OS 目录信息和磁带管理系统信息）移至 DR 测试站点。您可以使用 VSM 群集或通过将 VTV 副本迁移到 DR 站点的 MVC 来以电子方式移动应用程序数据。然后使用 CDRT 为 DR 测试系统创建特殊的 CDS 来镜像生产 CDS。生产和 DR 测试系统是单独的环境，DR 测试环境将使用特殊的 DR 测试 CDS 而非生产 CDS。由于 CDRT 根据生产 CDS 中的信息创建 DR 测试 CDS，因此它包含在 DR 测试之前以电子方式移至 DR 测试站点的所有卷的元数据。这样可使 DR 测试应用程序使用在生产系统中使用的卷序列号和磁带数据集名称。CDRT 在 DR 测试系统上实施操作限制，以防止 DR 环境干扰生产环境。您可以使用 ELS VOLPARM/POOLPARM 功能为 MVC 定义单独的卷序列号范围并暂存 VTV 使其由 CDRT 专用，从而加强这些保护。CDRT 允许 DR 测试系统从生产 MVC 中读取数据并将数据写入自己的将在每个 DR 测试周期过后以逻辑方式擦除的专用 MVC 池。对于虚拟磁带应用程序，CDRT 在 DR 测试周期内至少需要一个专用的 VTSS 设备。可以从生产系统临时重新分配这些专用的 VTSS 以促进 DR 测试，并且 DR 测试 VSM 系统可以与生产工作负荷并行访问生产 ACS。图 1.9 “远程群集节点 VTSS2 在 DR 测试站点的生产群集”和图 1.10 “借用 VTSS2 进行 CDRT DR 测试的生产配置”阐明了如何拆分生产 VSM 群集以将群集设备借给 CDRT DR 测试系统，在本例中是 DR 测试站点的 VTSS2。拆分该群集时，您必须更改生产策略以用迁移替换复制，这样 VTSS1 将在 DR 站点的 ACS01 中创建冗余 VTV 副本，并且拆分群集时 VTSS1 不会填充容量。VTSS2 将在生产系统中脱机而在 DR 测试 LPAR 中联机。在图 1.9 “远程群集节点 VTSS2 在 DR 测试站点的生产群集”中，CDRT 从生产 CDS 的远程副本创建了 DR 测试 CDS。只有生产系统可以在整个 DR 测试周期内访问 VTSS1 和 ACS00 中的卷，只有 DR 测试系统可以访问 VTSS2。生产和 DR 测试系统共享并发访问 ACS01 中的卷。图 1.9 “远程群集节点 VTSS2 在 DR 测试站点的生产群集”和图 1.10 “借用 VTSS2 进行 CDRT DR 测试的生产配置”在 DR 测试站点维护生产 CDS 的远程副本（例如通过远程镜像），以确保 DR 站点有最新的生产 CDS 可用于真正的灾难恢复。但请注意，由 CDRT 从远程 CDS 副本创建的 DR 测试 CDS 是生产 CDS 的特殊 DR 测试版本，只能由 CDRT 使用。在 DR 测试周期结束后重组生产群集之前，必须清除 DR VTSS 以避免生产数据丢失，因为如果 VTSS2 包含一个较新的 VTV 版本而该版本也存在于 VTSS1 中，将会发生这种情况。您还必须在重组群集后更改生产策略以从迁移恢复为复制。如果无法如此处所示拆分生产群集，替代方案是在 DR 站点专为 DR 测试维护一个单独的 VTSS。在这种情况下，将从 MVC 副本撤回测试所需的 VTV。

图 1.9. 远程群集节点 VTSS2 在 DR 测试站点的生产群集

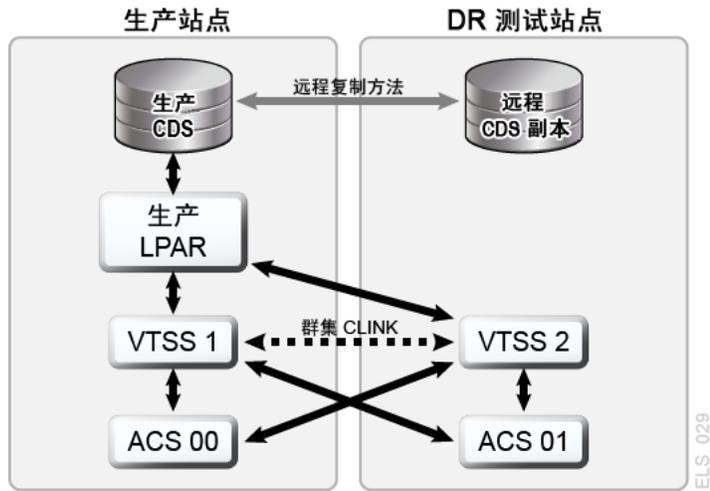
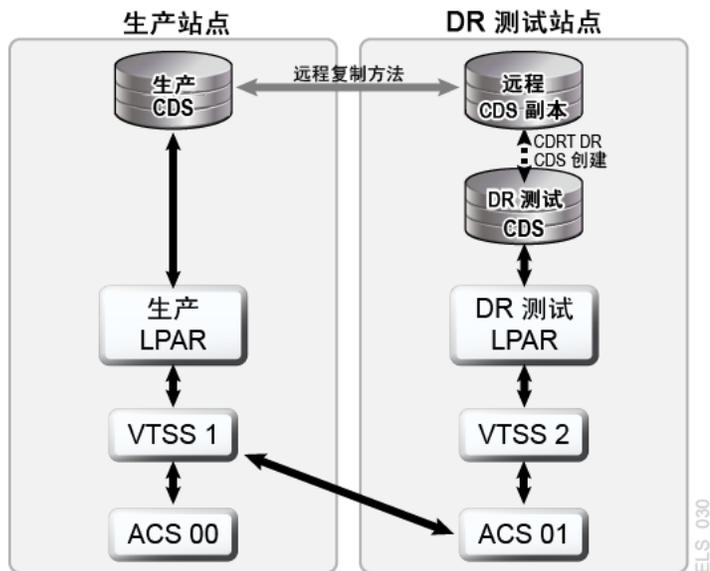


图 1.10. 借用 VTSS2 进行 CDRT DR 测试的生产配置

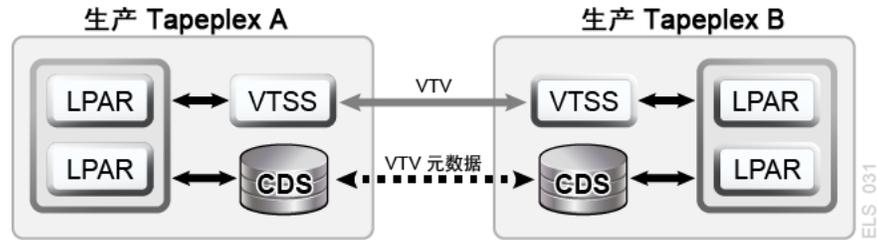


通过 VSM 跨 Tapeplex 复制进行 DR 测试

VSM 跨 Tapeplex 复制支持对称的群集生产 Tapeplex 设计，这些设计可促进 DR 测试而无需使用 CDRT、无需单纯为了 DR 测试的专用 VTSS 硬件并且无需为 DR 测试更改生产环境。例如，CTR 允许每个生产 Tapeplex 将数据复制到同一个 CTR 群集中的其他生产 Tapeplex。生产 CTR 对等群集可消除对专用 DR 测试站点的需要。CTR 支持多种不同类型的群集 TapePlex 设计并促进任何生产工作负荷或配置（具有任何可行的 RPO 或 RTO）的 DR 测试。举个简单的例子，一个双向 CTR 群集对称地将两个生产 TapePlex 相连，每个 TapePlex 将数据复制到另一个 TapePlex（图 1.11 “DR 测试的对称生产 CTR 群集”）。接收 TapePlex 以只读状态将复制的 VTV 装入其

CDS，并将 VTV 标记为由发送 TapePlex 拥有。在此示例中，TapePlex A 应用程序的 DR 测试涉及在 TapePlex B 上复制应用程序数据以及在 TapePlex B 上恢复应用程序。

图 1.11. DR 测试的对称生产 CTR 群集



这种对等 CTR 群集设计的对称性意味着，在对等站点进行测试的恢复的应用程序在 DR 测试期间的运行与在生产期间一样。对等 CDS 包含 DR 测试需要的所有复制的卷信息，DR 测试与生产并行进行，并且相同的 VTSS 硬件支持由生产和 DR 测试工作负荷并发使用。生产 VTSS 群集可能存在于每个 TapePlex 中，并且无需拆分即可在 TapePlex 之间共享硬件以便进行 DR 测试。从中执行应用程序 DR 测试的生产 TapePlex 无法修改任何 CTR 复制的 VTV，因此所有复制的生产数据在 DR 测试周期内完全受到保护。最重要的是，基于 CTR 的 DR 测试可保证经验证的 DR 测试过程在真正的灾难恢复期间将产生完全相同的结果。如果尝试更新一个 CTR 复制的 VTV，而它用来将应用程序标识为修改现有输入数据集的一个应用程序，则 SMC 主机软件将发出一条消息。按照上述管理同步点的最佳做法，您应该确保生产环境在应用程序修改该数据集之前保存该数据集的副本，以备同步点恢复需要备份副本。

第 2 章 执行物理导出和导入

借助 EXPORT 和 IMPORT 工具，可以创建能够以物理方式移植的 MVC。在源站点上，您使用 EXPORT 合并 MVC 上的 VTV（如有必要），并生成用于描述 MVC 内容（MVC 上的 VTV）的清单文件。随后从源站点弹出 MVC，以物理方式将其传送到目标站点，再对它们执行 IMPORT 操作，使用清单文件以有关所导入 MVC 和 VTV 的信息更新 CDS。请注意，您可以在 VTCS 未处于活动状态的情况下将 VTV 导入 CDS。然后将 MVC 装入目标站点。

注：

- 如果决定将导出的 MVC 返回到源系统，无需任何特殊的 VTCS 处理，只需将 MVC 装入源系统的 LSM 中即可。
- 对于导入的每个 VTV，唯一将要创建的 MVC 副本针对通过相同的语句导出和导入的 MVC。这在导入双工 VTV 时有特殊的意义。仅当两个 MVC 出现在同一个清单文件中并且由于使用了相同的 IMPORT 语句而导入时，导入后两个 MVC 上才会有此类 VTV 的副本。

可以通过以下任一常规方法导出：

- 按 VTV 或管理类导出，将选定的 VTV 合并到一组新的 MVC 上。由于合并需要时间并且需要 VTSS 资源，因此首选方法是按 MVC 或存储类执行导出。有关更多信息，请参见[“按管理类导出和导入”](#)。
- 按 MVC 或存储类导出。按存储类或 MVC 导出无需对 VTV 进行任何合并后处理，并且无需移动数据。导出只会创建用于描述选定 MVC 内容的清单文件。有关更多信息，请参见[“按存储类导出和导入”](#)。

注：

如果按如下方式导出：

- **VTV** 卷序列号—使用 TMS、LCM 或 VTVRPT 报告确定所需的 VTV。
- **MVC** 卷序列号—使用 LCM 或 MVC RPT 报告确定所需的 MVC。
- **管理类**—查看管理类定义以确定所需的管理类。
- **存储类**—查看存储类定义以确定所需的存储类。

按管理类导出和导入

以下示例显示了按管理类导出和导入 MVC。

注:

按 MGMTclas 或 VTV 完成导出时, CDS 不在 VTV 记录中保留 MVC 信息。在这种情况下, MVC 将标记为 "(E)" (导出)。

示例: 按管理类从源 VSM 系统导出

这是导出/导入的“发送”阶段, 在这一阶段中, 您将所需数据打包并移出源 VSM 系统。

要从源 VSM 系统导出, 请执行以下操作:

1. 确定用于导出的管理类。
2. 按管理类导出:

```
//EXPORT EXEC PGM=SLUADMIN, PARM=' MIXED' REGION=6M
//STEPLIB DD DSN=h1q.SEALINK, DISP=SHR
//MOVE1 DD DSN=h1q.REMOTE2, DISP=(, CATLG, DELETE),
// UNIT=SYSDA, SPACE=(CYL, (1, 1), RLSE),
// DCB=(RECFM=FB, LRECL=80, BLKSIZE=27920)
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
EXPORT MGMT (PAY, ACCOUNT) MANIFEST(MOVE1)
```

在此示例中, 输出清单文件为 MOVE1, 导入时需要该文件。由于是按管理类导出, 因此 EXPORT 将在导出 MVC 上合并选定的 VTV (为其创建副本)。导出 MVC 在 CDS 中标记为只读且标记为已导出, 现在可从源系统 LSM 中弹出。这些合并的 VTV 副本是额外的副本, 不会记录在 CDS 中。例如, 如果 VTV 在导出之前是双工, 则 CDS 将记录两个双工副本, 但是不会在 CDS 中记录用于合并的第三个额外副本。因此, 源系统仍可使用原始 VTV。您可以使用原始 VTV 上的数据或者暂存并重用它们。

注意:

计划当未在更新导出的数据时执行导出!

3. 从 MVC 池中移除要导出的 MVC。有关更多信息, 请参见管理 HSC 和 VTCS。
4. 从源 VSM 系统 LSM 中弹出要导出的 MVC。有关更多信息, 请参见管理 HSC 和 VTCS。
5. 如果需要, 在源系统中暂存导出的 VTV 或使其不可用亦或重用其中包含的数据。

导出后, 源系统保留已导出 VTV 和 MVC 的 CDS 记录。导出 MVC 在源系统 CDS 中标记为已导出且标记为只读。此时, 根据导出 VTV 的原因, 您有两种选择:

- 如果导出 VTV 是为了在第二个站点上提供备份副本, 请使 VTV 在源系统 CDS 中保持只读状态, 以使它们无法更新。

- 如果要将导出的 VTV 永久移至第二个站点，则在源系统 CDS 中暂存或使其不可用。使用 HSC 暂存实用程序或 LCM SYNCVTV 功能暂存导出的 VTV。

示例：按管理类导入目标 VSM 系统

一个月过后，您终于为导出/导入操作的“接收”（导入）部分做好了准备。

要导入目标 VSM 系统，请执行以下操作：

1. 如果要导入的 VTV 和 MVC 未在目标系统 CDS 中，请重做 *POOLPARAM/VOLPARAM* 定义以添加这些卷序列号，如配置 *HSC* 和 *VTCS* 中所述。

如有必要，增加目标 VSM 系统上的 CDS 大小。有关更多信息，请参见配置 *HSC* 和 *VTCS* 或管理 *HSC* 和 *VTCS*。

如果源系统和目标系统上存在重复的 VTV 卷序列号，应该怎样做呢？一般来说，执行以下操作：

- 如果源系统上存在较新的 VTV 且其卷序列号与目标系统上的相同，请指定 *REPLACE(ALL)*。
- 如果要将 VTV 从源系统移至目标系统（首次导出/导入），请指定 *REPLACE(NONE)*。在这种情况下，您必须根据具体情况确定如何处理重复的 VTV。

2. 将要导入的 MVC 装入目标 VSM 系统 LSM。

有关更多信息，请参见管理 *HSC* 和 *VTCS*。在使用 *IMPORT* 告知 CDS 具有一些新的 MVC 和 VTV 之前，您要以物理方式将 MVC 部署到位。

3. （可选）执行 *IMPORT* 的“验证”运行：

```
//IMPORT EXEC PGM=SLUADMIN, PARM='MIXED' REGION=6M
//STEPLIB DD DSN=h1q.SEALINK, DISP=SHR
//REMOTE1 DD DSN=h1q.REMOTE1, DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
      IMPORT MANIFEST(MOVE1) NOUPDATE
```

此处显示了运行 *IMPORT* 实用程序的示例 JCL，其中：

- 清单文件是在第 2 步中指定的导出清单。
- *REPLACE(NONE)*（默认值）指定 *VTCS* 不会覆写重复的 VTV。
- *IMMDRAIN(NO)*（默认值）指定 *VTCS* 不会将所有导入的 VTV 排空到 *VTSS* 空间。
- *NOUPDATE* 指定不会更新 CDS（仅限于验证运行）。
- 未指定 *INACTCDS*，所以 *HSC* 处于活动状态。

执行验证运行是可选操作，但是强烈建议执行，因为您确实想要在按按钮之前看看将会发生什么。仔细研究导入报告。与您看到的一样？继续执行第 4 步。

注：

- 仅当指定了 *FEATures VSM(ADVMGMT)* 时 *IMPORT* 才有效。
- 确保“目标”CDS 与“源”CDS 具有相同的功能（在 CDS 级别启用）。例如，如果“源”CDS 启用了大型 VTV 页面大小并且已创建 2/4 Gb VTV，则“目标”CDS 必须具有相同的功能，否则导入将会失败。

4. 执行 *IMPORT* 的实际运行：

```
//IMPORT EXEC PGM=SLUADMIN, PARM=' MIXED' REGION=6M
//STEPLIB DD DSN=h1q.SEALINK, DISP=SHR
//REMOTE1 DD DSN=h1q.REMOTE1, DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
IMPORT MANIFEST(MOVE1) REPLACE(ALL)
```

此处显示了运行 *IMPORT* 实用程序的示例 JCL，其中，与在“验证”运行中一样，*REPLACE(NONE)*（默认值）指定 VTCS 不会覆写重复的 VTV。

注：

如果您要将 MVC 返回到源系统，应该怎样做呢？如果是这样，您可以指定 *IMMDRAIN(YES)* 以排空导入 MVC。

5. 根据需要调整 VTV 定义。例如，您需要为目标系统的 TMS 定义任何新的 VTV。

6. 执行以下操作之一：

- （可选）运行 *MVCMaint* 以使导入的 MVC 可写。VTCS 以只读方式导入 MVC。要使其可写，您可以运行 *MVCMaint*，指定 *READONLY OFF*。您很可能希望在目标系统上使用新的 MVC，而这是第一步。

接下来，将导入的 MVC 添加到 MVC 池中，如管理 *HSC* 和 *VTCS* 中所述。此时，可以对 MVC 进行回收、排空、迁移、撤回等等。

- 如果在第 4 步中指定了 *IMMDRAIN(YES)*，则可将 MVC 返回到源系统。

按存储类导出和导入

以下示例显示了按存储类从源 VSM 导出和导入。

注：

按 *STORclas* 或 MVC 完成导出时，CDS 在 VTV 记录中保留 MVC 信息。在这种情况下，MVC 将标记为“(R)”（只读）。

示例：按存储类从源 VSM 系统导出

这是导出/导入的“发送”阶段，在这一阶段中，您将所需数据打包并移出源 VSM 系统。

要从源 VSM 系统导出，请执行以下操作：

1. 确定用于导出的存储类。
2. 按存储类导出：

```
//EXPORT EXEC PGM=SLUADMIN, PARM='MIXED' REGION=6M
//STEPLIB DD DSN=h1q.SEALINK, DISP=SHR
//MOVE2 DD DSN=h1q.REMOTE2, DISP=(, CATLG, DELETE),
// UNIT=SYSDA, SPACE=(CYL, (1, 1), RLSE),
// DCB=(RECFM=FB, LRECL=80, BLKSIZE=27920)
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
EXPORT STOR(OFF1, OFF2) MANIFEST(MOVE2)
```

在此示例中，输出清单文件为 *MOVE2*，导入时需要该文件。由于是按存储类导出，因此系统将创建清单文件，但是不会发生 VTV 合并。导出 MVC 在 CDS 中标记为只读，现在可从源系统 LSM 中弹出。之前位于从 LSM 中移除的 MVC 上的 VTV 仍可使用，前提是它们现在位于其他 MVC 上。

注意：

计划当未在更新导出的数据时执行导出！

3. 从 MVC 池中移除要导出的 MVC。有关更多信息，请参见管理 *HSC* 和 *VTCS*。
4. 从源 VSM 系统 LSM 中弹出要导出的 MVC。有关更多信息，请参见管理 *HSC* 和 *VTCS*。
5. 如果需要，在源系统中暂存导出的 VTV 或使其不可用亦或重用其中包含的数据。

导出后，源系统保留已导出 VTV 和 MVC 的 CDS 记录。导出 MVC 在源系统 CDS 中标记为已导出且标记为只读。此时，根据导出 VTV 的原因，您有两种选择：

- 如果导出 VTV 是为了在第二个站点上提供备份副本，请使 VTV 在源系统 CDS 中保持只读状态，以使它们无法更新。
- 如果要将导出的 VTV 永久移至第二个站点，则在源系统 CDS 中暂存或使其不可用。使用 HSC 暂存实用程序或 LCM SYNCVTV 功能暂存导出的 VTV。

示例：按存储类导入目标 VSM 系统

一个月过后，您终于为导出/导入操作的“接收”（导入）部分做好了准备。

要导入目标 VSM 系统，请执行以下操作：

1. 如果要导入的 VTV 和 MVC 未在目标系统 CDS 中，请重做 *POOLPARM/VOLPARM* 定义以添加这些卷序列号，如配置 *HSC* 和 *VTCS* 中所述。

如有必要，也要增加目标 VSM 系统上的 CDS 大小。有关更多信息，请参见“配置 *HSC* 和 *VTCS*”或管理 *HSC* 和 *VTCS*。

如果源系统和目标系统上存在重复的 VTV 卷序列号，应该怎样做呢？一般来说，执行以下操作：

- 如果源系统上存在较新的 VTV 且其卷序列号与目标系统上的相同，请指定 *REPLACE(ALL)*。
- 如果要将在源系统移至目标系统（首次导出/导入），请指定 *REPLACE(NONE)*。在这种情况下，您必须根据具体情况确定如何处理重复的 VTV。

2. 将要导入的 MVC 装入目标 VSM 系统 LSM。

有关更多信息，请参见管理 *HSC* 和 *VTCS*。您能看到这里发生了什么吗？在使用 *IMPORT* 告知 CDS 具有一些新的 MVC 和 VTV 之前，您实际上要以物理方式将 MVC 部署到位。

3. （可选）执行 *IMPORT* 的“验证”运行。

```
//IMPORT EXEC PGM=SLUADMIN, PARM=' MIXED' REGION=6M
//STEPLIB DD DSN=h1q.SEALINK, DISP=SHR
//REMOTE1 DD DSN=h1q.REMOTE1, DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
      IMPORT MANIFEST(REMOTE1) NOUPDATE
```

此处显示了运行 *IMPORT* 实用程序的示例 JCL，其中：

- 清单文件是在第 2 步中指定的导出清单。
- *REPLACE(NONE)*（默认值）指定 *VTCS* 不会覆写重复的 VTV。
- *IMMDRAIN(NO)*（默认值）指定 *VTCS* 不会将所有导入的 VTV 排空到 *VTSS* 空间。
- *NOUPDATE* 指定不会更新 CDS（仅限于验证运行）。
- 未指定 *INACTCDS*，所以 *HSC* 处于活动状态。

执行验证运行是可选操作，但是强烈建议执行，因为您确实想要在实际按按钮之前看看将会发生什么。仔细研究导入报告。与您看到的一样？继续执行第 4 步。

注：

- 仅当指定了 *FEATures VSM(ADVMGMT)* 时 *IMPORT* 才有效。
 - 确保“目标”CDS 与“源”CDS 具有相同的功能（在 CDS 级别启用）。例如，如果“源”CDS 启用了大型 VTV 页面大小并且已创建 2/4 Gb VTV，则“目标”CDS 必须具有相同的功能，否则导入将会失败。
-

4. 执行 *IMPORT* 的实际运行:

```
//IMPORT EXEC PGM=SLUADMIN, PARM='MIXED' REGION=6M
//STEPLIB DD DSN=h1q.SEALINK, DISP=SHR
//REMOTE1 DD DSN=h1q.REMOTE1, DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
      IMPORT MANIFEST(REMOTE1)
```

此处显示了运行 *IMPORT* 实用程序的示例 JCL，其中，与在“验证”运行中一样，*REPLACE(NONE)*（默认值）指定 VTCS 不会覆写重复的 VTV。

注:

如果您要将 MVC 返回到源系统，应该怎样做呢？如果是这样，您可以指定 *IMMDRAIN(YES)* 以排空导入 MVC。

5. 根据需要调整 VTV 定义。

6. 执行以下操作之一:

- （可选）运行 *MVCMAINT* 以使导入的 MVC 可写。VTCS 以只读方式导入 MVC。要使其可写，您可以运行 *MVCMAINT*，指定 *READONLY OFF*。您很可能希望在目标系统上使用新的 MVC，而这是第一步。

接下来，将导入的 MVC 添加到 MVC 池中，如管理 *HSC* 和 *VTCS* 中所述。此时，可以对 MVC 进行回收、排空、迁移、撤回等等。

- 如果在第 4 步中指定了 *IMMDRAIN(YES)*，则可将 MVC 返回到源系统。

VLE 导出/导入

可以将 VTV 和 MVC 导出到在两个 tapeplex 之间共享的 VLE，但前提是这两个 tapeplex 具有不同的 MVC 池。

要导出到与两个 tapeplex 相连的 VLE，请以下面的内容为例：

1. 假定 Tapeplex A 具有的 MVCPOOL 包含 A00-A10，Tapeplex B 具有的 MVCPOOL 包含 B00-B10。
2. 将 A00-A10 导出到 VLE Tapeplex A MVC 池。
3. 使用 *IMPORT REPLACE(UPDATED)* 命令将 A00-A10 MVC 池导入到 Tapeplex B。

第 3 章 使用 ELS 外部保管功能

ELS 外部保管功能（ELS 保管库）取代并显著改善了其前代产品“VSM 异地保管库功能”。ELS 保管库为保管实际磁带卷提供了以下增强功能：

- 使用 HSC CDS 存储保管库和保管的卷数据。使用 CDS 保管信息（而不是 TMS）可消除：
 - 将卷返回至自动化环境时出现人为错误的风险
 - 卷在返回的拉式列表中缺失时滞留在保管库中
 - 意外将返回的保管库卷遗留在自动化环境中
- 使用 LCM 可管理保管过程，有三种保管方法：
 - 保管用于灾难恢复 (Disaster Recovery, DR) 的 MVC；有关更多信息，请参见[“保管用于 DR 的 MVC”](#)。
 - 保管用于长期保留 (Long Term Retention, LTR) 的 MVC；有关更多信息，请参见[“保管用于 LTR 的 MVC”](#)。
 - 将卷弹出至基础保管库；有关更多信息，请参见[“将特定的卷弹出至本地（基础）保管库”](#)。

准备进行 ELS 外部保管

第一步是定义 HSC CDS 的保管卷区域。为此，您需运行 `SLUADMIN SET VAULTVOL` 实用程序。例如：

```
SET VAULTVOL NBRVOLS(40000)
```

注：

- 如果初始定义之后您需要添加更多的保管库卷，则需使用 `MERGEcds` 来完成，从而允许在初始定义中具有足够多的卷来满足您的预期需求。
- 您在 CDS 中必须具有足够的空闲块，才可容纳您计划要保管的卷（实际磁带卷）以及某些增长开销。有关计算保管库卷空间的信息，请参见配置 `HSC` 和 `VTCS`；有关在 CDS 不足以容纳保管卷时如何扩展 CDS 的信息，请参见管理 `HSC` 和 `VTCS`。

第二步是定义用来容纳保管卷的保管库。为此，您需针对每个保管库运行 `SLUADMIN SET VAULT` 实用程序。例如：

```
SET VAULT ADD NAME(DRVLT1) SLOTS(10000) DESC('DR Vault')
```

```
SET VAULT ADD NAME(LTRVLT1) SLOTS(20000) DESC('LTR Vault')
SET VAULT ADD NAME(FLOOR) SLOTS(500) DESC('Floor Vault')
```

注:

您定义的所有保管库中的插槽总数不能超出 *VAULTVOL* 语句中指定的卷数。

HSC 负责定义保管库和保管卷，LCM 则负责加以管理。特别地，请注意以下 LCM 保管参数：

GRACEPERIOD

选择从保管库中返回的卷时与该卷实际返回至自动化环境时之间的天数。宽限期提供了一个安全边界，使新卷能够在旧卷返回之前到达保管库中。如果未指定，则默认值为三天。

DEFAULT

DEFAULT 与 *GRACEPERIOD* 相互排斥，通常用于“基础”（手动机架）保管库。该保管库通过使用 *LCM EJECT(ASNEEDED)* 自动包含从自动化环境中弹出的所有卷。其他从 ACS 弹出的卷也可分配给此保管库，例如，某个处于活动状态、但不再适合自动化的卷。通常，这是指实际位于数据中心基底上的机架的既定保管库。*DEFAULT* 是一个时间为零天的宽限期，允许这些卷随时重新装入 ACS。

另请注意，标准弹出选项适用于所有保管库卷。这包括选择要使用的 CAP、定义弹出消息、选择弹出模式，以及确定弹出是按插槽还是卷序列顺序进行。

为 DR 和 LTR 创建 MVC

无论何时，只要针对 DR 保管 MVC，均应为每个 VTV 至少创建两个副本来分隔 MVC。其中一个 MVC 保留在现场，另一个 MVC 则会弹出并放在外部保管库中。这可通过在管理类（已分配给 VTV）中分配两个存储类来完成。

因此，难题在于：您既想保护数据的安全，但又想尽量按照如下所示以经济实惠的方式使用 MVC 空间：

- 尽量定义较少的保管类。过多的存储类通常意味着过多的 MVC 和/或过多仅含少数 VTV 的 MVC。
- 尽量使用较少的 VTSS 创建 MVC。如有可能，请仅使用一个 VTSS 创建保管库 MVC。

创建和保管 MVC 时还应考虑哪些事项？考虑以下事项：

- 首先，VTV 需尽快迁移至保管库 MVC，以便能够弹出并移至保管库，这是因为这些 VTV 通常不会用作其他作业步骤的输入）。
- 第二，由于 DR VTV 将先后失效，因此，请考虑将 MVC 中具有类似失效日期的 VTV 进行分组。但是，若要减少创建以发送至保管库的 MVC 总数，请限制这些组的数量。由于系统中已有活动会将 VTV 合并至较少的 MVC 中，因此，超过两个组（一个组用于 VTV，失效期间较短，例如未来 7 天内失效；另一个组用于所有其他

卷)并不会带来更多益处。由于无法知道实际的失效日期,因此目录控制下的 VTV 应视为第二组的一部分。

- 第三,尽管不必出于 DR 和 LTR 目的分别设立单独的保管库,但为 DR MVC 设立单独的保管库可能很有益。这允许收集这些卷并根据需要发送至 DR 站点。

对于 LTR 数据,考虑事项略有不同。首先,定义的 LTR 数据在延长期间不会失效。因此,与那些随着时间推移失效的 DR 数据不同,LTR MVC 不会碎片化。这些 MVC 将定期进行初始处理,使保管的 MVC 尽量填满,正如 DR MVC 一样。但一旦被填满,这些 MVC 将保持静态。这些卷无需多个存储类。但请注意,您可能需要立即迁移某些 LTR 数据,其他数据可在 VTCS 自动迁移选择时迁移。因此,您可能需要为 LTR 数据设置一个存储类,但设置两个管理类。

使用保管功能时的 DELSCR 考虑事项

您可以使用 *MGMTclas* 语句的 *DELSCR* 参数指定 VSM 是否删除已暂存的 VTV,其中 *DELSCR(YES)* 将导致 VSM 删除已暂存的 VTV,从而释放 VTSS 缓冲区空间和 MVC 空间。请考虑为 DR 和 LTR 管理类指定 *DELSCR(YES)*。如果指定 *DELSCR(YES)*,则仅将 LCM *SYNCVTV* 用于暂存同步。有关使用 LCM 管理暂存同步的更多信息,请参见《*LCM User's Guide*》。

保管卷返回到 ACS 时会发生什么?

HSC 装入过程已针对 ELS 保管功能进行修改,以便检查装入的每个卷,从而确定该卷是否属于外部保管卷。对于此类保管卷,根据 CDS 保管库记录中的返回日期字段,将发生以下两项操作之一:

- 如果返回日期出现,则系统将装入卷,且恢复弹出过程中存储的卷元数据,并从保管库记录中删除该卷。
- 如果返回日期未出现,或未对卷设置返回日期,则系统将装入卷,恢复弹出过程中存储的卷元数据,但该卷将留在保管库记录中,并由下一弹出过程自动弹出。为何会出现此操作?原因有多个,但最常见的两个原因如下:为了某种数据恢复过程而返回卷;或者从保管库中拉出的卷不正确(这是一种极为常见的现象)。无论何种原因,该卷均属于保管库,且会返回至该位置并恢复其受保护状态。

请注意,这是保管在物理保管库中的卷的过程。对于保管在远程磁带库中的卷,上述过程略有不同。请参见[“远程磁带库中的含 MVC 的 DR 保管”](#)。

保管用于 DR 的 MVC

在 DR 方案中,您的整体业务目标是优化 VTSS 的缓冲区使用,确保在快速迁移关键数据的同时,维持数据的可用性。

含 MVC 的基本 DR 保管

在此方法中,系统每天均会创建 DR 卷,因此每天均会运行将其移至 DR 保管库的处理,确保 MVC 能够异地安全移动且受到保护。

注:

- 所有出于 DR 目的而创建的 VTV 以及所有本机磁带（包括在“[第 2 步—导出保管库 MVC](#)”中创建的清单文件磁带）均由站点的 TMS 控制（用于卷失效）。该过程仅限于保管相关的 MVC 以及所选的相关本机卷。
- MVC 只能分配给一个保管库。要将卷分配给新保管库，则必须先从上一保管库分配中删除该卷。
- “[第 7 步—准备要返回的保管 MVC](#)”将开始进行定期处理，允许回收那些已碎片化或创建时仅部分填充的 DR MVC。这一点将通过使用本地 MVC 中仍属于当前 VTV 的副本对保管 MVC 执行“逻辑”MVC 排空来实现。定期处理通过使用适于特定环境的选择标准，在最大程度地减少保管库中卷总数的同时，确保最大程度地减少整体的相关活动。成功执行“逻辑”排空保管的 MVC 之后，CDS 中将设置该卷的返回日期。如果是本机卷（例如清单文件磁带），则在 TMS 中进入暂存状态的卷将会被选中，并设置为待返回。您需决定多长时间（从每天到每月不等）运行一次定期处理。

第 1 步—创建保管库 VTV/MVC

DR 保管库 VTV 是使用一种指向两个存储类的管理类创建的。其中，一个存储类将创建保留在本地环境内的 MVC，另一个存储类则创建已保管的 MVC。例如：

```
STOR NAME(DRLOC)ACS(00) MEDIA(STK1RD)
STOR NAME(DRVLT1) ACS(00) MEDIA(STK1RD)
```

第 2 步—导出保管库 MVC

您通过 LCM 参数文件导出保管库 MVC，如以下示例所示。

```
Options
  NoSync
  NoTMS
;
Vault
  Name('DRVLT')
  NoSync
  GracePeriod(3)
;
Action
  Export
  Control(Serial )
  MVC
  DSN(DRVAULT.MANIFEST)
  Storageclass(DRVLT1)
  Vault('DRVLT')
;
```

在此示例中：

- *OPTIONS* 语句将指定 *NOSYNC* 和 *NOTMS*，这是因为保管功能不使用 TMS 信息进行保管，且无需 TMS 元数据。
- *VAULT* 语句将 *DRVLT* 指定为 DR 保管库。
- *ACTION EXPORT* 语句指定：

- 按卷序列号导出 MVC。
- 创建导出清单文件 (*DRVAULT.MANIFEST*)，在本例中，是指 ACS 中使用 DR MVC 弹出和存储的卷。
- 指向“[第 1 步—创建保管库 VTV/MVC](#)”中创建的保管库存储类。
- 定义 DR 保管库 (*DRVLT*)，并将 MVC 分配给该保管库（如果以前尚未分配）。

注：

导出的 MVC 由导出处理标记为只读。

您可以创建多个保管库存储类（例如，为了将 VTV 具有不同失效日期的 MVC 进行隔离）。如果需要将不同的保管库存储类分配给相同的保管库，则可使用单一 *ACTION EXPORT* 语句实现。例如，以下语句可将存储类 *DRVLT1* 和 *DRVLT2* 分配给相同的保管库 (*DRVLT*)：

```
Action
  Export
  Control(Serial )
  MVC
  DSN(DRVAULT.MANIFEST)
  Storageclass(DRVLT1
               DRVLT2)
  Vault('DRVLT')
;
```

第 3 步—（可选）将其他数据集写入清单文件磁带

在“[第 2 步—导出保管库 MVC](#)”中创建清单文件磁带之后，您可以运行一项作业，将 HSC 控制数据集 (Control Data Set, CDS)、TMS 目录、系统目录以及其他重要的“时间点”数据集复制到清单文件磁带，以提供其他 DR 恢复点。

第 4 步—弹出保管库 MVC

您使用 LCM 参数文件弹出保管库 MVC，如以下示例所示。

```
Options
  NoSync
  NoTMS
;
Vault
  Name('DRVLT')
  GracePeriod(3)
;
Action
  Eject
  When(
    (inLsm)
    and
    (VaultName EQ 'DRVLT')
  )
  Control(Serial)
  Ejmsg('Move to DR Vault')
;
```

在此示例中：

- *OPTIONS* 语句将指定 *NOSYNC* 和 *NOTMS*，这是因为保管功能不使用 TMS 信息进行保管，且无需 TMS 元数据。
- *VAULT* 语句将 *DRVLT* 指定为 DR 保管库。
- *ACTION EJECT* 语句指定：
 - 弹出分配给 *DRVLT* 的 MVC；按卷序列号弹出。
 - 弹出消息。

第 5 步—弹出本机卷（包括清单文件磁带）

您使用 *LCM* 参数文件弹出本机卷（包括清单文件磁带），如以下示例所示。

```
Options
  NoSync
  ;
TMS
  RMM
  Dateform(J)
  DDname(LCMTMSDB)
  ;
Vault
  Name('DRVLT')
  GracePeriod(3)
  ;
Action
  Eject
  When(
    (InLsm)
    and
    (DataSetName EQ 'DRVLT.MANIFEST')
    and
    (TMSScratch EQ False)
  )
  Control(Serial)
  Ejmsg('Move to DR Vault')
  ;
```

在此示例中：

- *OPTIONS* 语句指定了 *NOSYNC*，这是因为保管功能不使用 TMS 信息进行保管。
- *VAULT* 语句将 *DRVLT* 指定为 DR 保管库。
- *ACTION EJECT* 语句指定：
 - 弹出清单文件磁带。
 - 弹出 TMS 未暂存的任何本机卷。
 - 弹出消息。

第 6 步—创建要从保管库中返回的卷的拉式列表

您使用 *LCM* 参数文件创建拉式列表，如以下示例所示。

```

Options
  NoSync
  NoTMS
;
Vault
  Name('DRVLT')
  GracePeriod(3)
;
Report
  Volume
  Sysout(*)
  Title('Return Report')
  When(
    (VaultName EQ 'DRVLT')
    and
    (VaultReturnDate LE TODAY)
    and
    (VaultReturnDate NE MISSING)
  )
  Column (Serial,
          VaultSlot)
;

```

在此示例中：

- *OPTIONS* 语句将指定 *NOSYNC* 和 *NOTMS*，这是因为保管功能不使用 TMS 信息进行保管，且无需 TMS 元数据。
- *VAULT* 语句将 *DRVLT* 指定为 DR 保管库。
- *REPORT VOLUME* 语句创建一份报告，列出保管库中已达到其先前指定的返回日期的卷。这是一个简单的示例，您可以为要返回的卷添加更多选择标准。

注：

- 在考虑日期时，*TODAY* 和 *MISSING* 是唯一值。*TODAY* 将转换为 LCM 的运行日期。*MISSING* 意味着没有日期值。在此示例中，意味着没有设置任何日期。这两个条件均需指定，因为缺少的日期将视为小于当前日期。
 - 第 4、5 和 6 步可以合并成一个作业步骤。在某些情况下，第 6 步将定期执行，通常在从保管库中返回任何卷之前的一天执行。
-

第 7 步 — 准备要返回的保管 MVC

您使用 *LCM* 参数文件准备要返回的保管 MVC，如以下示例所示。

```

Options
  NoSync
  NoTMS
;
Vault
  Name('DRVLT')
  GracePeriod(3)
;
Action
  Drain
  When(

```

```

(MVC EQ True)
and
(VaultName EQ 'DRVLT')
and
(MVCVTVCount LE 30)
and
(MVCInUse LE 30)
and
(Days_Since(VaultAssignmentDate) GT 7)
)
Control(MVCVTVCount
Ascending)
Limit(30)
;

```

在此示例中：

- *OPTIONS* 语句将指定 *NOSYNC* 和 *NOTMS*，这是因为保管功能不使用 TMS 信息进行保管，且无需 TMS 元数据。
- *VAULT* 语句将 *DRVLT* 指定为 DR 保管库。
- 对于目前位于 DR 保管库中的 MVC，*ACTION DRAIN* 语句将指定符合以下条件的排空 MVC：
 - 其 VTV 少于 30 个。
 - 使用时间不足 30%。
 - 在保管库中至少已有 7 天。
 - 返回的 MVC 最大数量限制为 30。
 - *GracePeriod* 参数将返回日期设置为 3 天。

创建参数文件以排空 MVC 时，您需要平衡排空的处理周期以及回收碎片 MVC 或回收部分填充 MVC 的需求。相应地，*MVCVTVCount*、*MVCInUse*、*Days_Since* 和 *LIMIT* 参数有助于平衡这些需求。

第 8 步 — 准备要返回的保管本机卷

您使用 *LCM* 参数文件准备要返回的本机卷，如以下示例所示。

```

Options
  NoSync
  ;
TMS
  RMM
  Dateform(J)
  DDname(LCMTMSDB)
  ;
Vault
  Name('DRVLT')
  GracePeriod(3)
  ;
Action
  Vault
  Return

```

```

When(
  Not (MVC)
  and
  (VaultName EQ 'DRVLT')
  and
  (TMSScratch EQ True)
)
;

```

在此示例中：

- *OPTIONS* 语句指定了 *NOSYNC*，这是因为保管功能不使用 TMS 信息进行保管。
- *TMS RMM* 语句对于添加 TMS 元数据的处理必不可少。
- *VAULT* 语句将 *DRVLT* 指定为 DR 保管库。
- *ACTION VAULT RETURN* 语句将为不是 MVC 的卷以及在 TMS 中处于暂存状态的卷设置返回日期（使用 *GracePeriod* 参数）。

第 9 步—装入返回的卷

报告（“[第 6 步—创建要从保管库中返回的卷的拉式列表](#)”中创建）中显示的卷将从 DR 保管库中删除并返回本地环境。在 ACS 中装入这些卷时，HSC 将检查每个卷的指定的保管库返回日期是否已出现。如果出现，则该卷将达到计划的返回日期，并将从保管库中删除。稍后，返回的 MVC 可进行迁移，且下次处理 LCM SYNC 时，本机卷将暂存在 CDS 中。如果保管库返回日期未出现，则该卷通过“[第 4 步—弹出保管库 MVC](#)”弹出。

如果需要，您可以将第 7 步和第 8 步合并到一个 LCM 参数文件，如以下示例所示：

```

Options
  NoSync
;
TMS
  RMM
  Dateform(J)
  DDname(LCMTMSDB)
Vault
  Name('DRVLT')
  GracePeriod(3)
;
Action
  Drain
  When(
    (MVC EQ True)
    and
    (VaultName EQ 'DRVLT')
    and
    (MVCVTVCount LE 30)
    and
    (MVCInUse LE 30)
    and
    (Days_Since(VaultAssignmentDate) GT 7)
  )
  Control(MVCVTVCount

```

```

Ascending)
Limit(30)
;
Action
Vault
Return
When(
Not (MVC)
and
(VaultName EQ 'DRVLT')
and
(TMSScratch EQ True)
)
;

```

长达数周的含 MVC 的 DR 保管

某些站点可能会选择使用长达数周的过程，其中，DR 处理包括关键数据每周进行一次完全卷备份，并在接下来的六天里每天增量备份。外部保管过程将在创建当日异地移动卷。该过程假设以四周为一个周期，它将 DR 数据异地保存，直到第四周开始时完成整个过程。请注意，“含 MVC 的基本 DR 保管”和数周过程之间的唯一变化是第 7 步和第 8 步的选择标准不同，如下所述。这可通过设置失效日期来完成，使关联 MVC 和清单文件磁带（以及其他任何相关的本机磁带）中的保管库 VTV 在创建后的特定日期（22 天）全部失效。

数周过程的时间线如下所示：

- 第 1 天—完全卷备份（在第 22 天失效）。
- 第 2 天—第一次增量备份（在第 22 天失效）。
- 第 3 天—第二次增量备份（在第 22 天失效）。
- 第 4 天—第三次增量备份（在第 22 天失效）。
- 第 5 天—第四次增量备份（在第 22 天失效）。
- 第 6 天—第五次增量备份（在第 22 天失效）。
- 第 7 天—第六次增量备份（在第 22 天失效）。
- 第 8-21 天—卷保留在异地。
- 第 22 天—第 1 天至第 7 天的备份和清单文件磁带失效，且 VTV 通过 LCM VTVSYNC 进程暂存在 CDS 中。通过使用“[第 7 步—准备要返回的保管 MVC](#)”和“[第 8 步—准备要返回的保管本机卷](#)”的选择标准中的以下参数，可以排空 MVC。

```
DAYS SINCE (VaultAssignmentDate) GT 15
```

已排空的 MVC 和清单文件磁带的返回日期设置为第 25 天。

注：

整个周期前七天中分配给保管库的所有 MVC 均会排空。如果 DR VTV 的失效日期设置正确，则此时应没有当前 VTV，且逻辑排空过程将会快速执行。如果当前 VTV 写入新的 MVC，则原因将是所设置的失效日期不正确。

- 第 23 天和第 24 天一卷保留在异地。
- 第 25 天—第 1 天至第 7 天保管的卷均会返回，并从保管库状态中删除且可重用。
- 第 29 天—整个周期重复进行。

注：

某些站点可能会选择仅将完全卷备份保留在异地，而将增量备份保留在站点中。在这种情况下，请将增量 MVC 和相关的清单文件以及本机卷放在容器中。该容器将会锁定并直到第 25 天返回时才会重新打开。此时，第 1 天创建的所有卷均应失效。

远程磁带库中的含 MVC 的 DR 保管

在此过程中，卷将保管至远程磁带库 (ACS) 中，而不是在物理保管库中保管卷。除了以下几点以外，此过程与“含 MVC 的基本 DR 保管”类似：

- 第 1 步至第 3 步—没有变化。
- 第 4 步—已被消除（因为不弹出保管的 MVC）。
- 第 5 步—与第 4 步相同，不弹出本机卷。相反，*Action Vault Assign* 语句会使用与执行和弹出这些本机卷相同的选择标准将本机卷分配给保管库，如以下示例所示。

```
Options
  NoSync
  ;
TMS
  RMM
  Dateform(J)
  DDname(LCMTMSDB)
  ;
Vault
  Name('DRVLT')
  GracePeriod(3)
  ;
Action
  Vault
  Assign
  Vault('DRVLT')
  When(
    (InLsm)
    and
    (DataSetName EQ 'DRVLT.MANIFEST')
    and
    (TMSScratch EQ False)
  )
  ;
```

- 第 6 步—已被消除（因为没有重新装入卷）。
- 第 7 步—没有变化。

注：

因为排空处理发生在远程磁带库内，因此执行时比手动保管库中的卷逻辑排空处理更有效。

- 第 8 步—没有变化。
- 第 9 步—在基本过程中，保管库分配将通过装入处理（这在远程磁带库方案中不会发生）删除。要删除保管卷，请使用 *Action Vault Release* 语句。要释放的保管卷由 *Vault Name* 和 *Return Date* 选择，像以前生成拉式列表一样。*Action Vault Release* 语句仅处理其 *Return Date* 已发生的保管卷。

下面显示了 *Action Vault Release* 语句的示例。

```
Options
  NoSync
  NoTMS
;
Vault
  Name('DRVLT')
  GracePeriod(3)
;
Action
  Vault
  Release
  When(
    (VaultName EQ 'DRVAULT')
    and
    (VaultReturnDate LE TODAY)
    and
    (VaultReturnDate NE MISSING)
  )
;
```

保管用于 LTR 的 MVC

使用 MVC 保存长期保留 (Long Term Retention, LTR) MVC 的过程基本上与“[含 MVC 的基本 DR 保管](#)”中所述的灾难恢复基本过程相同。对于 LTR 使用，两大主要考虑事项包括：

- 移动至保管库（“[第 2 步—导出保管库 MVC](#)”至“[第 5 步—弹出本机卷（包括清单文件磁带）](#)”）可能并非每天均会执行，因此越来越多的 LTR 卷在保管之前便已完全填充。
- LTR VTV 不会长时间失效，因此上述定期处理可能仅会在更长的时间间隔发生。系统中仍会有一些仅部分填充的 LTR MVC。因此，过一段时间必须处理这些使用少

量 VTV 部分填充的 MVC 和使用较少的 MVC，使部分填充的 MVC 中的 VTV 合并至较少的 LTR MVC 中。

在未来的某个时刻，可能需要执行 LTR MVC 的逻辑排空，将归档的数据移动到新介质。基本过程可轻松执行该活动，只需选择相应的选择标准，并限制一次处理的保管 LTR MVC 数量。每次执行时，仍处于当前状态的 VTVS 将移动到新介质，这些新的 MVC 将移动到保管库，且逻辑排空的 MVC 将返回以供重用或销毁。

将特定的卷弹出至本地（基础）保管库

许多站点可能要求从自动化环境中删除特定的卷，并将这些卷高效存储至本地环境中的机架上。这种要求可能源自某些活动，例如：停止使用卷；希望将卷保留一段时间。多个本地/基础保管库可以定义为将特定的卷发送到不同的保管库，但只有一个本地/基础保管库可定义为“默认”保管库。本地/基础保管库中的所有卷均会自动分配当前日期作为返回日期。这样一来，无需任何进一步操作，这些卷均返回至自动化环境并从保管库分配中删除。

以下示例显示了将特定卷弹出至基础保管库的 LCM 参数文件示例。

```
Options
  NoSync
  ;
TMS
  RMM
  Dateform(J)
  DDname(LCMTMSDB)
  ;
Vault
  Name('FLOOR')
  Default
  ;
Action
  Eject
  When(
    (InLsm EQ True)
    and
    (DaysSinceReference GT 100)
    and
    (MVC EQ False)
    and
    Not
    (DataSetName Matches 'HMIG.**')
  )
  Control(
    VaultSlot
    Ascending
  )
  Ejmsg('Move to Floor Vault')
  ;
Manage
  ACSID(00)
  Numfree(500)
  ;
```

在此示例中：

- *OPTIONS* 语句指定了 *NOSYNCH*，这是因为保管功能不使用 TMS 信息进行保管。
- *TMS RMM* 语句对于添加 TMS 元数据的处理必不可少。
- *VAULT* 语句将 *FLOOR* 指定为默认基础保管库。
- *ACTION EJECT* 语句指定将卷弹出至具有以下特点的基础保管库：
 - 位于 LSM 中。
 - 超过 100 天未被引用。
 - 不是 MVC。
 - 其数据集名称掩码为 *HMIG.***。
- *ACTION EJECT* 语句还指定：
 - 以卷序列号升序处理卷。
 - 按 TMS 插槽编号弹出。
 - 弹出消息。
- *MANAGE* 语句指定：
 - 在 ACS 00 中管理卷。
 - 确保 ACS 中有 500 个空闲单元。

第 4 章 在 DR 解决方案中使用跨 TapePlex 复制

"执行物理导出和导入"一章介绍了如何从源站点创建"导出"可移植 MVC，以便您能够以物理方式将这些 MVC 移至目标站点并将 MVC（及其包含的 VTV）导入到目标站点。有了跨 Tapeplex 复制 (Cross-Tapeplex Replication, CTR)，您不再使用 PTAM (Pickup Truck Access Method, 卡车运送存取方法) 将 MVC 从一个站点移至另一个站点，而是以电子方式将 VTV 从源站点移至目标站点，即从一个 TapePlex 移至另一个 TapePlex（随后在此处将 VTV 迁移到 MVC），这样可消除 PTAM 步骤。随着 VTV 的副本从源 TapePlex 移至目标 TapePlex，VTV 的元数据副本也会从源 TapePlex 的 CDS 移至目标 TapePlex 的 CDS。请注意，源 TapePlex 继续"拥有"CTR VTV 并管理其暂存。

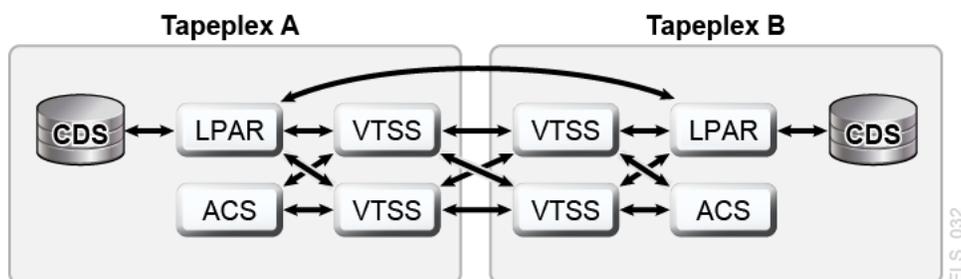
注意：

请注意，如果使用的是 CTR，停止 SMC 会使 VTCS 停止向 CTR TapePlex 发送元数据，这样实际上会停止数据传输。因此，如果使用的是使用 SMC 通信服务（例如 CTR）的 HSC 功能，则应该确保在停止 SMC 之前 HSC 活动已停顿或 HSC 已终止。

CTR 如何工作？

看一下如图 4.1 "ELS CTR 配置"中所示的 CTR。

图 4.1. ELS CTR 配置



如图 4.1 "ELS CTR 配置"中所示：

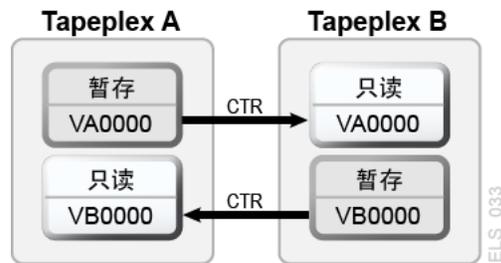
- CTR 在两个单独的 TapePlex 中使用两个 VTSS 之间的连接 (CLINK) 将数据从一个 VTSS 发送到另一个 VTSS。连接可以是单向或双向。
- CTR 使用 SMC 客户机/服务器服务功能将元数据从发送 TapePlex 发送到接收 TapePlex。请注意，您无需使用客户机/服务器功能在 SMC 与 HSC 之间进行通信即可使用 CTR，但是您必须在 SMC 中定义 HTTP 和 SERVER 命令以允许传输元数据。

- 每个站点都有单独的（并且单独维护的）CDS，因此在一个站点中失去连接或硬件可用性不会直接影响其他任何站点。
- 配置和物理连接要求简单明了。
- 您现在可以更简单地运行并发 DR 测试并且不会对现有工作造成中断（不使用 CDRT 实用程序）。
- 您现在可以将工作负荷从一个站点自动切换到另一个站点。
- 图 4.2 “站点内 VTV 卷关系” 中显示了两个 TapePlex 的 VTV 卷范围。请注意，每个 TapePlex 都有自己的一组可写卷，这些卷在另一个 TapePlex 上按只读版本进行镜像。
- 在显示的配置中，发送 TapePlex 上的两个 VTSS 连接到接收 TapePlex 上的两个 VTSS，以实现最大的弹性。

注：

在群集 VTSS 和 CTR 配置中，必须确保为复制保留每个 VTSS 中的前 16 个 VTD (0-F)。这些设备必须在 MVS 中脱机，而其路径必须在每个 HSC 服务器主机中联机。VTCS 并不向 SMC/HSC 注册前 16 个 VTD，这样可防止在这些 VTD 上挂载 VTV。

图 4.2. 站点内 VTV 卷关系



请阅读“CTR VTV 只读注意事项”，然后转至“配置 CTR”。

CTR VTV 只读注意事项

当您使用 CTR 时，从一个站点复制到另一个站点的所有 VTV 在远程站点都处于只读模式。尽管这些 VTV 在实际发生灾难时可由远程 TapePlex 进行暂存（并重用其各自的卷序列号），但是只要卷未处于 SCRATCH 状态便无法更改只读状态。请注意，切勿将 `POOLPARM EXTERNAL` 池中的卷设置为 SCRATCH 状态。

因此，如果选择使用 CTR 作为业务连续性或灾难恢复策略，则必须确保应用程序不会在 DR 测试期间或实际发生灾难时更新这些卷。应考虑以下情况：

1. 在 JCL 中使用属性 `DISP=MOD` 或动态分配将数据附加到现有数据集的应用程序必须实施检查点/重新启动机制，并且必须在创建任何 `DISP=MOD` 卷之前记录一个检查点。通过在检查点重新启动来恢复这些应用程序，在适当的情况下，必须在重新启动后重新创建这些 `DISP=MOD` 卷。请注意，使用 `DISP=MOD` 本身对跨 TapePlex 复制来说不是问题。只要应用程序的检查点允许其收回部分更新，或者其设计允许输出新数据以在新卷上开始，那么针对只读 VTV 运行就应该没有问题。

2. 如果复制到另一个 TapePlex 的 VTV 由 HSM 拥有，可按以下过程收集数据以在新卷上开始并避免更新现有 HSM VTV：
 - a. 将现有卷标记为已满。
 - b. 如有必要，为 *USERUNITTABLE*、*MIGRATION*、*BACKUP* 和 *RECYCLE* 修改 *ARCCMD*。
 - c. 确保将 *RECYCLEDALLOCFREQ* 设置为 1。这样允许 HSM 分配在适当的情况下分配新卷和设备。
 - d. 根据 *MGMTCLAS VTVSIZE*，设置 *PERCENTFULL*：

对于 800 MB VTV，将 HSM PERCENTFULL 设置为 97。

对于 4 GB VTV，将 HSM PERCENTFULL 设置为 450。

对于 32 GB VTV，将 HSC PERCENTFULL 设置为 3600。

在现有卷上堆放数据集的应用程序同样受到上述 *DISP=MOD* 限制。

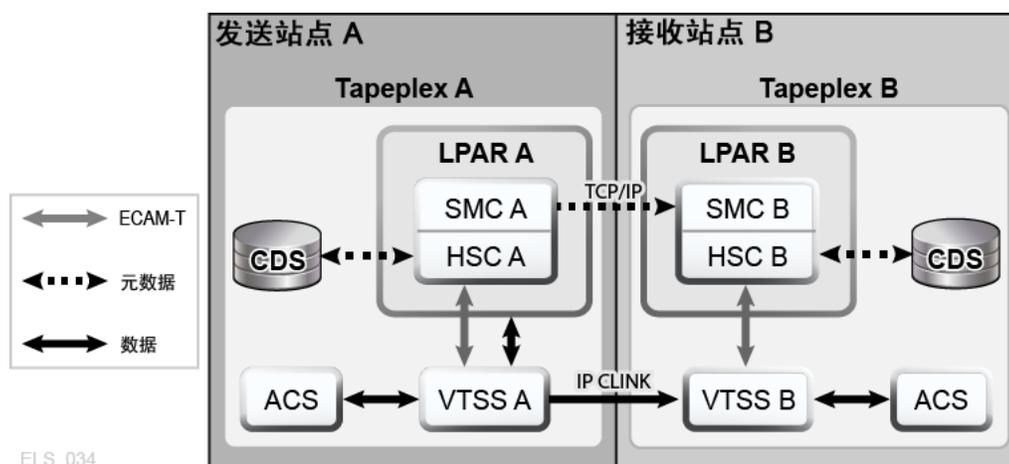
配置 CTR

图 4.3 “CTR 配置”显示了 CTR 配置的示例。在此系统中，VTSS VTSSA 位于 TapePlex TAPEPLXA 中，通过“伙伴”CLINK 连接到 TapePlex TAPEPLXB 中的 VTSS VTSSB。复制到 VTSSB 的 VTV 现在位于 TAPEPLXB 的 CDS 中，与 VTV 随后迁移到的 MVC 一样。也就是说，VTV 跨 TapePlex 进行复制，然后在本地迁移。发送 TapePlex 中的 VTSS 无法与接收 TapePlex 中的 RTD 建立连接。

注：

以下示例显示了单向 CTR。要执行双向 CTR，只需在两个 TapePlex 上以相同的方式定义配置和 SMC 客户机/服务器控制语句。请注意，单个 TapePlex 也可以从多个其他 TapePlex 接收 VTV。要定义一个 TapePlex 从多个其他 TapePlex 接收数据的配置，只需将附加 TapePlex 名称添加到 *CONFIG of TAPEPLXB*。

图 4.3. CTR 配置



设置：配置和启动 CTR

要配置和启动“配置 CTR”中显示的示例 CTR 系统，请执行以下操作：

1. 确保您的系统满足安装 *ELS* 中所述的群集 VTSS 要求。
2. 在正在主机 LPARB 上运行的 SMC 下启动 HTTP 服务器。

您可能要在 SMC CMDS 文件中执行该操作。例如：

```
HTTP START PORT(999)
```

3. 在主机 LPARB 上定义 TAPEPLEX 和 SERVER 命令。

同样，您可能要在 SMC CMDS 文件中执行该操作。例如：

```
TAPEPLEX NAME(TAPEPLXA) LOCSUB(HSCA)
TAPEPLEX NAME(TAPEPLXB)
SERVER NAME(REMB)TAPEPLEX(TAPEPLXB) HOSTNAME(LPARB) PORT(999)
```

注：

在示例配置中，TapePlex TAPEPLXB 存在（从 TapePlex TAPEPLXA 的角度来看）的唯一目的就是维护包含有关已从 TAPEPLXA 复制的 VTV 的元数据的 CDS。但是，如果 TapePlexes TAPEPLXA 和 TAPEPLXB 中的 HSC 或 VTCS 定义使用相同的设备地址引用不同的物理设备，则您必须定义 *SMC UNITATTR* 命令来告知 SMC 哪个 TapePlex 定义其主机上的设备。尽管 *UNITATTR* 必须指定 *MODEL*，但是如果指定的模型与 TapePlex 报告的模型不匹配，则实际模型将覆盖 *UNITATTR MODEL*。下面是 TapePlexes TAPEPLXA 和 TAPEPLXB 均定义了地址范围 9000-90FF 时将会使用的 SMC *UNITATTR* 语句的示例：

```
UNITATTR ADDR(9000-90FF) TAPEPLEX(TAPEPLXA) MODEL(VIRTUAL)
```

4. 为 TapePlex A 编码 *CONFIG* 面板，如例 4.1 “TapePlex A 的 *CONFIG* 面板” 中所示。

在此图中，请注意：

- *TAPEPLEX* 语句定义此 TapePlex。
- *CLINK* 语句定义用于从 VTSSA 到 VTSSB 的 CTR 的 *CLINK*。
- 对于 TAPEPLXA，*CONFIG GLOBAL* 语句中的条件性复制设置是 *CHANGED*。

5. 为 TapePlex B 编码 *CONFIG* 面板，如例 4.2 “TapePlex B 的 *CONFIG* 面板” 中所示。

在此图中，请注意：

- *TAPEPLEX* 语句包括 *RECVPLEX=TAPEPLXA* 参数来指定 TAPEPLXB 可以从 TAPEPLXA 接收 VTV。
- 没有 *CLINK* 语句，因为在 TAPEPLXA 的 *CONFIG* 面板中定义了 *CLINK*。

示例 4.1. TapePlex A 的 CONFIG 面板

```
//CREATCFG EXEC PGM=SLUADMIN,PARM='MIXED'
//STEPLIB DD DSN=h1q.SEALINK,DISP=SHR
//SLSCNTL DD DSN=h1q.TAPEPLXA.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=h1q.TAPEPLXA.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=h1q.TAPEPLXA.DBASESBY,DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
CONFIG RESET CDSLEVEL(V62ABOVE)
GLOBAL MAXVTV=65000 MVCFREE=60 VTVATTR=SCRATCH RECALWER=YES
LOCKSTR=STK_VTCS_LOCKS VTPAGE=LARGE REPLICAT=CHANGED
RECLAIM THRESHLD=70 MAXMVC=30 START=98 CONMVC=1
TAPEPLEX THISPLEX=TAPEPLXA
VTSS NAME=VTSSA LOW=71 HIGH=80 MAXMIG=8 MINMIG=1 RETAIN=10
RTD NAME=VSMA1A00 DEVNO=1A00 CHANIF=0C
RTD NAME=VSMA1A01 DEVNO=1A01 CHANIF=0D
RTD NAME=VSMA1A02 DEVNO=1A02 CHANIF=0K
RTD NAME=VSMA1A03 DEVNO=1A03 CHANIF=0L
RTD NAME=VSMA2A08 DEVNO=2A08 CHANIF=1C
RTD NAME=VSMA2A09 DEVNO=2A09 CHANIF=1D
RTD NAME=VSMA2A0A DEVNO=2A0A CHANIF=1K
RTD NAME=VSMA2A0B DEVNO=2A0B CHANIF=1L
VTD LOW=7900 HIGH=79FF
VTD LOW=8900 HIGH=89FF
CLINK VTSS=VTSSA CHANIF=0G REMPLEX=TAPEPLXB PARTNER=VTSSB
CLINK VTSS=VTSSA CHANIF=0O REMPLEX=TAPEPLXB PARTNER=VTSSB
```

示例 4.2. TapePlex B 的 CONFIG 面板

```
//CREATCFG EXEC PGM=SLUADMIN,PARM='MIXED'
//STEPLIB DD DSN=h1q.SEALINK,DISP=SHR
//SLSCNTL DD DSN=h1q.TAPEPLXB.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=h1q.TAPEPLXB.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=h1q.TAPEPLXB.DBASESBY,DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
CONFIG RESET CDSLEVEL(V62ABOVE)
GLOBAL MAXVTV=65000 MVCFREE=60 VTVATTR=SCRATCH RECALWER=YES
LOCKSTR=STK_VTCS_LOCKS VTPAGE=LARGE
```

```

RECLAIM THRESHLD=70 MAXMVC=30 START=98 CONMVC=1
TAPEPLEX THISPLEX=TAPEPLXB RECVPLEX=TAPEPLXA
VTSS NAME=VTSSB LOW=75 HIGH=80 MAXMIG=8 MINMIG=1 RETAIN=10
RTD NAME=VSMB3A00 DEVNO=3A00 CHANIF=0C
RTD NAME=VSMB3A01 DEVNO=3A01 CHANIF=0D
RTD NAME=VSMB3A02 DEVNO=3A02 CHANIF=0K
RTD NAME=VSMB3A03 DEVNO=3A03 CHANIF=0L
RTD NAME=VSMB4A08 DEVNO=4A08 CHANIF=1C
RTD NAME=VSMB4A09 DEVNO=4A09 CHANIF=1D
RTD NAME=VSMB4A0A DEVNO=4A0A CHANIF=1K
RTD NAME=VSMB4A0B DEVNO=4A0B CHANIF=1L

```

为 CTR 定义策略

要为 CTR 定义策略，请参见以下过程。

发送 TapePlex 的策略

要为图 4.3 “CTR 配置”中显示的示例 CTR 系统的发送 TapePlex (TAPEPLXA) 定义策略，请执行以下操作：

1. 为 TAPEPLXA 创建 MVC *POOLPARAM/VOLPARAM* 定义：

```

POOLPARAM TYPE(MVC) NAME(MVCPLA) INITMVC(YES) MCVFREE(25) -
MAXMVC(98) THRESH(85) START(98)
VOLPARAM VOLSER(AM1000-AM1299) MEDIA(STK1R)

```

2. 为 TAPEPLXA 创建 VTV *POOLPARAM/VOLPARAM* 暂存池定义：

```

POOLPARAM TYPE(SCRATCH) NAME(ASCRPL)
VOLPARAM VOLSER(AV1000-AV1999) MEDIA(VIRTUAL) REC(VIRTUAL)

```

3. 针对 TAPEPLXA，为包含本地迁移的 VTV 的 MVC 创建存储类以及 CTR 存储类。

```

STOR NAME(LOCAL1) ACS(00) MEDIA(STK1R)
STOR NAME(EEPA1) TAPEPLEX(TAPEPLXB)

```

在以上示例中，*STORclas* 语句定义了：

- 存储类 LOCAL1，它是从每个 VTSS 中本地迁移的 VTV 的存储类。
 - 存储类 EEPA1，它是 CTR 的存储类，并指定了接收 TapePlex (TAPEPLXB)。
4. 创建指向第 3 步中存储类的管理类。

```

MGMT NAME(LOCEEX1) MIGPOL(LOCAL1) EEXPOL(EEPA1)

```

5. 创建 SMC 策略，用以指定虚拟介质并分配在第 4 步中创建的管理类。

```
POLICY NAME(PPAY) MEDIA(VIRTUAL) MGMT(LOCEEX1)
```

6. 创建 *TAPEREQ* 语句，以将关键数据发送到 VSM 并为数据分配相应的策略。

```
TAPEREQ DSN(*.PAYROLL.***) POLICY(PPAY)
```

在以上示例中，*TAPEREQ* 语句指定将包含 HLQ 掩码 **.PAYROLL.*** 的数据集发送到 VSM 并分配策略 PPAY。

注：

- 尽管您可以使用 SMC 策略使 CTR 指向特定专用设备组，但是 StorageTek 建议仅使用 MGMTCLAS，以便 SMC/VTCS 分配影响因素可以使用符合 MGMTCLAS 要求的任何 VTSS。
- 您可以使用 EEXPORT 命令执行手动 CTR。有关更多信息，请参见《*ELS Command, Control Statement, and Utility Reference*》。

7. 检查 SYS1.PARMLIB SMFPRMxx 成员，以确保启用了子类型 28 记录。

如果启用了该选项，VTSS 将针对每个 CTR 事件编写一条包括目标 VTSS 名称的子类型 28 记录。当 VTV 成功到达接收站点的 VTSS 时，您便知道 CTR 已成功。是否有办法确认这一点？是的，使用 *DRMONitr* 实用程序，如第 8 步中所述。

8. 创建 JCL 来监视 CTR。

为此，使用 *DRMONitr* 实用程序监视 CTR。*DRMONitr* 会导致关联的 MVS 作业暂停，直到成功完成 CTR。例如：

```
//MONITOR EXEC PGM=SLUADMIN,PARM='MIXED'
//STEPLIB DD DSN=h1q.SEALINK,DISP=SHR
//* If HSC IS NOT OR MAY NOT BE ACTIVE, INCLUDE THE
//* FOLLOWING:
//SLSCNTL DD DSN=primary.cds.name,DISP=SHR
//SLSCNTL2 DD DSN=secondary.cds.name,DISP=SHR
//SLSSTBY DD DSN=standby.cds.name,DISP=SHR
//SLSPARMP DD DSN=h1q.PARMLIB(BKPCNTL),DISP=SHR
//SLSPARMS DD DSN=h1q.PARMLIB(BKPCNTL2),DISP=SHR
//SLSPARMB DD DSN=h1q.PARMLIB(BKPSTBY),DISP=SHR
//SYSIN DD UNIT=SYSDA,SPACE=(TRK,1)
//* THE FOLLOWING IS USED BY THE SNAPSHOT UTILITY:
//SYSPRINT DD SYSOUT=*
//SLSPRINT DD SYSOUT=*
//SLSPRINT DD SYSOUT=*
DRMON MGMT(LOCEEX1) STOR(EIPA1) MAXAGE(24) TIMEOUT(120)
```

在此示例中，*DRMON* 实用程序监视存储类 *EEPA1* 的迁移，*EEPA1* 是第 4 步中的管理类 *LOCEEX1* 指定的 CTR 存储类。此外，仅监视在过去 24 小时内更新的 VTV，并且使 *DRMON* 在 120 分钟后超时。

接收 TapePlex 的策略

要为“配置 CTR”中显示的示例 CTR 系统的接收 TapePlex (TAPEPLXB) 定义策略，请执行以下操作：

1. 为针对 TapePlex TAPEPLXB 定义的用于保存来自 TAPEPLXA 的 CTR VTV 的 MVC 池创建 MVC POOLPARAM/VOLPARAM 定义：

```
POOLPARAM TYPE(MVC) NAME(AMVCCTR) INITMVC(YES) MCVFREE(25) -
MAXMVC(98) THRESH(85) START(98)
VOLPARAM VOLSER(BM1000-BM1099) MEDIA(STK1R)
```

注：

StorageTek 强烈建议您使用 POOLPARAM/VOLPARAM 功能确保在远程站点为 CTR 复制的卷保留卷范围。

2. 为 TAPEPLXA 导出的 VTV 创建外部 VTV 池：

```
POOLPARAM TYPE(EXTERNAL) NAME(AEXTBPL) OWNRPLEX(TAPEPLXA)
VOLPARAM VOLSER(AV1000-AV1999) MEDIA(VIRTUAL) REC(VIRTUAL)
```

注：

在此示例中，没有为 TAPEPLXB 上的生产工作定义任何池，仅定义了 TAPEPLXA 使用的池。如果要在 TAPEPLXB 上运行生产工作，则 TAPEPLXB 工作的暂存和 MVC 池需要额外的 POOLPARAM 和 VOLPARAM 定义。

3. 为用于 TAPEPLXA 工作的 TapePlex TAPEPLXB 创建 VTV 暂存池：

```
POOLPARAM TYPE(SCRATCH) NAME(ASCRPL)
VOLPARAM VOLSER(BV1000-BV1999) MEDIA(VIRTUAL) REC(VIRTUAL)
```

4. 为 TapePlex TAPEPLXB 创建 MVC 池用于保存来自 TAPEPLXA DR 测试或生产系统的 VTV（发生灾难时）：

```
POOLPARAM TYPE(MVC) NAME(AMVCDR) INITMVC(YES) MCVFREE(25) -
MAXMVC(98) THRESH(85) START(98)
VOLPARAM VOLSER(BM2000-BM2099) MEDIA(STK1R)
```

5. 针对 TAPEPLXB，为本地迁移创建存储类。

```
STOR NAME(TPEPLXA1) MVCP00L(AMVCCTR)
STOR NAME(TPEPLXA2) MVCP00L(AMVCDR)
```

在此示例中，STORclas 语句为本地迁移定义了存储类 TPEPLXA1 和 TPEPLXA2。通过存储类名称，可以将此工作与 TAPEPLXB 本地工作隔离。

6. 创建指向第 5 步中存储类的管理类。

```
MGMT NAME(LOCEEX1) MIGPOL(TPEPLXA1)
MGMT NAME(LOCPLXA) MIGPOL(TPEPLXA2)
```

请注意，名称 LOCEEX1 与在 TAPEPLXA 上使用的管理类名称匹配（该管理类在从 TAPEPLXA 上的 VTSS 发送的 VTV 元数据中指定），但是示例引用了本地迁移的存储类。TAPEPLXB 上管理类和存储类的定义可以使用任何参数（包括 EEXPOL）以复制到第三个 TapePlex。此外，您还可以创建要在 TAPEPLXA 工作负荷的 DR 测试期间用于迁移的另一个 MGMTCLAS，即 LOCPLXA。

当远程站点没有 LPAR 时使用 CTR

在某些环境中，只有一个站点中使用 LPAR 执行磁带活动，而另一个站点仅包含磁带库和 VTSS 硬件而没有 MVS LPAR。可以将这样的环境设置成 CTR 用作 DR 和 DR 测试机制。

为此，您必须：

1. 在生产环境中运行 SMC 客户机/服务器功能，以便至少有一个生产 LPAR 未在运行 HSC/VTCS。

或者，您也可以使用 MULT 模式功能与生产 TapePlex 在相同的 LPAR 上运行 DR TapePlex。有关使用该功能的更多信息，请参见配置 HSC 和 VTCS。

本例中的生产 TapePlex 是 TAPEPLXA。

2. 创建一个新的 CDS 用来定义远程站点中的硬件（磁带库和 VTSS）。
3. 在当前未运行生产 HSC/VTCS 的 LPAR (MVSX) 上使用新的 CDS 启动 HSC/VTCS，或者在您决定使用 MULT 模式功能运行多个 HSC/VTCS 副本的 LPAR 上启动 HSC/VTCS。

注：

为了确保可靠性，建议您可能要运行两个 HSC/VTCS 实例，使其指向两个不同 LPAR 上的 TAPEPLXB，这样如果一个实例不可用，跨 TapePlex 复制的 VTV 的元数据可以发送到第二个实例。

此系统是 TapePlex TAPEPLXB。

4. 在定义 TAPEPLXA 和 TAPEPLXB TapePlex 的 MVSX 上定义 SMC 系统的参数。

复合系统中的每个 SMC 系统必须定义 TapePlex TAPEPLXA（即生产 TapePlex）和 TAPEPLXB（即 DR TapePlex）。要支持在 DR 测试期间继续复制 VTV，必须为 TapePlex TAPEPLXB 定义一个服务器，使其指向远程站点的主机。例如：

```
TAPEPLEX NAME(TAPEPLXA) LOCSUB(HSCA)
TAPEPLEX NAME(TAPEPLXB)
SERVER NAME(TPLXBPR) TAPEPLEX(TAPEPLXB) HOST(MVSX) PORT(999)
SERVER NAME(TPLXBDR) TAPEPLEX(TAPEPLXB) HOST(MVSXDR) PORT(1234)
```

注：

此示例假定尽管生产和 DR 站点上的 LPAR 名称 (MVSX) 可能完全相同，但是两个站点具有唯一的 TCP/IP 主机名称。

- 为 TAPEPLXA 定义 VTCS 策略，以允许向 TAPEPLXB 执行 CTR。

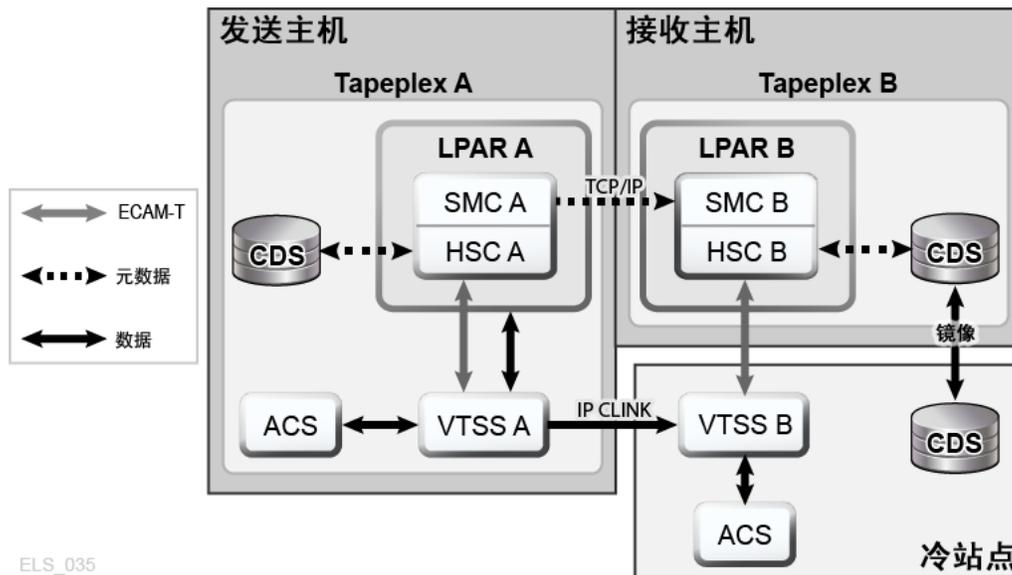
请参见“为 CTR 定义策略”。

- 使用磁盘复制解决方案，在远程位置为 TAPEPLXB 维护 CDS 内容的副本。

或者，如果存在可靠的连接，您可能要使用 FICON 连接从生产站点访问 CDS，在 DR 站点维护 HSC CDS 的主副本（以及其他副本）。

图 4.4 “CDS 副本—远程站点没有 LPAR” 显示了远程站点没有 LPAR 时的 CDS 副本。

图 4.4. CDS 副本—远程站点没有 LPAR



ELS_035

使用 CTR 作为 DR 解决方案

DR 解决方案始终需要执行以下三项操作：

- 设置和启动解决方案，如“[设置：配置和启动 CTR](#)”中所述。
- 如果发生灾难，使用解决方案在远程站点继续开展业务，如“[使用 CTR 实现业务连续性](#)”中所述。
- 在本地站点重新启动并运行后，使用解决方案在本地站点继续开展业务，如“[使用 CTR 实现业务恢复](#)”中所述。

使用 CTR 实现业务连续性

如果站点 TAPEPLXA 发生了中断，您可以在站点 TAPEPLXB 继续开展业务，只需使用 TAPEPLXB 的 TapePlex 运行工作负荷即可。为了保护数据，从 TAPEPLXA 复制的 VTV 将保持只读状态（请参见“[CTR VTV 只读注意事项](#)”）。但是，成功地重新建立 TAPEPLXA 工作负荷后，您可能要暂存从 TAPEPLXA 复制的某些 VTV。请注意，在执行该步骤前应该小心，确保 TAPEPLXA 和 TAPEPLXB 的生产工作稳定。此外，在将来某个时间，您或许要为 TAPEPLXA 重新创建单独的 TapePlex 环境以返回原始配置。

要使用 CTR 实现业务连续性，请执行以下操作：

1. 在 TAPEPLXB TapePlex CDS 中将名为 *AEXTBPL* 的池的 *POOLPARM/VOLPARM* 定义从 *TYPE(EXTERNAL)* 更改为 *TYPE(SCRATCH)*：

```
POOLPARM TYPE(SCRATCH) NAME(AEXTBPL)
VOLPARM VOLSER(AV1000-AV1999) MEDIA(VIRTUAL)
```

请注意，VOLPARM VOLSER 范围保持不变。

2. 您现在可以在 TAPEPLXB 下运行暂存同步作业，以根据 TMS 中的暂存状态在 AV1000-AV1999 范围内暂存 VTV，以返回检查点或执行正常的暂存更新处理。

开始生产处理后可能要经过一些时间才允许 AV1000-AV1999 范围内的 VTV 卷序列号重用作暂存卷。使用 *POOLPARM/VOLPARM* 功能可确保不能选择这些卷作为暂存卷，除非策略明确要求 *SUBPOOL(AEXTBPL)*。

在此期间，您将 TAPEPLXB 暂存子池 ASCRPL 中的卷（卷序列号范围 BV1000-BV1999）用于 TAPEPLXA 生产工作。

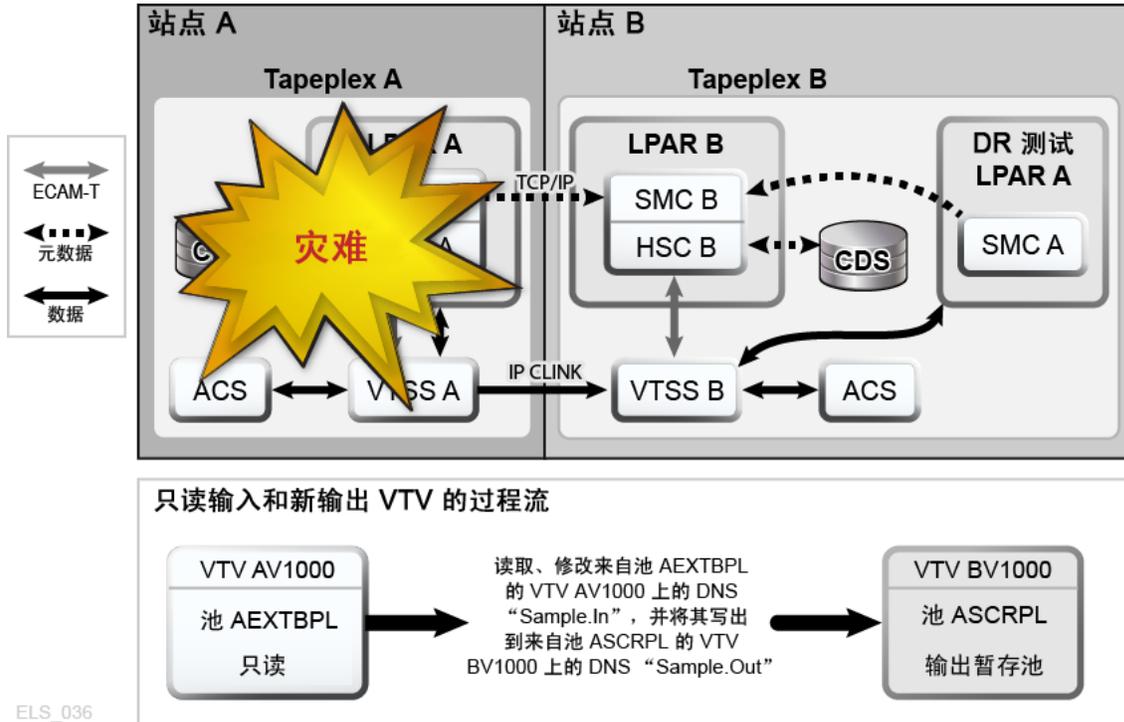
灾难恢复环境稳定后，您可以再次更改 *POOLPARM/VOLPARM* 定义，以允许从 AV1000-AV1999 范围内选择暂存卷：

```
POOLPARM TYPE(SCRATCH) NAME(ASCRPL)
VOLPARM VOLSER(AV1000-AV1999) MEDIA(VIRTUAL)
```

VOLPARM VOLSER(BV1000-BV1999) MEDIA(VIRTUAL)

图 4.5 “业务连续性期间的系统” 显示了业务连续性期间的 CTR 示例。

图 4.5. 业务连续性期间的系统



ELS_036

使用 CTR 实现业务恢复

本地站点发生了中断，您在远程站点继续开展业务，现在本地站点再次启动并运行，如何在本地站点继续开展业务呢？从根本上讲，业务恢复取决于中断期间和中断过后发生了什么。假定所有原始本地数据都已丢失，本地站点有一个全新的空 VTSS。

要在本地站点丢失所有数据后恢复业务，请执行以下操作：

1. 创建一个新的 CDS 并运行 HSC 审计，以确定物理磁带库的内容。

随后需要将数据和元数据从远程站点“反向复制”到本地站点。

2. 设置远程站点的 CONFIG 面板，以使其可以向本地站点发送数据。
3. 使用 EEXPORT 反向复制。

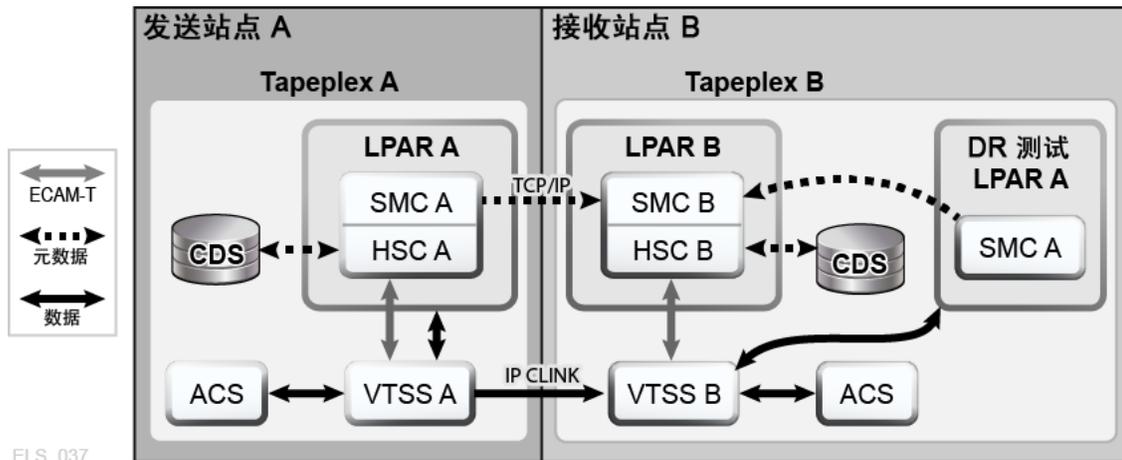
例如：

```
EEXPORT MGMTCLAS(LOCEEX1, LOCEEX2) TOPLEX(TAPEPLXA)
```

使用跨 Tapeplex 复制进行灾难恢复测试

继续举个例子，假设有两个站点 TAPEPLXA 和 TAPEPLXB，每个都定义为自己的 TapePlex (HSC CDS)，并且您使用跨 TapePlex 复制功能将关键 VTV 从 TAPEPLXA 复制到了 TAPEPLXB。图 4.6 “使用跨 Tapeplex 复制进行灾难恢复测试” 显示了使用 CTR 进行的 DR 测试。

图 4.6. 使用跨 Tapeplex 复制进行灾难恢复测试



要在 TAPEPLXB 上为 TAPEPLXA 的工作执行 DR 测试，建议采用以下过程：

1. 确保 TAPEPLXB CDS 包含一个或多个暂存子池来存储 TAPEPLXA 输出数据，这些暂存子池与用于 TAPEPLXB 工作的暂存子池分离。

有关示例，请参见“[接收 TapePlex 的策略](#)”。

2. 确保来自 TAPEPLXA 的目录和磁带管理数据可用。
3. 在 TAPEPLXA 测试 LPAR 上初启 SMC，将其 TapePlex 定义为 TAPEPLXB，为 TAPEPLXB 中的一个或多个 HSC 主机指定 `SERVER` 命令。
4. 开始执行测试工作负荷。

SMC 将自动可以访问在开始测试之前或之后存在的 VTV，因为继续从 TAPEPLXA 复制这些 VTV。确保 DR 测试将要使用的 VTV 不会由 TAPEPLXA TapePlex 暂存或更改。

5. 当测试完成时，暂存测试使用的 DR 测试子池中的所有 VTV。

使用这种方法时，无需特殊的 CDS 也无需特殊的规则来确保两个单独的 HSC 系统可以共享硬件资源。但是，这种方法要求使用最新数据或者至少是当前可用的数据来执行 DR 测试。请注意，DR 测试输出和 TAPEPLXA 复制的 VTV 将使用 TAPEPLXB VTSS 缓冲区空间。

由于从 TAPEPLXA TapePlex 复制的数据是只读数据，因此 DR 测试要修改这些数据的任何尝试都将导致出现消息 SMC0247 "Mount failed for write-

protected VTV *vvvvv* on drive *dddd* from SMC indicating that the VTV cannot be mounted". 出现该消息可能表明 DR 过程未清楚地定义应用程序检查点（请参见“CTR VTV 只读注意事项”）。如果确实如此，将 CTR 用于 DR 策略可能不是很好的选择。

注：

如果使用跨 TapePlex 复制在远程站点创建 VTV 副本，建议不要将 CDRT 用于 DR 测试，因为使用 CDRT 将不允许更新只读 VTV，即使在单独的 CDRT 环境中也是如此。

当 DR 站点没有 LPAR 时进行 DR 测试

当您使用在生产站点执行的 TapePlex 管理 DR 站点的 CTR 硬件时，对 DR 测试有一些额外的注意事项。在此示例中，将 TAPEPLXA 用于生产 TapePlex，将 TAPEPLXB 用于通常在生产站点运行但是在 DR 测试期间在 DR 站点运行的 TapePlex。

1. 您必须在测试之前在生产站点停止 DR TapePlex TAPEPLXB。

在 DR 测试期间，TAPEPLXB 将在 DR 站点的 TAPEPLXB CDS 副本上执行。

2. 生产 VTV 将继续发送到 TAPEPLXB 并反映在 DR 站点的 CDS 中。

在此期间，生产站点的 TAPEPLXB CDS 变得过时，因为它不再反映在 DR 测试期间复制的 VTV。生产 LPAR 上的 TAPEPLEX 和 SERVER 语句可确保在 DR 测试期间继续复制数据：

```
TAPEPLEX NAME(TAPEPLXB)
SERVER NAME(TPLXBPR) TAPEPLEX(TAPEPLXB) HOSTNAME(MVSX) PORT(999)
SERVER NAME(TPLXBDR) TAPEPLEX(TAPEPLXB) HOST(MVSXDR) PORT(1234)
```

3. 在 DR 站点为 TapePlex TAPEPLXB 启动 HSC/VTCS 时，必须确保在 SMC 上启动 HTTP 服务器：

```
HTTP START PORT(1234)
```

端口号 (1234) 与在 TAPEPLXBDR SERVER 语句中定义的内容匹配。

4. 当测试结束时，暂存测试创建的所有 VTV。

您不必确定测试实际上创建了哪些 VTV，只需暂存子池中的所有卷。例如：

```
SCRATCH VOL(BV1000-BV2999)
```

5. 在 DR 站点为 TAPEPLXB 停止 HSC/VTCS。
6. 您现在必须确保将 DR 站点的 TAPEPLXB CDS 发送回生产站点。

理想情况下，这可通过在 DR 测试期间将 TAPEPLXB CDS 镜像回生产站点来完成。如果不能做到这一点，您可以使用 FTP 或自己选择的其他机制将 CDS 从 DR 站点上的最新版本复制回生产站点。

7. 在生产站点的 LPAR 上重新启动 TAPEPLXB。

尽管没有活动的 TAPEPLXB 副本，但是计划 CTR 到 TAPEPLXB 的 VTV 将留在 VTSS 缓冲区中。当 TAPEPLXB 在生产站点再次处于活动状态时，会将这些 VTV 复制到 DR 站点的 VTSS。

管理使用跨 TapePlex 复制 (Cross-TapePlex Replication, CTR) 复制的 VTV

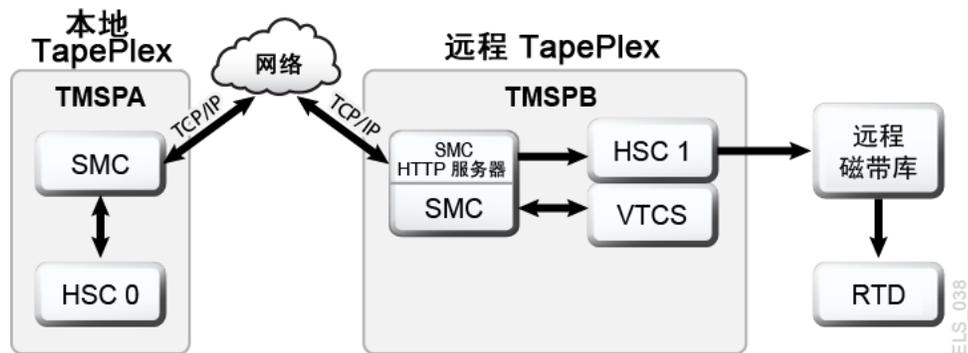
您可以使用 `VTVMaint` 更改通过 CTR 复制的 VTV 的状态，如下所示：

- 使用 `VTVMaint DELEXpot` 删除引用 VTV 的 TapePlex 的名称。例如，如果将 VTV 从 TAPEPLXA 复制到 TAPEPLXB，然后删除 TAPEPLXA 上的副本，您可以使用 `VTVMaint DELEXpot` 删除 TAPEPLXA 对 VTV 的引用。
- 使用 `VTVMaint ADDEXpot` 添加引用 VTV 的 TapePlex 的名称，如“[使用 CTR 实现业务连续性](#)”中所述。
- 使用 `VTVMaint` 实用程序更改通过 CTR 接收的 VTV 的所有权，但是 VTV 当前必须处于暂存状态。例如，`VTVMaint OWNRPlex(TAPEPLXB)` 会将从 TAPEPLXA 发送的 VTV 的所有权更改为由其当前所在的 TapePlex 拥有。

第 5 章 配置远程磁带库

图 5.1 “远程 TapePlex 下的 RTD 操作” 显示了远程磁带库的配置示例；以下各节介绍如何配置此示例。

图 5.1. 远程 TapePlex 下的 RTD 操作



修改 SMC SCMDS 文件

SMC 管理 VTCS 和远程 TapePlex 之间的所有通信，因此 SMC 必须了解如何连接到远程 TapePlex。为此，请定义本地和远程 TapePlex，以及一个或多个用于定义远程 TapePlex 的 TCP/IP 控制路径的 `SMC SERVER` 语句。您可能需要在 `SMC CMDS` 文件中执行此操作，如下例中所示。

```
TAPEPLEX NAME(TMVSA)LOCSUB(HSC0)
TAPEPLEX NAME(TMVSB)LOCSUB(HSC1)
SERVER NAME(TMVSB)IP(192.168.1.10)PORT(60000)
```

此示例包含：

- 一个 `TAPEPLEX` 语句，该语句定义一个本地 TapePlex（即 TMVSA）以及在本地 MVS 主机上运行的 HSC (HSC0)。
- 另一个 `TAPEPLEX` 语句，该语句定义远程 TapePlex（即 TMVSB）以及在远程 MVS 主机上运行的 HSC (HSC1)。
- 一个 `SERVER` 命令，该命令定义 TMVSB 的 UUI 通信路径，其中：
 - 远程服务器名称是 TMVSB。
 - `IP` 参数值为 UUI 通信的 ELS 端口 IP 地址 192.168.1.10。

- *PORT* 参数值为 60000；该值通常用于 *SERVER PORT* 参数，以便 SMC 与 TMVSB 通信。

更新 VTCS CONFIG 面板以定义远程磁带库

您必须更新 *VTCS CONFIG* 面板，定义远程磁带库以及从 VTSS 到远程磁带库的连接。如下例中所示，远程磁带库由 *CONFIG STORMNGR* 语句定义。

```
TAPEPLEX THISPLEX=TMVSA
STORMNGR NAME=TMVSB
VTSS NAME=VTSS1 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
RTDPATH NAME=RM1RTD1 STORMNGR=TMVSB DEVNO=1A00 CHANIF=0A:0
RTDPATH NAME=RM1RTD2 STORMNGR=TMVSB DEVNO=1A01 CHANIF=0A:1
RTDPATH NAME=RM1RTD3 STORMNGR=TMVSB DEVNO=1I00 CHANIF=0I:0
RTDPATH NAME=RM1RTD4 STORMNGR=TMVSB DEVNO=1I01 CHANIF=0I:1
VTD LOW=6900 HIGH=69FF
```

在本例中，请注意：

- *CONFIG TAPEPLEX* 语句将 TMVSA 指定为本地 TapePlex。
- *STORMNGR* 语句将 TMVSB 指定为远程磁带库。
- 针对 VTSS1 的 *CONFIG RTDPATH* 语句，指定了：
 - *RTDPATH* 的名称。
 - 到远程磁带库的连接 (*STORMNGR=TMVSB*)。
 - 设备编号 (*DEVNO*)。
 - VTSS 到 RTD 的每个连接的 *CHANIF* 值（格式为 *ci:p*），其中：
 - › *c* 是 0 或 1。
 - › *i* 是 A 或 I。
 - › *p* 是 0 到 3。

注：

对于 VSM5，*CHANIF* 值必须与 VSM5 "IFF Configuration Status" 屏幕上指定的值匹配。对于 VSM 6，该值必须对每个 VTSS 唯一，但不对应于 VSM 6 TCP/IP 端口上的实际值。

现在，您可以使用 *STORclas STORMNGR* 参数将数据传送到远程磁带库。例如：

```
STOR NAME(REMLIB) STORMNGR(TMVSB)
```

MVC 池注意事项

远程磁带库中的任何 MVC 均必须包含在本地 MVC 池定义（最好是通过 *VOLPARM/POOLPARM* 定义）中。本地 HSC 将远程 MVC 视为非磁带库。

通常，任何远程磁带库服务器均需将 MVC 定义为以某种方式“在外部管理”。由于未序列化，因此不支持远程 TapePlex 共享本地 MVC 的池。

如果出于数据传输目的共享 MVC 的池，则只有一个 TapePlex 能够主动写入 MVC。其他 TapePlex 可能仅具有只读访问权限。用户负责序列化磁带机上的 MVC。

第 6 章 使用群集 VTSS 配置

通过群集 VTSS，可将 VTV 从一个 VTSS 复制到另一个 VTSS。群集 VTSS 对于应用程序来说是一个强大的工具，例如但不限于 DR（Disaster Recovery，灾难恢复）解决方案。下面几节介绍基础知识，包括 VTSS 群集的定义及其工作原理：

- "什么是群集 VTSS? "
- "群集 VTSS 要求"
- "群集 VTSS 配置的工作原理"

掌握基础知识后，您还希望了解群集 VTSS 的附加变体和特性：

- "单向和双向群集"
- "扩展群集"
- "同步或异步复制"
- "使用 TCP/IP 连接形成群集"

什么是群集 VTSS?

VTSS 群集是一个高可用性 (High Availability, HA) 解决方案，可提供最高的数据可用性。它由使用 FICON 或 TCP/IP 通信链路 (CLINK) 连接的两个或多个 VTSS 系统组成。此外，群集中的每个 VTSS 系统均可访问在群集（VTSS 驻留或迁移到的群集）中创建的所有数据。在群集中创建的数据 (VTV) 在 VTCS 策略的控制下从一个 VTSS 系统复制到同一个群集中的另一个 VTSS 系统。

注：

为确保每个 VTSS 系统均可访问在群集中创建的所有数据，群集配置可以是以下任一项：

- 群集中的每个 VTSS 均连接了 RTD 或 VLE。
- “无磁带”—所有 VTSS 均未连接 VLE 或 RTD。

因此，如果群集中的一个 VTSS 发生中断，群集配置可凭借热恢复提供最高的数据可用性（复制的数据保持可用，而无需从 MVC 撤回）。

在 VTCS 7.0 以前，群集只能由两个 VTSS 组成。在 VTCS 7.0 中，多个 VTSS 可以构成单个群集。但是，VTV 在任何时间点都只能位于两个 VTSS 中。

群集可以跨越地理位置。但是，群集必须在单个 **TapePlex** 中（由单个 CDS 控制）。

可按以下任一方式将 VTV 从一个 VTSS 复制到另一个 VTSS：

- 与 VTV 的创建异步。安排为在卸载 VTV 之后尽快完成
- 与 VTV 的创建同步。复制完成后 VTV 卸载才会完成。

注：

对于通过 CLINK 或增强复制进行的同步复制，如果复制会导致超过缺少中断处理程序 (Missing Interrupt Handler, MIH) 超时值，VTSS 会终止该复制。

对于 VSM 6 和 VSM 7，MIH 超时值可以配置为 20 到 45 分钟（增量为 5 分钟）。对于 VSM4 和 VSM 5，MIH 超时值可以配置为 20 或 45 分钟。在 VTSS 中设置的值必须与在大型机上设置的值匹配。

群集中 VTSS 之间的连接可以是单向连接，即数据 (VTV) 只在一个方向上流动，也可以是双向连接，即数据 (VTV) 可以在两个方向上流动。CONFIG 实用程序可以指定群集是单向还是双向，VTV 管理类可以确定其复制策略（如果有）以及是同步还是异步完成复制。

所以，保管 MVC（如“使用 [ELS 外部保管功能](#)”中所述）和 VTV 复制都可以促进灾难恢复/业务连续性解决方案。但是，VTV 复制是较好的高可用性解决方案，因为复制时：

- 可以同步备份数据。
- 可以更快地恢复已复制到“恢复”VTSS 的近期数据，因为不必挂载 MVC。

群集 VTSS 要求

表 6.1 “群集 VTSS 要求”显示了群集 VTSS 的要求。

表 6.1. 群集 VTSS 要求

| 组件 | 要求 |
|----------------------------|--|
| 扩展群集 | 对于带有 FICON 连接的 VSM4 或 VSM5，只能运行 D02.07.00.00 或更高级别的 VTSS 微代码。对于 VSM6 和 VSM 7，可以运行所有级别的微代码。 |
| 一个群集中的 2 个 VTSS (ESCON 接口) | 主和辅助 VTSS 可以是 VSM4 的任意组合，其中辅助 VTSS 可以是任意容量。VSM5 没有 ESCON 接口，不能与使用 ESCON 的其他 VTSS 在一个群集中 |
| 一个群集中的 2 个 VTSS (FICON 接口) | 主和辅助 VTSS 可以是 VSM4 和 VSM5 的任意组合，其中辅助 VTSS 可以是任意容量。例如，以下所有项均有效： <ul style="list-style-type: none"> • 主 VSM5，辅助 VSM4 • 主 VSM5，辅助 VSM5 • 主 VSM4，辅助 VSM4 • 主 VSM4，辅助 VSM5（不建议） |
| 主和辅助 VTSS 微代码 | 主 VTSS 微代码所在的级别必须支持发送复制的 VTV。辅助 VTSS 微代码所在的级别必须支持接收复制的 VTV 并支持使用辅助 VTSS 作为生产 VTSS。安装微代码后，必须使用可选的软盘在主和辅助 VTSS 上均启用群集功能。有关详细信息，请与 StorageTek 硬件服务代表联系。 |

| 组件 | 要求 |
|-----------------------|--|
| 为群集保留的 VTD | 在群集 VTSS 配置中，必须确保为群集保留每个 VTSS 中的前 16 个 VTD (0-F)。这些设备必须在 MVS 中脱机，而其路径必须在每个 HSC 服务器主机中联机。这也适用于跨 TapePlex 复制中涉及的所有 VTSS。VTCS 并不向 SMC/HSC 注册前 16 个 VTD，这样可防止在这些 VTD 上挂载 VTV。 |
| RTD | <p>在双 ACS 环境中，必须用附加到每个 ACS 的 RTD 表示相同的设备类型，以便由群集中的任何 VTSS 迁移的数据可以由群集中的任何其他 VTSS 撤回。用于迁移的 MVC 数量、介质类型和位置由 MGMTclas 语句的 MIGPOL 参数确定。</p> <p>每个 ACS 都需要加以考虑，如果一个 ACS 中的某个磁带机类型连接到群集 VTSS 环境中的一个 VTSS，则需要将同一个 ACS 中同一类型的磁带机连接到该群集环境中的所有其他 VTSS。</p> |
| 本机 IP（使用 TCP/IP 形成群集） | <p>本机 IP 需要 CDSLEVEL F 及更高级别，PTF 如下：</p> <ul style="list-style-type: none"> 对于 6.2: <ul style="list-style-type: none"> L1A00P7—SMC6200 L1H14IM—SMS6200 L1H14O2—SOS6200 L1H14IL—SWS6200 对于 7.0, L1H150G (SES7000) 对于 7.1 及更高版本，包括基本级别的支持。 <p>本机 IP 支持以下连接：</p> <ul style="list-style-type: none"> VSM5 到 VSM5 VSM5 到 VSM6 VSM6 到 VSM6 VSM6 或 VSM5 到 VLE |

同步复制（仅适用于 VSM4 及更高版本）满足表 6.2 “同步复制要求”中所述的要求。

表 6.2. 同步复制要求

| 同步复制要求 | VTSS 微代码级别 | CDS 级别 |
|-------------------------|---|-----------|
| CLINK 的本机 IP 或 FICON 端口 | 对于 VSM4 和 VSM5, D02.03.00 .00 或更高版本。对于 VSM6 和 VSM 7, 可以运行所有级别的微代码 | "F" 或更高级别 |

同步复制（仅适用于 VSM6 及更高版本）满足表 6.3 “增强复制要求”中所述的要求。

表 6.3. 增强复制要求

| 增强复制要求 | VTSS 微代码级别 | CDS 级别 |
|-------------------------|--|-----------|
| CLINK 的本机 IP 或 FICON 端口 | 对于 VSM 6, 使用 VSM 6.3.0.03 .000 (建议使用 6.3.0.05.000) ; 对于 VSM 7, 使用 VSM 7.0.0 .05.000。 | "H" 或更高级别 |

| 增强复制要求 | VTSS 微代码级别 | CDS 级别 |
|--------|-----------------|--------|
| | 应用 PTF L1H18MB。 | |

群集 VTSS 配置的工作原理

您可以使用 VSM 通过群集链路 (CLINK) 连接两个 VTSS 来构成群集 VTSS 配置。您可以使用以下语句实施群集配置：

- 群集可以是单向或双向，具体取决于 *CLINK* 语句。
- 辅助 VTSS（或第二个对等 VTSS）可以与主 VTSS（或第一个对等 VTSS）在相同的物理位置，也可以在远程位置。
- *CONFIG CLUSTER* 语句指定构成群集的 VTSS。
- *CONFIG CLINK* 语句定义连接 VTSS 的 CLINK。编写 *CLINK* 语句的方式决定了复制是单向还是双向。有关示例，请参见“[单向群集 VTSS](#)”和“[双向群集 VTSS](#)”。
- *MGMTclas REPLICAT* 参数标识包含 VSM 从群集中的一个 VTSS 复制到另一个 VTSS 的 VTV 的管理类。

CONFIG GLOBAL REPLICAT 参数现在指定何时复制 VTV，如下所示：

REPLICAT

指定 VSM 何时复制 VTV。

ALWAYS

每次卸载 VTV 时都将复制请求添加到 VTCS 复制队列中，而不管 VTV 在挂载时是否发生了更改（默认值）。

CHANGED

当 VTV 符合以下条件时，将复制请求添加到 VTCS 复制队列中：

- 挂载时发生了更改或者
- 挂载时为只读，但是存在的 VTV 的 MVC 副本少于预期数量。

无论 *CONFIG GLOBAL REPLICAT* 设置为何，复制还要求：

- 必须在支持复制的 VTSS 中卸载 VTV 且群集中的另一个 VTSS 中不能存在完全相同的 VTV 副本。
- 除 *CONFIG GLOBAL REPLICAT* 值以外，您还必须在 VTV 的管理类上指定 *REPLICAT(YES)* 才能执行复制。

有关更多信息，请参见《*ELS Command, Control Statement, and Utility Reference*》。

- VTCS 立即迁移（使用 *KEEP*）复制的 VTV。您可以使用 *STORclas* 语句的 *MIGRATE* 参数为迁移复制的 VTV 指定源 VTSS。另请注意，您必须使用 *MIGRATE* 参数值在指向存储类的管理类上指定复制才能从所需 VTSS 进行迁移。否则，可能不会从所需 VTSS 进行迁移。

由于 VTCS 立即迁移（使用 *KEEP*）复制的 VTV 而不管 *MGMTclas IMMDELAY* 设置为何，因此 StorageTek 强烈建议您不要为复制的 VTV 显式设置 *MGMTclas*

IMMDELAY 策略。如果执行了该操作，VTCS 将满足显式立即迁移请求，立即从最先能够执行迁移的 VTSS（即，存在驻留的 VTV 副本和可用的 RTD 来满足迁移的第一个 VTSS）迁移受影响的 VTV。因此，设置显式 *MGMTclas IMMDELAY* 策略是多余操作，可能会妨碍最佳的 VTV 复制和迁移。

另请注意，复制后的立即迁移 (*KEEP*) 与自动迁移不同。也就是说，在隐式立即迁移期间，不会从任一 VTSS 中删除 VTV 来管理 DBU。相反，VTV 只是通过迁移从接收 VTSS“预先暂存”到 MVC，从而使两个 VTSS 缓冲区的内容都保持不变。为了在 VTSS 群集中进行空间管理，VTCS 将根据任一 VTSS 的空间管理/迁移周期自动迁移 VTV。如果接收 VTSS 的容量大于或等于发送 VTSS 的容量，发送 VTSS 上的自动迁移将从两个 VTSS 上删除复制 VTV。如果接收 VTSS 的容量小于发送 VTSS 的容量，自动迁移可能会在接收 VTSS 上启动。在这种情况下，自动迁移只从接收 VTSS 上删除复制 VTV，而使发送 VTSS 上的副本仍驻留。

- 请注意，要在卸载而非撤回后确定数据的复制要求。撤回 VTV 不会引起复制，所以 *demand recall*、*MVCdrain* 和 *reclaim* 不会引起复制。但是，如果撤回了 VTV 并将其挂载在 VTD 上，则卸载时会将其复制到辅助或对等 VTSS。
- 群集可以在四种运行模式中的每种模式下支持不同的工作负荷。例如，只有全功能群集可以支持活动复制，但是在降级主模式下，您可以将辅助 VTSS 的 VTD 变为联机到 MVS 来接管工作负荷。您可以使用 *Query* 显示群集、群集链路、VTV 复制和 VTSS 状态。您可以使用 *VARY VTSS* 更改 VTSS 状态，使用 *VARY CLink* 更改 CLINK 状态。

VTSS 调节工作原理

- 每当群集 VTSS 对恢复全功能状态时，VTCS 都会调节两个 VTSS 的内容。这发生在 VTCS 初始化期间或者当一个 VTSS 联机且其伙伴 VTSS 也联机时。
- 调节包括删除或迁移并删除 VTV（或者复制 VTV—如果之前未成功完成该操作）。也就是说，调节 VTSS 内容并不涉及撤回。

例如，在 VTV 驻留在接收方而非发送方 VTSS 的单向群集中，VTCS 将从接收方删除 VTV（在确保已创建需要的所有 MVC 副本后）。这样可避免撤回到发送方。

同样，在 VTV 驻留在发送方而非接收方 VTSS 的单向群集中，VTCS 会将 VTV 复制到接收方而非从 MVC 撤回它。

- 调节过程假定，如果复制的 VTV 驻留在发送方 VTSS 上，则它是有效的副本。如果接收方上的副本不同，VTCS 会将其删除。
- 为了在双向群集中保持一致的调节操作，将 VTV 当前所在或上次驻留在的 VTSS（由 CDS VTV 记录指明）视为发送方 VTSS。调节处理与上述单向群集的相同。

单向和双向群集

由两个 VTSS 组成的群集可以采用以下任一配置：

- 单向，其中一个 VTSS 是主 VTSS，另一个 VTSS 是辅助 VTSS。有关更多信息，请参见“[单向群集 VTSS](#)”。
- 双向，其中两个 VTSS 是对等 VTSS，在任一方向上进行对等复制。有关更多信息，请参见“[双向群集 VTSS](#)”。

单向群集

如图 6.1 “单向群集 VTSS” 中所示，在单向群集中，仅从主 VTSS 到辅助 VTSS 进行复制。

图 6.1. 单向群集 VTSS



单向 VTSS 群集工作原理

- 辅助 VTSS 既可以从主 VTSS 接收复制的 VTV，也可以通过任意标准路由方法（例如 *TAPEREQ*）接收非复制生产工作负荷。您需要使辅助 VTSS 中的 VTD 变为联机到 MVS，以使辅助 VTSS 可以接受生产工作。您不能使 CLINK 端接使用的 VTD 地址变为联机到 MVS，如“[群集 VTSS 配置的工作原理](#)”中所述。
- 会将启用了复制的 VTV 分配给联机主 VTSS，除非没有可用的联机主 VTSS；在这种情况下，会将 VTV 分配给联机辅助 VTSS。如果没有可用的联机辅助 VTSS，会将 VTV 分配给非群集 VTSS。可以将未启用复制的 VTV 分配给任何联机 VTSS，包括全功能群集中的辅助 VTSS。
- 卸载时，驻留在全功能群集上的启用了复制的 VTV 将排队等待复制到辅助 VTSS。如果从不属于全功能群集的 VTSS 中的 VTD 中卸载了启用了复制的 VTV，则 VTV 将排队等待立即迁移。

辅助 VTSS 从主 VTSS 接收复制的 VTV 后，将立即迁移 VTV（使用 *KEEP* 选项），而不管该 VTV 的立即迁移管理类设置为何。

- **主和辅助 VTSS** 均可管理所有空间回收。
- 如果使用的是 ESCON 或 FICON 接口，则在主 VTSS 上，将在 *Nearlink* 模式下配置 CLINK CIP/FIP，而在辅助 VTSS 上，将在主机模式下配置 CIP/FIP。

因此，仅为主 VTSS 配置 *CLINK*，如以下示例中所示，其中 VTSS1 是主 VTSS。

```

.
.
.
CLUSTER NAME=CLUSTER1 VTSSs(VTSS1,VTSS2)
CLINK VTSS=VTSS1 CHANIF=0G
CLINK VTSS=VTSS1 CHANIF=0O
CLINK VTSS=VTSS1 CHANIF=1G
    
```

```
CLINK VTSS=VTSS1 CHANIF=10
```

双向群集

如图 6.2 “双向群集 VTSS” 中所示，双向群集需要成对的单向 *CLINK*，以便数据在 *CLINK* 上以相反的方向流动。

图 6.2. 双向群集 VTSS



双向 VTSS 群集工作原理

在双向群集中，在正常操作中，两个 VTSS 均联机到 VTCS，如下所示：

- 在双向群集中，每个对等 VTSS 均可通过标准路由方法（例如 TAPEREQ）接收生产工作。您需要使两个 VTSS 中的 VTD 变为联机到 MVS，以使每个 VTSS 可以接受生产工作。但是请注意，您不能使 *CLINK* 连接使用的 VTD 地址变为联机到 MVS，如“群集 VTSS 配置的工作原理”中所述。
- 在双向群集中，会将启用了复制的 VTV 分配给对等 VTSS 中的任意一个。如果两个对等 VTSS 中的一个脱机或停顿，生产工作负荷可以在剩下的联机 VTSS 上运行。但是，仅当其他任何全功能群集均不可用且不合适时才会将需要复制的 VTV 分配给剩下的 VTSS。在这种情况下，当另一个 VTSS 联机时，将使用 *keep* 立即迁移复制的 VTV，并且复制的 VTV 将排队等待复制。
- 在双向群集中，卸载时，驻留在全功能群集上的启用了复制的 VTV 将排队等待复制到另一个对等 VTSS。如果从不属于全功能群集的 VTSS 中的 VTD 中卸载了启用了复制的 VTV，则 VTV 将排队等待立即迁移。请注意，要在卸载而非撤回后确定数据的复制要求。撤回 VTV 不会引起复制，所以 *demand recall*、*MVCdrain* 和 *reclaim* 不会引起复制。但是，如果撤回了 VTV 并将其挂载在 VTD 上，则卸载时会将其复制到辅助 VTSS，除非您指定 *REPLICAT(CHANGED)*（建议的选项），指定该选项后，仅当数据发生更改时才会再次复制 VTV。
- 两个对等 VTSS 均可管理所有空间回收。
- 如果使用的是 *ESCON* 或 *FICON* 接口：
 - 在每个对等 VTSS 上，将在 *Nearlink* 模式下配置“发送”*CLINK CIP/FIP*，而在主机模式下配置“接收”*CLINK CIP/FIP*。

因此，仅为每个对等 VTSS 配置“发送”*CLINK*，如以下示例中所示，其中 *VSMR1* 和 *VSMR2* 是对等 VTSS。

```

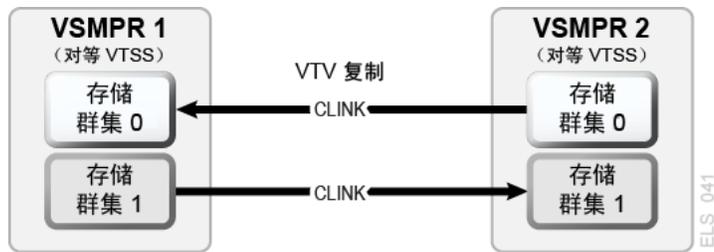
.
.
CLUSTER NAME=CLUSTER1 VTSSs(VSMPR1, VSMPR2)
CLINK VTSS=VSMPR1 CHANIF=00:0
CLINK VTSS=VSMPR1 CHANIF=00:1
CLINK VTSS=VSMPR2 CHANIF=10:0
CLINK VTSS=VSMPR2 CHANIF=10:1
.
.

```

- 每个 CLINK 必须连接到每个 VTSS 上的同一个存储群集（存储群集 0 到存储群集 0 或者存储群集 1 到存储群集 1）。未能以这种方式配置会产生复制、通道和通信错误。

如图 6.3 “双向群集 VTSS 的 ESCON/FICON CLINK” 中的示例中所示，VSMPR1 上的发送（Nearlink 模式）CLINK 端口在存储群集 1 上，它连接到 VSMPR2 上也在存储群集 1 上的接收（主机模式）CLINK 端口。同样，VSMPR2 的存储群集 0 上的发送 CLINK 端口连接到 VSMPR1 的存储群集 0 上的接收 CLINK 端口。

图 6.3. 双向群集 VTSS 的 ESCON/FICON CLINK

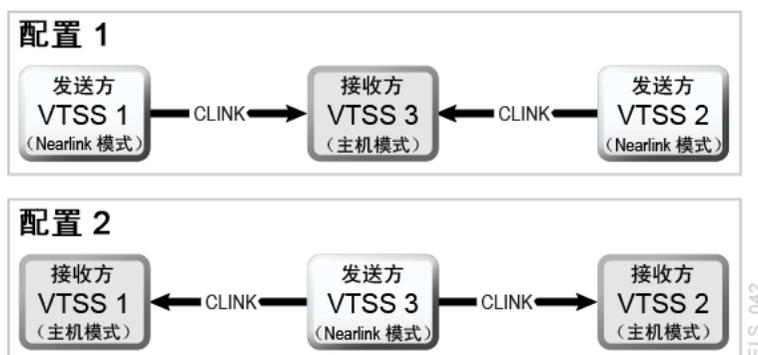


扩展群集

扩展群集 (Extended Clustering, EC) 允许三个或更多 VTSS 由单个 Tapeplex (1 个 CDS) 配置中的 Clink 进行连接。群集是一个高可用性解决方案，设计成在 VTSS 发生中断时工作负荷可以无中断继续运行。群集要求属于一个群集的所有 VTSS 子系统均可访问由该群集中的任何一个 VTSS 子系统生成的所有 MVC。如果群集中的一个 VTSS 连接到一个远程 Tapeplex (CTR)，则群集中的所有 VTSS 系统都必须都连接到同一个 Tapeplex 才能保持 HA 功能。

借助扩展群集，您可以配置一个 VSM 使其 Clink 连接到多个 VTSS，并且 Clink 连接数仅受可用物理连接数的限制。需要 D02.07.00.00 或更高级别的微代码。所有可用的群集和复制规则都适用于 EC。所有扩展群集配置都基于图 6.4 “基本扩展群集配置” 中显示的两种基本单向配置构建而成。

图 6.4. 基本扩展群集配置



同步或异步复制

您可以选择：根据站点的策略，您可以同步或异步复制。

实施同步复制

注意：

如果使用同步复制，复制虚拟卷所需的时间将推迟创建具有同步复制策略的数据的任何作业的完成。

1. 确保您的系统满足表 6.2 “同步复制要求” 中所述的同步复制要求。
2. 在所有 HSC/VTCS 系统关闭的情况下，使用 `CONFIG GLOBAL` 启用同步复制：

```
CONFIG GLOBAL SYNCHREP=YES
```

3. 确保根据需要设置 `CONFIG GLOBAL REPLICAT` 参数：

ALWAYS

每次卸载 VTV 时都将复制请求添加到 VTCS 复制队列中，而不管 VTV 在挂载时是否发生了更改（默认值）。

CHANGED

当 VTV 符合以下条件时，将复制请求添加到 VTCS 复制队列中：

- 挂载时发生了更改或者
- 挂载时为只读，但是存在的 VTV 的 MVC 副本少于预期数量。

4. 在所需的 `MGMTClass` 语句中指定同步复制：

```
MGMT (name) ..... REP(YES_SYNC)
```

利用作业监视实施异步复制

您可能会选择使用异步复制，但是可能也想要知道已成功完成复制。在以下过程中，您将使用 DRMONitr 实用程序进行监视以暂停关联的 MVS 作业，直到成功完成复制。

1. 确保您的系统满足表 6.2 “同步复制要求” 中所述的同步复制要求。
2. 在所有 HSC/VTCS 系统关闭的情况下，使用 `CONFIG GLOBAL` 启用异步复制：

```
CONFIG GLOBAL SYNCHREP=NO
```

3. 确保根据需要设置 `CONFIG GLOBAL REPLICAT` 参数：

ALWAYS

每次卸载 VTV 时都将复制请求添加到 VTCS 复制队列中，而不管 VTV 在挂载时是否发生了更改（默认值）。

CHANGED

当 VTV 符合以下条件时，将复制请求添加到 VTCS 复制队列中：

- 挂载时发生了更改或者
- 挂载时为只读，但是存在的 VTV 的 MVC 副本少于预期数量。

4. 在所需的 `MGMTclass` 语句中指定异步复制：

```
MGMT (mgmtname) ..... REP(YES)
```

5. 创建 JCL 来监视异步复制。

为此，使用 `DRMONitr` 实用程序监视复制。`DRMONitr` 会导致关联的 MVS 作业暂停，直到成功完成复制。例如：

```
//MONITOR EXEC PGM=SLUADMIN, PARM='MIXED'
//STEPLIB DD DSN=hlq.SEALINK, DISP=SHR
/* If HSC IS NOT OR MAY NOT BE ACTIVE, INCLUDE THE
/* FOLLOWING:
//SLS_CNTL DD DSN=primary.cds.name, DISP=SHR
//SLS_CNTL2 DD DSN=secondary.cds.name, DISP=SHR
//SLS_STBY DD DSN=standby.cds.name, DISP=SHR
//SLS_PARM DD DSN=hlq.PARMLIB(BKPCNTL), DISP=SHR
//SLS_PARM2 DD DSN=hlq.PARMLIB(BKPCNTL2), DISP=SHR
//SLS_PARM3 DD DSN=hlq.PARMLIB(BKPSTBY), DISP=SHR
//SYSIN DD UNIT=SYSDA, SPACE=(TRK,1)
/* THE FOLLOWING IS USED BY THE SNAPSHOT UTILITY:
//SYSPRINT DD SYSOUT=*
//SLS_PRINT DD SYSOUT=*
//SLSIN DD *
DRMON MGMT(mgmtname) REPL MAXAGE(24) TIMEOUT(120)
```

在此示例中，`DRMON` 实用程序监视指定管理类的复制。此外，仅监视在过去 24 小时内更新的 VTV，并且使 `DRMON` 在 120 分钟后超时。

使用 TCP/IP 连接形成群集

借助 VTSS 本机 IP 连接功能，您可以使用 TCP/IP 协议“聚集”（连接）两个或多个 VTSS 来复制 VTV。对于本机 IP 群集，每个 VTSS 都有以太网端口，用于连接到 TCP/IP 网络。以前，您仅限于使用 ESCON 或 FICON 连接进行复制。将 TCP/IP 用于 CLINK 可提供优于 ESCON 或 FICON 协议的更高复制性能，如果需要，还能将现有的 ESCON 或 FICON 端口专用于 RTD 和主机连接，支持的连接如下：

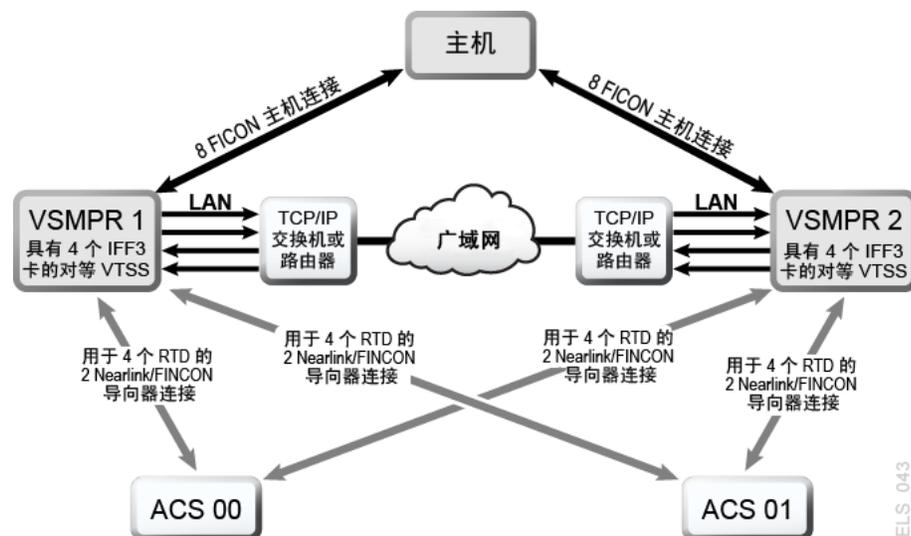
- VSM5 到 VSM5
- VSM5 到 VSM6
- VSM6 到 VSM6

本节仅介绍本机 IP 的 VTCS 实施。您的 StorageTek 硬件支持人员或其他 QSP 负责 VTSS 方面的配置。

TCP/IP 环境

连接了 TCP/IP 的 CLINK 与连接了 FICON 或 ESCON 通道的 CLINK 起着相同的作用，但是 TCP/IP CLINK 使用 VTSS 上的以太网端口连接而非从 ESCON 或 FICON 端口连接。图 6.5 “包含两个 VSM5 的 TCP/IP 环境”中的示例显示了对等 VSM5，每个 VSM5 包含 4 个带有以太网端口的 IFF3 卡。来自 IFF3 卡上以太网端口的以太网电缆连接到局域网 (Local Area Network, LAN)，每个 VTSS 对应一个 LAN，LAN 通过广域网 (Wide Area Network, WAN) 连接。

图 6.5. 包含两个 VSM5 的 TCP/IP 环境



为 TCP/IP CLINK 配置 VTCS

下面显示了 `CONFIG CLINK` 语句的参数。

CONFIG CLINK 语句

CONFIG CLINK 语句通过以下参数提供两种类型的 VTSS 到 VTSS 连接：

CLINK CHANIF=*nn* 或 *nn:n*

定义用作 CLINK 的 FICON 或 ESCON 端口。

CLINK IPIF=*ci:p*

定义用作 CLINK 的以太网端口。对于 VSM5 和 VSM6, *CONFIG RTD IPIF* 的有效值为 *c*=0 或 1, *i*=A 或 I, *p*=0 到 3。对于 VSM5, 该值必须与 VSM5 "IFF Configuration Status" 屏幕上指定的值匹配。对于 VSM6, 该值必须对每个 VTSS 唯一；且不对应于 VSM6 TCP/IP 端口上的实际值。

注：

CLINK 语句必须包含 *CHANIF* 或 *IPIF* 参数，但不能同时包含两者。

第 7 章 使用 Concurrent Disaster Recovery Test 软件

对于将灾难恢复 (Disaster Recovery, DR) 站点用作或维护为业务连续性计划一部分的客户而言，可能需要在实际灾难发生之前定期验证能否继续正常的生产处理；其他客户则没有选择的余地，必须定期演示业务连续性模型的准备就绪状况，以满足保险要求和/或审计员需求。

通过使用 Concurrent Disaster Recovery Test (CDRT) 工具（如今已成为 StorageTek ELS 软件的集成功能），那些当前使用 StorageTek Streamline 和/或 Nearline（实际硬件）磁带库、VSM（虚拟硬件）以及关联软件 (HSC, VTCS) 的企业无需额外购买硬件和软件，即可验证实际的和虚拟的磁带业务连续性功能。

CDRT 支持并行测试生产主机和应用程序，并支持按生产系统和 DR 测试系统同时访问生产数据。

CDRT 的主要概念包括：

- 通过使用 CDRT，可以对实际硬件和/或虚拟硬件执行 DR 测试。
- CDRT、HSC 和 VTCS 将在 CDS 准备和 DR 实际测试期间以编程方式强制执行特定的功能限制，以尝试确保系统完整性。
- CDRT 将在 DR 测试期间逻辑隔离现有部分生产中的实际和虚拟硬件和磁带卷池。这样允许在并发运行生产工作的同时测试 DR 配置，确保生产数据的完整性，并最大程度地减少磁带卷与硬件资源之间的冲突。
- CDRT 创建生产 CDS 的测试副本。因此，生产 ELS 子系统和 DR 测试 ELS 子系统不会彼此通信。发生在 DR 测试 CDS 中的变化不会反映在生产 CDS 副本中，反之亦然。DR 测试主机仅使用逻辑隔离的硬件。生产主机继续使用所有硬件，但有一个例外，即：DR 主机在 DR 测试期间专门使用所有逻辑隔离的 VTSS。其他资源（例如，RTD、多个虚拟磁带 (Multiple Virtual Cartridge, MVC) 和实际暂存磁带）必须通过为每组主机定义单独的池进行控制。
- 可以仅使用本地资源执行 DR 测试，也可结合使用本地资源和远程资源执行 DR 测试。此外，还支持以下配置：远程站点 + 仅实际硬件和虚拟硬件（或者实际硬件和虚拟硬件以及大型机处理器）。
- DR 测试硬件可包括 ACS、VTSS 和 VLE 的任意组合。
- DR 测试结束后，通常会丢弃 CDS 的测试副本以及从 DR 测试创建的所有数据，并将逻辑隔离的硬件重新部署回正常生产环境。

注：

- 要满足从实际灾难中恢复的要求，DR 测试作业流中的作业不得更新生产环境中创建的任何卷（通过 *DISP=MOD* 或通过覆写这些卷），这一点至关重要。采用此类做法意味着，如果发生实际灾难，这些卷的状态将不可预测。
 - 对于 DR 测试运行，强烈建议在测试开始时将 DR 测试期间可能修改的生产卷复制到新卷，且通过 DR 测试（而不是原始卷）更新复制的卷。此外还应修改 JCL（如有可能），以便灾难发生时所有磁带卷的状态均已知。
-

元数据注意事项

使用 CDRT 成功执行 DR 测试的关键在于：ELS 软件以及实际硬件和虚拟硬件管理的所有磁带卷均具有一致的状态副本。这种在 DR 测试开始时生产主机与 DR 主机之间磁带卷状态的一致性保证了客户应用程序的并行处理。由于 CDS 反映了实际硬件和虚拟硬件中所有磁带卷和资源的状态，因此 CDRT 在创建 CDS 的测试副本时，可以在一定程度上满足您的一致性要求。

但在磁带卷环境中，对于这种磁带卷状态数据（元数据），经常会有一些是在 ELS 子系统以及实际硬件和虚拟硬件外部保留和管理的。通常，磁带卷元数据（即 VOLSER、DSN、失效日期、暂存状态、实际或虚拟的名称等）存储在一个或多个磁带管理目录 (Tape Management Catalog, TMC)、一个或多个 z/OS 目录以及 CDS 中。

您必须协调在 ELS（以及实际硬件和虚拟硬件）外部保留和管理的元数据副本的创建与 CDRT 对 CDS 测试副本的创建。

CDRT 从何处获得其 VTV 数据？

CDRT 从以下一个或多个 DR 站点资源获得其 VTV 数据：

- MVC
- VLE
- VTSS

由于生产 CDS 是可用于 DR 测试的 VTV 信息来源，因此确保暂存同步周期允许 DR 测试中使用的卷在测试开始之前不进入暂存状态，这一点至关重要。此外，StorageTek 还建议您在执行 *DRTEST CREATE* 之前将测试 VTSS 变为脱机。

请注意，如果使用 *ACTMVCGN* 实用程序将 DR 测试 VTSS 和 MVC 的内容设置为 *READONLY*，则在测试期间运行暂存不会影响这些内容。

DR 站点 CDS 副本提供复制时的 VTV 位置，该位置通常是在 MVC 或 VLE 上。但有时，您可能会选择使用测试站点存在的 VTSS 中的 VTV 副本。通常，您可在以下情况下进行此选择：

- 您使用 *SHARE* 关键字定义 DR VTSS，防止更新任何生产 VTV 的内容。

- 您拥有两个群集 VTSS，分别位于生产站点和 DR 站点，且 DR 测试不会修改任何生产 VTV 的内容。
- 您拥有两个群集 VTSS，分别位于生产站点和 DR 站点，且在 DR 测试结束之后您愿意花时间来标识并手动迁移 DR 测试所更新的任何 VTV。

注：

运行 DR 测试之前，您可以使用 *DRMONitr* 实用程序确保关键的 DR 数据达到指定的恢复位置。

测试结束时数据会发生什么情况？

由 DR 测试创建的数据仅在 DR 测试 CDS（测试结束之后被丢弃）和 DR 测试 MVC（与生产 MVC 中的卷隔离）中会有所体现。此外，除非您执行以下操作之一，否则测试结束之后，DR 测试所创建的 VTV 将保留在 DR 测试 VTSS 中：

1. 使用 DR 测试环境中的 *SLUADMIN SCRATCH* 实用程序将所有 VTV 暂存（通过 *MGMTCLAS DELSCR(YES)*）在 DR 测试子池范围内。此选项要求使用 *POOLPARM/VOLPARM* 功能。
2. 通过结合使用 *MIGRATE* 命令与 *DELETE(YES)*，确保 DR 测试修改的所有生产 VTV 均从 DR 测试系统迁移到 DR 测试 MVC。如果未做到这一点，则生产系统将拾取 DR 测试已修改的数据。
3. 测试完成后，使用 StorageTek CSE 或其他 QSP“清理”DR 测试 VTSS，以删除 VTSS 中的所有数据。

注：

如果使用选项 2 或选项 3，则 VTSS 的内容与生产系统不匹配。这是因为返回到生产环境时，VTV 会从 DR 测试 CDS 中缺失。尽管该软件会处理这种情况，但生产性能可能会降级。

管理 CDRT 资源

以下各节介绍了如何管理 CDRT 资源。

卷资源

管理 CDRT 资源的第一步是使用 *POOLPARM/VOLPARM* 实用程序定义系统中的卷。此功能将简化磁带池和磁带卷的整体管理，并为隔离写入 CDRT 配置中的卷提供了一种方法。共享 DR 测试 VTSS 时，需要使用 *POOLPARM/VOLPARM*。对于其他情况，强烈建议也这样做。

注：

SLUADMIN SET VOLPARM 实用程序必须使用生产 CDS 运行，并且必须同时定义生产和 DR 测试池。*SET VOLPARM* 实用程序对 DR 测试 CDS 无效。

暂存子池

暂存子池适用于所有 DR 测试方案。以下语法显示了如何使用 *POOLPARAM/VOLPARAM* 定义来定义生产和 DR 测试暂存子池。通过对生产和 DR 测试中的子池（具有不同的卷范围）使用相同的名称，您无需在运行 DR 测试时更改生产策略，如下例中所示。

```
*   SCRATCH POOLS
POOLPARAM NAME(SCRCP1) TYPE(SCRATCH)
VOLPARAM VOLSER(T11000-T11999) MEDIA(T10000T1) RECTECH(T1AE)
POOLPARAM NAME(SCRCP1) TYPE(SCRATCH) DRTEST
VOLPARAM VOLSER(T12000-T12999) MEDIA(T10000T1) RECTECH(T1AE)
POOLPARAM NAME(SCRVTV1) TYPE(SCRATCH)
VOLPARAM VOLSER(V1000-V1999) MEDIA(VIRTUAL)
POOLPARAM NAME(SCRVTV1) TYPE(SCRATCH) DRTEST
VOLPARAM VOLSER(V2000-V2999) MEDIA(VIRTUAL)
```

使用 *POOLPARAM/VOLPARAM* 实用程序定义暂存子池时，您可以使用 HSC *SLUADMIN SCRATCH* 实用程序将 DR 测试输出卷暂存在 DR 测试环境内。通过与指定 **DELSCR(YES)** 的管理类结合使用，运行暂存实用程序可将 DR 测试创建的 VTV 从 VTSS 中删除。

MVC 资源

MVC 资源可用于除仅含实际磁带和无磁带 VSM 以外的所有 DR 测试方案。以下示例显示了如何使用 *POOLPARAM/VOLPARAM* 定义来定义生产和 DR 测试 MVC 池。

```
*   MVC POOLS
POOLPARAM NAME(MVCP1) TYPE(MVC) MCVCFREE(40) MAXMVC(4) THRESH(60) +
START(70)
VOLPARAM VOLSER(T14000-T14999) MEDIA(T10000T1) RECTECH(T1AE)
POOLPARAM NAME(MVCP1) TYPE(MVC) MCVCFREE(40) MAXMVC(4) THRESH(60) +
START(70) DRTEST
VOLPARAM VOLSER(T13000-T13999) MEDIA(T10000T1) RECTECH(T1AE)
```

通过以下方式保留生产 MVC 的内容，以供 DR 测试方案使用：

- 使用 *ACTMVCGN* 实用程序，将要用作 DR 测试输入的 MVC 设置为 *READONLY*，如下例中所示。

```
//ACTMVCG1 EXEC PGM=SLUADMIN,PARM='MIXED'
//STEPLIB DD DSN=h1q.SEALINK,DISP=SHR
//SLSPRINT DD SYSOUT=*
//* NOTE: MVCMAINT READONLY(ON) STATEMENTS
//SLUSMVON DD DSN=h1q.SLUSMVON,DISP=(NEW,CATLG,DELETE),
// SPACE=(CYL,1)
//* NOTE: MVCMAINT READONLY(OFF) STATEMENTS
//SLUSMVOF DD DSN=h1q.SLUSMVOF,DISP=(NEW,CATLG,DELETE),
SPACE=(CYL,1)
//* NOTE: THE FOLLOWING STEP SELECTS ALL "ACTIVE" MVCS
//* IN ACS 01.
```

```
//SLSIN DD *
ACTMVCGN ACS(01)
/*
//ACTMVCG2 EXEC PGM=SLUADMIN,PARM='MIXED'
//STEPLIB DD DSN=h1q.SEALINK,DISP=SHR
//SLSPRINT DD SYSOUT=*
/* NOTE: EXEC MVCMAINT TO SET READONLY(ON)
```

- 使用 *VTCS CONFIG* 确保所有生产主机中均不运行回收：

```
CONFIG HOST NAME(host) NORECLAM
```

VTSS 资源

VTSS 资源分为两种类型：非共享和共享。

非共享 VTSS 资源

在 VTV 迁移到 MVC 或 VLE 的环境中，VTSS 资源在 DR 测试期间便已隔离。DR 测试系统仅有权访问通过 *DRTEST PRIMEPRD/CREATE* 实用程序定义的 VTSS，且这些 VTSS 必须在生产环境中保持脱机状态。如果 DR VTSS 在生产环境中处于联机状态，则 *DRTEST START* 命令将被拒绝。

在某些情况下，生产站点和测试站点中可能同时存在两个同名的 VTSS。在这种情况下，*DRTEST PRIMEPRD/CREATE* 实用程序的 *SPARE* 参数将指定 DR 测试 VTSS 与生产 VTSS 具有相同的名称，但实际上二者却并非同一设备，从而允许生产设备在测试期间保持联机状态。对于其他任何情况，请勿使用 *SPARE* 参数！

CDRT 中的群集 VTSS

"方案 4: 生产站点和 DR 测试站点的群集 VTSS"使用群集 VTSS 将数据传送给 DR 站点的 VTSS。在此方案中，群集在 DR 测试期间不起作用，这是因为 DR 测试 VTSS 在生产环境中处于脱机状态。

如果在 DR 测试期间以任何方式修改生产 VTV，则您必须确保在生产环境中将测试 VTSS 变回联机状态之前，将修改后的 VTV 从 VTSS 中删除。否则，生产站点可能会拾取修改后的 VTV。要防止出现这种情况，**StorageTek** 强烈建议对 DR 测试进行设计，使测试不更改生产 VTV。

在群集 VTSS 方案中，生产主机可能无法访问测试 VTSS 中的 VTD。要允许从生产到 DR 测试 VTSS 进行 ECAM 通信，请在 *VTCS CONFIG* 中指定以下内容：

```
VTD LOW=xxxx HIGH=xxxx NOVERIFY
```

为 DR 测试准备 VTSS 群集

1. 将 DR 测试 VTSS 更改为停顿状态。例如：

```
VARY VTSS1 QUIESCED
```

其目的在于（妥善）关闭对 VTSS1 的复制，以便您能够将其专用于 DR 测试。

2. 使用 *Display REPLicat* 监视复制操作，直到复制完成为止。此时，复制仍处于活动状态：

```
VTSS  HOST  QDEPTHVTSS0  PRODUCTION    1
```

当您看到以下内容时，意味着复制已完成：

```
VTSS  HOST  QDEPTH
VTSS0  PRODUCTION  0
```

3. 通过使用 *Display CLINK* 检查 CLINK 状态，交叉检查复制是否已完成。此时，CLINK 仍处于活动状态：

```
VTSS  CLINK  STATUS USAGE      HOST
VTSS0  7     ONLINE REPLICATING PRODUCTION
VTSS0  8     ONLINE REPLICATING PRODUCTION
You know the CLINK is no longer active when you see this:
VTSS  CLINK  STATUS USAGE      HOST
VTSS0  7     ONLINE FREE
VTSS0  7     ONLINE FREE
```

4. 将 DR 测试 VTSS 变为脱机：

```
VARY VTSS1 OFFLINE
```

在 DR 测试前后管理 VTSS 内容

在开始 DR 测试之前，您必须：

1. 计划测试，以便测试输出不会在无意中进入生产环境。
2. 准备测试，以便 DR 测试 CDS 与 DR VTSS 内容相匹配。

此外，**StorageTek** 强烈建议在测试结束之后尽快清理所有测试站点 VTSS。这种做法可确保下次测试开始时测试站点 VTSS 为空，任何 DR 测试数据均未保留在返回到生产环境的 VTSS 中。

您可通过执行以下任一操作清理测试 VTSS：

1. 禁止 DR 测试对生产 VTV 执行任何更新（覆写或附加），并使用 *POOLPARM/VOLPARM* 定义 DR 测试暂存子池。通过使用这种方法，您可以运行暂存实用程序

将所有 VTV 暂存在 DR 测试范围内。如果管理类指定 *DELSCR(YES)*，这些 VTV 会自动从 VTSS 中删除。

2. 如果 DR 测试可以更新生产 VTV，则将 VTSS 返回到生产环境之前，您必须确保用于 DR 测试输出的所有 VTSS 均已清空。这一点可通过以下两种方式实现：在 DR 测试环境中迁移到零；或者使用 StorageTek CSE 或其他 QSP“清理”VTSS。

运行 *DRTEST CREATE* 实用程序创建 DR 测试 CDS 时，VTSS 内容的元数据将传播到测试环境。DR 测试 VTSS 中的 VTV 可用于 DR 测试环境。

如果 DR 测试 VTSS 定义为备用，则物理 DR VTSS 的内容与 CDS 元数据（引用备用 VTSS 的生产实例）不匹配。要消除差异，请在创建 DR 测试 CDS 之前，将备用 VTSS 的生产实例迁移到生产环境中的 0。您可以运行 *VTCS VTVRPT OPTION(UNAVAIL)*，确保所有 VTV 均已迁移且可用于其他 VTSS。如果未执行此步骤，则 DR 测试尝试访问备用 VTSS 上的 VTV 时，将针对 VT 挂载发出 SLS6680E 消息。

共享 VTSS 资源

如果磁带环境不包括实际磁带或 VLE MVC，则您可以在共享 VTSS 环境中运行 CDRT。*DRTEST PRIMEPRD/CREATE SHARE* 参数指定测试期间在生产环境和 DR 测试环境之间共享 VTSS。此环境强制执行以下限制：

1. *DRTEST CREATE* 也无法定义任何 DRACS 或 STORMNGR 资源。即：数据无法从共享的 VTSS 迁移到外部介质。
2. 生产系统必须使用 *POOLPARM/VOLPARM* 工具定义卷资源。
3. 在 DR 测试环境中挂载非 *DRTEST* 子池 VTV 时，将强制使 VTV 变为只读。即：不允许 DR 测试覆写或附加到任何生产 VTV。

ACS 资源

在非无磁带的 VSM 环境中，ACS 资源用于仅含实际磁带或包含虚拟磁带、但不含 VLE 的 DR 测试方案。CDRT 的 ACS 资源是在 *DRTEST PRIMEPRD/CREATE DRACS* 参数中指定的。

DR 测试 ACS 限制

运行 *DRTEST CREATE* 实用程序之前，您必须发出 *HSC CAPPREF* 命令以将 CAP 设置为手动。该软件确保只要测试有效，CAP 便会始终保持该状态。输入 *DRTEST START* 命令之后，生产环境和 DR 测试环境中将对 DR 测试 ACS 强制执行限制，确保硬件和相关 CDS 之间的一致性，从而允许这两个站点访问数据。如果 DR 测试 ACS 中的 CAP 在生产环境中处于自动模式，则 *DRTEST START* 命令会被拒绝。

该软件将自动执行以下限制。

DR 生产主机要求

在活动的 DR 测试期间，DR ACS 的 DR 生产主机要求包括：

- CAP 必须保留在手动模式。
- 无论 MNTD 命令中的设置如何，*FLOAT(OFF)* 和 *EJCTAUTO(OFF)* 均自动执行。
- 您不能对 DR 测试 ACS 运行弹出、移动、编辑和暂存实用程序。

DR 测试主机要求

在 DR 测试期间，DR 测试主机要求包括：

- 非 DR 测试 ACS 不能变为联机状态。
- CAP 必须保留在手动模式。
- *FLOAT(OFF)* 和 *EJCTAUTO(OFF)* 将自动执行，且 MNTD 命令中的其他值无效。
- 您不能对 DR 测试 ACS 运行弹出、移动、编辑和暂存实用程序。
- 只有对于那些使用 *POOLPARAM/VOLPARAM* 定义为 DR 测试暂存子池卷的卷，才允许暂存更新。

VLE 资源

您可以使用 DRTEST *PRIMEPRD* 和 *CREATE* 命令的 *STORMNGR* 参数定义 VLE 资源。通常，DR 测试资源是 ACS 或 VLE（尽管对使用二者并无限制）。

与物理 MVC 一样，VLE VMVC 也是使用 *POOLPARAM/VOLPARAM* 工具定义的，同时保留了一定范围的 VMVC 供 DR 测试使用。DR 测试还对生产 VMVC 具有读取访问权限，如下例中所示。

```
//ACTMVCG1 EXEC PGM=SLUADMIN, PARM='MIXED'
//STEPLIB DD DSN=h1q.SEALINK, DISP=SHR
//SLSPRINT DD SYSOUT=*
/* NOTE: MVCMAINT READONLY(ON) STATEMENTS
//SLUSMVON DD DSN=h1q.SLUSMVON, DISP=(NEW, CATLG, DELETE),
// SPACE=(CYL, 1)
/* NOTE: MVCMAINT READONLY(OFF) STATEMENTS
//SLUSMVOF DD DSN=h1q.SLUSMVOF, DISP=(NEW, CATLG, DELETE),
SPACE=(CYL, 1)
/* NOTE: THE FOLLOWING STEP SELECTS ALL "ACTIVE" MVCS
/* IN VLE1 AND MVCPool MVCP1.
//SLSIN DD *
ACTMVCGN STORMNGR=VLE1, MVCP=MVCP1
/*
//ACTMVCG2 EXEC PGM=SLUADMIN, PARM='MIXED'
//STEPLIB DD DSN=h1q.SEALINK, DISP=SHR
//SLSPRINT DD SYSOUT=*
/* NOTE: EXEC MVCMAINT TO SET READONLY(ON)
//SLSIN DD DSN=h1q.SLUSMVON, DISP=SHR
```

VTCS 策略

为了最大程度地减少生产环境和 DR 测试环境之间的差异，CDRT 允许您定义各种策略，有效管理在生产环境和 DR 测试环境中具有相同名称、但策略定义不同的 VTC。

请注意，生产站点和测试站点可以通过使用 *DRTEST DRVTTSS SHARE* 参数共享 VTSS。另请注意，对于共享 VTSS 或 DR 测试，无需再使用 VLE 指定虚拟 DR ACS。指定共享 VTSS 时，具有以下限制：

- 共享 DR 测试 VTSS 与生产站点或 DR 测试站点之间不得具有活动的 RTD 连接。
- CDS 必须包含 *VOLPARM* 定义，才能定义 DR 测试暂存子池。
- 生产 VTV（不位于 DR 测试子池中）在挂载后设置为只读，无法进行修改。

为非共享 VTSS 定义 MGMTCLAS/STORCLAS

为 DR 测试系统定义管理类时，您通常仅定义输出 VTV 的一个 MVC 副本。要允许在测试结束之后清除 VTSS，建议按照以下示例中所述指定 *DELSCR(YES)*。

```
STOR NAME(LOCAL) ACS(01) MVCPPOOL(MVCP1)
MGMT NAME(CRITICAL) MIGPOL(LOCAL) IMMWAIT(0) DELSCR(YES)
```

为共享 VTSS 定义 MGMTCLAS

为 DR 测试使用共享 VTSS 时，不允许迁移输出副本。您必须指定 *DELSCR(YES)* 允许清除以下测试，如下例中所示。

```
MGMT NAME(CRITICAL) DELSCR(YES)
```

优化对测试资源和生产资源的访问

在 DR 测试期间，建议您强制执行有效过程，优化测试环境和生产环境中的资源访问。具体而言：

- 开始 DR 测试之前，定义用于指定立即迁移到生产和 DR 测试 ACS 的生产管理类，使 VTV 在 DR 测试系统和生产系统均可访问的 MVC 上可用。
- 由于通常只有一个 ACS 可用于 DR 测试站点，因此请定义将指定单一迁移副本的 DR 测试管理类。
- 使用 *POOLPARM/VOLPARM* 工具隔离生产和 DR 测试之间的暂存子池和 MVC 池。
- 如果可能，请确保 DR 测试处理不会更新任何现有 VTV (*DISP=MOD* 或使用 *DISP=OLD* 覆写)。
- 在生产环境中通过运行 *ACTMVCN* 实用程序将活动的 MVC 标记为只读，最大程度地减少迁移到 DR 测试 ACS 的生产作业与在 DR 测试 ACS 中访问 MVC 上 VTV 的 DR 测试作业之间的争用现象。
- 在 DR 测试期间对生产 MVC 禁用 MVC 空间回收（使用 *CONFIG HOST NORECLAM*），保留 DR 测试系统使用的卷的 MVC 内容。

运行 DR 测试

注：

有关此过程中使用的命令和实用程序的更多信息，请参见《*ELS Command, Control Statement, and Utility Reference*》。

要运行 DR 测试，请执行以下操作：

1. 使用 *SLSPARM DD* 中的 *SET VOLPARM* 命令和 *POOLPARM/VOLPARM* 语句，在生产 CDS 中定义卷池。

有关更多信息，请参见“[卷资源](#)”。

2. 确保已正确设置测试资源。

有关更多信息，请参见：

- ["ACS 资源"](#)
- ["VTSS 资源"](#)
- ["VLE 资源"](#)

3. 为 DRTEST 环境创建 *MGMTCLAS/STORCLAS* 语句。

有关更多信息，请参见：

- ["为非共享 VTSS 定义 MGMTCLAS/STORCLAS"](#)
- ["为共享 VTSS 定义 MGMTCLAS"](#)

4. 如果需要，为 DR 测试站点复制 MVS 目录。
5. （可选）为 DR 测试站点复制 TMS 数据库（如果使用 TMS）。
6. 在生产站点运行 DRTEST 实用程序（使用 *PRIMEprd* 关键字），为 DR 测试准备生产 CDS。

有关示例，请参见“[方案 1 的示例 JCL](#)”。

无论运行多少次 DRTEST 迭代，除非配置发生更改，否则仅需在环境中运行 *PRIMEprd* 一次。如果 DR 测试配置以任何方式发生更改，则需重新运行 *PRIMEprd*。另请注意，DR 测试完成后，您不必运行 *DRTEST RESET* 实用程序。尽管标志在生产 CDS 中仍保持设置状态，但只要 DR 测试处于非活动状态，它们便不会影响处理。

7. 在生产系统中，使用 *HSC CAPPREF* 命令将 DR 测试 ACS 中的所有 CAP 设置为手动模式。
8. 在 DR 测试站点，针对生产 CDS 的镜像或备份副本运行 DRTEST 实用程序（使用 *CREATE* 关键字），以便为 DR 测试创建新的 DR 测试 CDS。

每种方案提供一个 *DRTEST CREATE* 示例。

CDS 必须通过使用实用程序中的 DD 语句加以分配。请注意，使用 *NOUPD* 时，仅需 *SLSCNTL DD* 语句，且该语句可以是实际的主 CDS、备份副本或镜像副本。

9. 在生产站点开始 DR 测试，指向您在第 3 步中创建的 DRTEST *MGMTCLAS/STORCLAS* 定义。

例如：

```
/PRIME EXEC PGM=SLUADMIN,PARM='MIXED'
//STEPLIB DD DSN=h1q.SEALINK,DISP=SHR
//SLSIN DD *
```

DRTEST START

注：

您还可以通过从控制台输入 `DRTEST START` 命令来开始测试。

10. 在 DRTEST 客户机主机上启动 SMC 系统。
11. 在 DR 测试系统上启动 SMC/HSC/VTCS 系统（指向您在第 8 步中创建的 CDS）。
12. 验证测试 VTSS 的 VTD 和路径是否处于联机状态。
13. 将 DR 系统中的 DR VTSS 变为联机。
14. 如果适用，请将 DR 系统中的 DR RTD 变为联机。
15. 在 DR 测试站点运行测试。

在 DR 测试期间，以下条件将以编程方式强制执行：

- 生产站点 ACS 从 DR 测试主机断开连接。
- 生产站点 VTSS 在 DR 测试主机中处于脱机状态。
- DR 测试站点不可出现浮动卸载、弹出、移动、暂存更新、审计或暂存重新分配。
- 生产站点不可出现 DR 测试 ACS 的浮动卸载、装入/弹出、移动、审计或暂存重新分配。
- DR 测试 ACS 中的所有 CAP 均处于手动模式。

注：

您可以将卷装入 DR 测试 ACS，但在测试完成之后，必须弹出卷或对单元进行审计，使生产 CDS 与实际磁带库卷同步。

DR 测试完成后进行清理

正如本章开头所述，DR 测试作业流中的作业不得更新生产环境中创建的任何卷，这一点至关重要。

注：

有关 `DRTEST` 命令和 `DRTEST` 实用程序的信息，请参见《*ELS Command, Control Statement, and Utility Reference*》。有关 CDRT 消息的信息，请参见《*ELS Messages and Codes*》。

DR 测试完成后进行清理

注：

对于 ELS 7.1 及更高版本，如果已安装 CDRT Cleanup Enhancement SPE，则不必在 DR 测试之后和恢复正常生产环境之前立即运行此过程。相反，您可以不间断地（例如，在下次 DR 测试之前）运行该过程。

1. 运行 `SLUADMIN SCRAtch` 实用程序，将所有可能的 VTV 暂存到 DRTEST 子池中。

由于您在管理类中设置了 *DELSCR(YES)*，因此测试结束后暂存 VTV 时，它们将从缓冲区中自动删除。

警告：

如果不使用 *SET VOLPARM*，且未设置单独的暂存池，则会面临数据丢失的风险。

2. 如果 DR 测试已修改或可能修改任何生产 VTV，则您还必须执行以下操作，确保生产 VTSS 中没有保留任何 DR 测试数据：
 - 对 DRTEST CDS 运行 VTV 报告，并检查输出，确定测试期间是否已修改生产 VTV 范围中的任何 VTV。

请注意，*VTVRPT COPIES* 目前在 DRT 列中使用 "D" 标记属于 DR 测试副本的 VTV 副本。

 - 如果已修改任何 VTV，则必须执行以下操作之一：
 - 要求根据 VTV 报告迁移修改后的 VTV。
 - 将 DR 测试 VTSS 迁移到 0。
 - 使用 CSE“清理”测试 VTSS。
3. 停止 DR TEST MVS 系统上的 HSC/VTCS 和 SMC。
4. 如果在测试将 MVC 置于 *READONLY* 状态之前已运行 *ACTMVCGN* 实用程序，则使用 *SLUSMVOF DD* 语句的输出作为输入运行 *SLUADMIN*，以重置 *READONLY* 状态。

恢复正常操作

遵循以下过程重新启动操作并停止 DR 测试。

1. 停止 *PRODUCTION* MVS 系统中的 DR 测试。

例如：

```
/STOP EXEC PGM=SLUADMIN, PARM='MIXED'
//STEPLIB DD DSN=h1q.SEALINK, DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
DRTEST STOP
```

另请注意，DR 测试完成后，您不必运行 *DRTEST RESET* 实用程序。尽管标志在生产 CDS 中仍保持设置状态，但只要 DR 测试处于非活动状态，它们便不会影响处理。

2. 如果需要，将 CAP 置于自动模式。

操作方案

本节介绍如何使用 DR 测试软件设置开始和停止 DR 测试的环境。本节包括以下信息：

- "方案 1: 生产和测试站点, 其中每个站点均有 ACS, 且测试站点有备用 VTSS"
- "方案 2: 生产站点和测试站点, 其中每个站点均有 ACS, 且测试站点有 VTSS 接管"
- "方案 3: 生产站点和测试站点, 其中每个站点均有 ACS, 无 VTSS"
- "方案 4: 生产站点和 DR 测试站点的群集 VTSS"
- "方案 5: 生产站点和测试站点, 其中每个站点均有 ACS 和 VLE"
- "方案 6: 生产站点和测试站点, 其中每个站点只有 VLE"
- "方案 7: 生产站点和 DR 测试站点的群集 VTSS (无磁带) "

有关 DRTEST 命令和 DRTEST 实用程序的信息, 请参见《*ELS Command, Control Statement, and Utility Reference*》。有关 CDRT 消息的信息, 请参见《*ELS Messages and Codes*》。

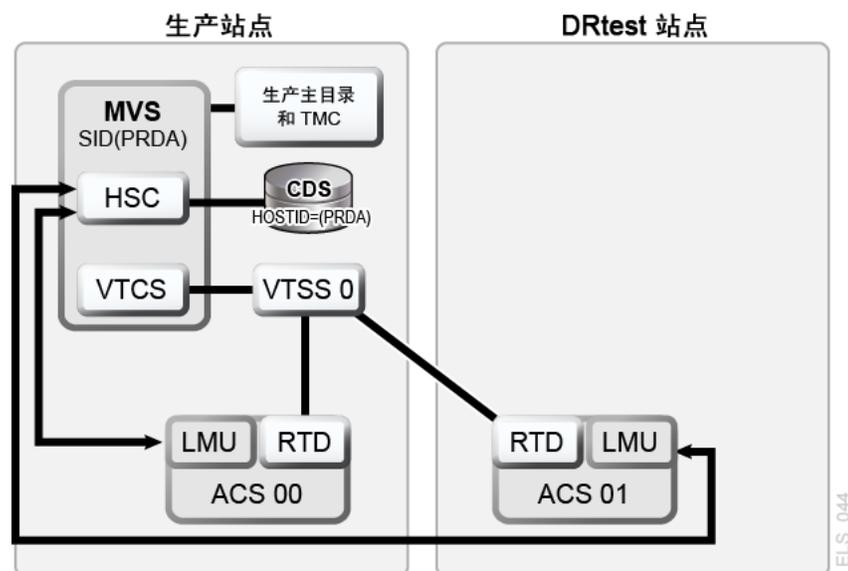
注:

对于所有方案, 在测试之后运行“DR 测试完成后进行清理”中的过程。

方案 1: 生产和测试站点, 其中每个站点均有 ACS, 且测试站点有备用 VTSS

在方案 1 中, 生产站点和测试站点均有一个 ACS, 测试站点的“备用”VTSS 仅用于测试 (不要求迁移或恢复“备用”VTSS 内容)。在正常操作中, 生产站点将在自身站点的 VTSS 中写入并访问 VTV, 且输出 VTV 将始终立即迁移并进行双工处理以分隔 MVC, 每个 ACS 中分别放一个 MVC。图 7.1 “备用 VTSS 配置—运行 DRTEST 实用程序之前”显示了运行 DRTEST 实用程序之前方案 1 的系统。

图 7.1. 备用 VTSS 配置—运行 DRTEST 实用程序之前



方案 1 的示例 JCL

PRIMEPRD 步骤:

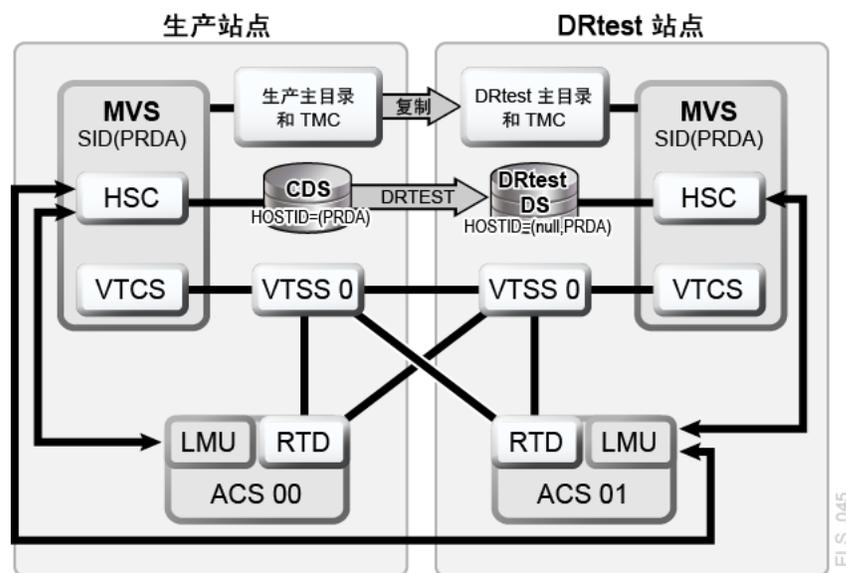
```
//DRTPRIME EXEC PGM=SLUADMIN, PARM='MIXED'
//STEPLIB DD DSN=h1q.SEALINK, DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
DRTEST PRIMEPRD +
DRACS(01) DRVTSS(VTSS0) SPARE HOST(MVS1, MVS2)
```

CREATE 步骤:

```
//DRTCREAT EXEC PGM=SLUADMIN, PARM='MIXED'
//STEPLIB DD DSN=h1q.SEALINK, DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSNEW1 DD DSN=h1q.DRTEST.SLSCNTL, DISP=(NEW, CATLG, DELETE),
// UNIT=SYSDA, SPACE=(CYL, x)
//SLSNEW2 DD DSN=h1q.DRTEST.SLSCNTL2, DISP=(NEW, CATLG, DELETE),
// UNIT=SYSDA, SPACE=(CYL, x)
//SLSNEW3 DD DSN=h1q.DRTEST.SLSSTBY, DISP=(NEW, CATLG, DELETE),
// UNIT=SYSDA, SPACE=(CYL, x)
//SLSIN DD *
DRTEST CREATE NOUPDRD +
DRACS(01) DRVTSS(VTSS0) SPARE HOST(MVS1, MVS2)
```

图 7.2 “备用 VTSS 配置—运行 DRTEST 实用程序之后” 显示了运行 DRTEST 实用程序之后方案 1 的系统（测试站点的备用 VTSS）。

图 7.2. 备用 VTSS 配置—运行 DRTEST 实用程序之后

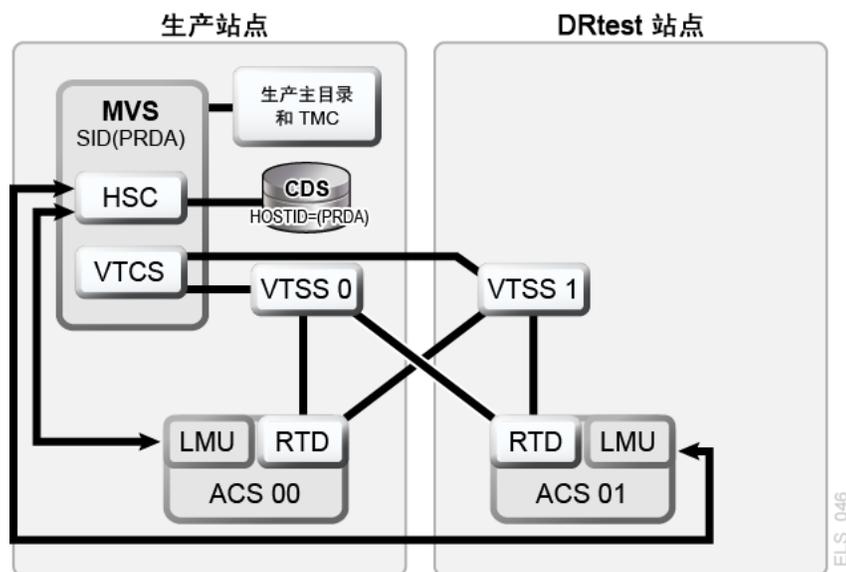


方案 2：生产站点和测试站点，其中每个站点均有 ACS，且测试站点有 VTSS 接管

在方案 2 中，生产站点和测试站点均有一个 ACS，但测试站点没有用于测试的“备用”VTSS。在正常操作中，生产站点在这两个站点的 VTSS 中写入并访问 VTV，且输出 VTV 始终会立即迁移并进行双工处理以分隔 MVC，每个 ACS 中分别放一个 MVC。在此配置中，您必须要求迁移至测试站点的零个、一个或多个 VTSS，并将这些 VTSS 在生产系统中变为脱机，使测试能够接管所需要的 VTSS 资源。此外，测试站点的一个或多个 LPAR 充当移位后的生产系统，与实际生产系统并行运行。这两个 ACS 在生产系统中均处于联机状态。

图 7.3 “VTSS 接管配置—运行 DRTEST 实用程序之前”显示了在运行 DRTEST 实用程序之前方案 2 的系统（测试站点的 VTSS 接管）。

图 7.3. VTSS 接管配置—运行 DRTEST 实用程序之前



方案 2 的示例 JCL

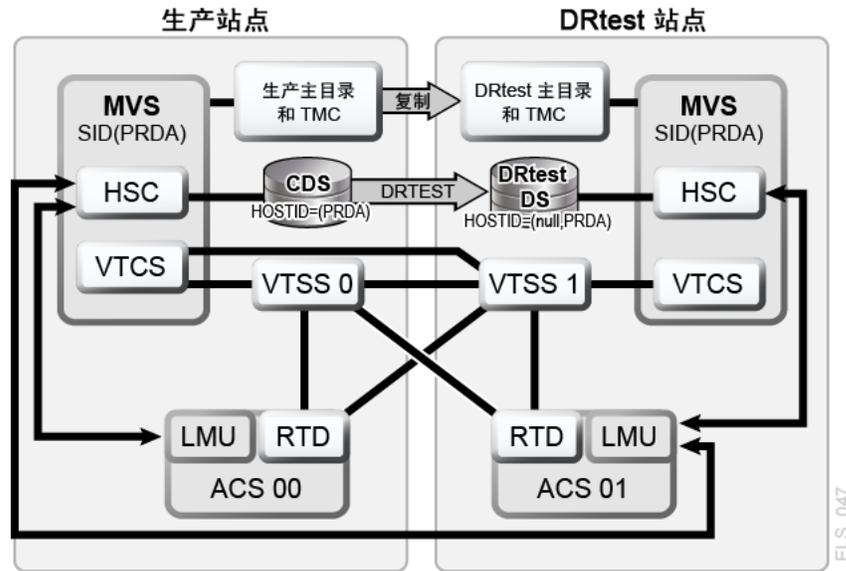
仅限 CREATE 步骤，先前已运行 PRIMEPRD:

```
//DRTCREAT EXEC PGM=SLUADMIN, PARM='MIXED'
//STEPLIB DD DSN=h1q.SEALINK, DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSNEW1 DD DSN=h1q.DRTEST.SLSCNTL, DISP=(NEW, CATLG, DELETE),
// UNIT=SYSDA, SPACE=(CYL, x)
//SLSNEW2 DD DSN=h1q.DRTEST.SLSCNTL2, DISP=(NEW, CATLG, DELETE),
// UNIT=SYSDA, SPACE=(CYL, x)
//SLSNEW3 DD DSN=h1q.DRTEST.SLSSTBY, DISP=(NEW, CATLG, DELETE),
// UNIT=SYSDA, SPACE=(CYL, x)
//SLSIN DD *
DRTEST CREATE NOUPDRD +
```

DRACS(01) DRVTSS(VTSS1) HOST(MVS1,MVS2)

图 7.4 “VTSS 接管配置—运行 DRTEST 实用程序后” 显示了在运行 DRTEST 实用程序之后方案 2 的系统（测试站点的 VTSS 接管）。

图 7.4. VTSS 接管配置—运行 DRTEST 实用程序后



方案 3：生产站点和测试站点，其中每个站点均有 ACS，无 VTSS

在方案 3 中，生产站点和测试站点均有一个 ACS，但测试站点没有用于测试的 VTSS。在正常操作中，生产站点在这两个站点中写入并访问磁带数据。图 7.5 “只读配置—运行 DRTEST 实用程序之前” 显示了在运行 DRTEST 实用程序之前方案 3 的系统（只读配置）。

图 7.5. 只读配置—运行 DRTEST 实用程序之前

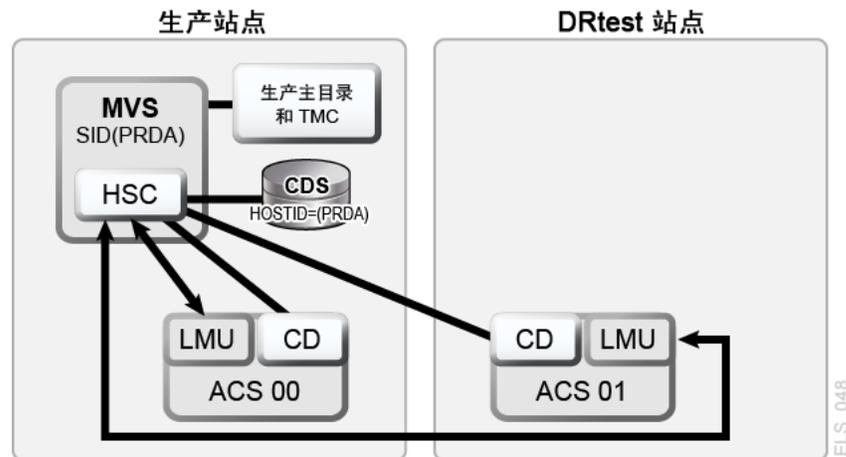
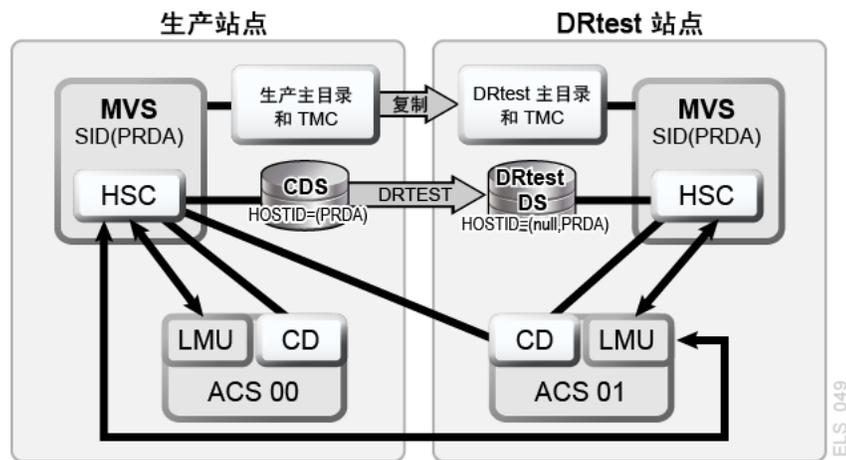


图 7.6 “只读配置—运行 DRTEST 实用程序之后”显示了运行 DRTEST 实用程序之前方案 3 的系统（只读配置）。

图 7.6. 只读配置—运行 DRTEST 实用程序之后



方案 3 的示例 JCL

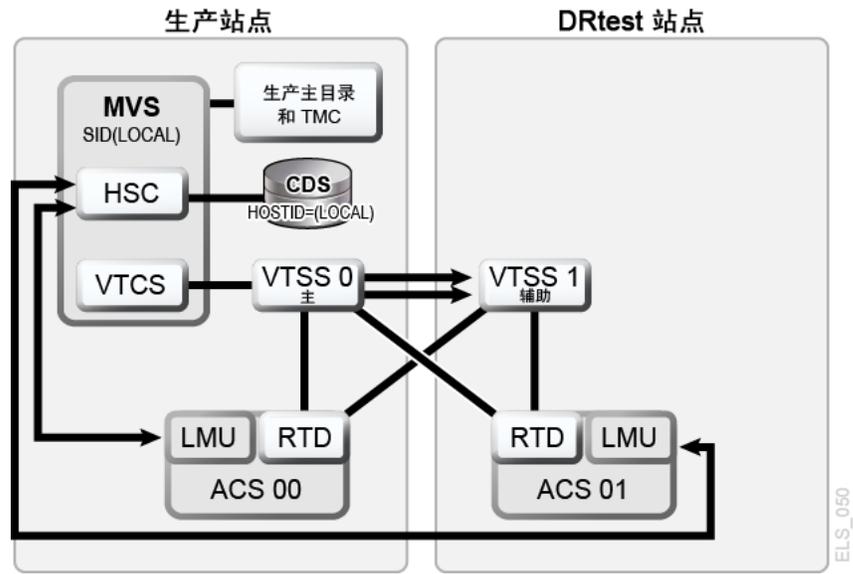
仅限 CREATE 步骤，先前已运行 PRIMEPRD:

```
//DRTCREAT EXEC PGM=SLUADMIN, PARM='MIXED'
//STEPLIB DD DSN=h1q.SEALINK, DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSNEW1 DD DSN=h1q.DRTEST.SLSCNTL, DISP=(NEW, CATLG, DELETE),
// UNIT=SYSDA, SPACE=(CYL, x)
//SLSNEW2 DD DSN=h1q.DRTEST.SLSCNTL2, DISP=(NEW, CATLG, DELETE),
// UNIT=SYSDA, SPACE=(CYL, x)
//SLSNEW3 DD DSN=h1q.DRTEST.SLSSTBY, DISP=(NEW, CATLG, DELETE),
// UNIT=SYSDA, SPACE=(CYL, x)
//SLSIN DD *
DRTEST CREATE NOUPDPRD +
DRACS(01) HOST(MVS1, MVS2)
```

方案 4: 生产站点和 DR 测试站点的群集 VTSS

如图 7.7 “主/辅助群集 VTSS 配置—正常操作”中所示，在正常操作中，方案 4 是用于 DR 的群集 VTSS 配置，其中生产站点和 DR 测试站点交叉连接到生产和 DR 测试 ACS。VTSS0 是生产站点的主配置；VTSS1 是 DR 测试站点的辅助配置。

图 7.7. 主/辅助群集 VTSS 配置—正常操作



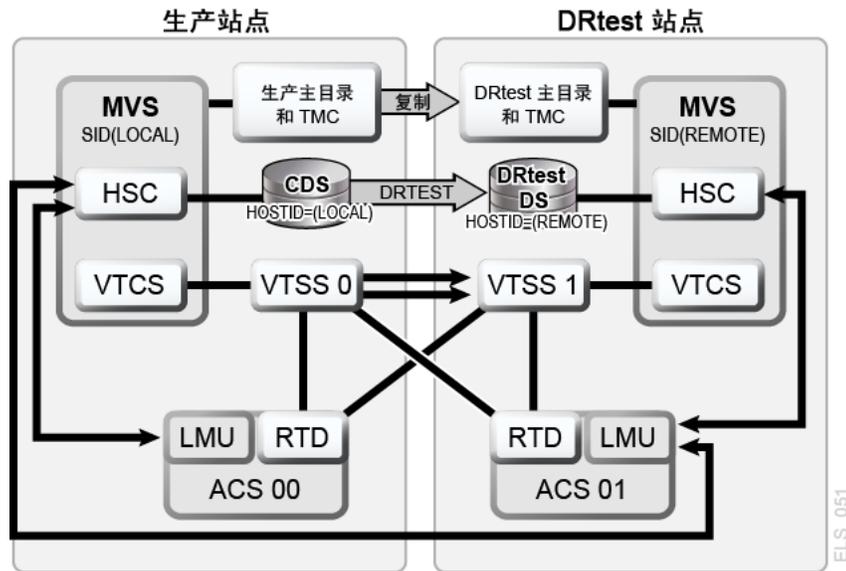
方案 4 的示例 JCL

仅限 CREATE 步骤，先前已运行 PRIMEPRD：

```
//DRTCREAT EXEC PGM=SLUADMIN, PARM='MIXED'
//STEPLIB DD DSN=h1q.SEALINK, DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSNEW1 DD DSN=h1q.DRTEST.SLSCNTL, DISP=(NEW, CATLG, DELETE),
// UNIT=SYSDA, SPACE=(CYL, x)
//SLSNEW2 DD DSN=h1q.DRTEST.SLSCNTL2, DISP=(NEW, CATLG, DELETE),
// UNIT=SYSDA, SPACE=(CYL, x)
//SLSNEW3 DD DSN=h1q.DRTEST.SLSSTBY, DISP=(NEW, CATLG, DELETE),
// UNIT=SYSDA, SPACE=(CYL, x)
//SLSIN DD *
DRTEST CREATE NOUPDRD +
DRACS(01) DRVTSS(VTSS1) SHARE HOST(MVS1, MVS2)
```

如果要使用 DR 测试站点进行测试会怎样？图 7.8 “主/辅助群集 VTSS 配置—测试期间” 显示了 DR 测试期间方案 4 的系统。

图 7.8. 主/辅助群集 VTSS 配置—测试期间

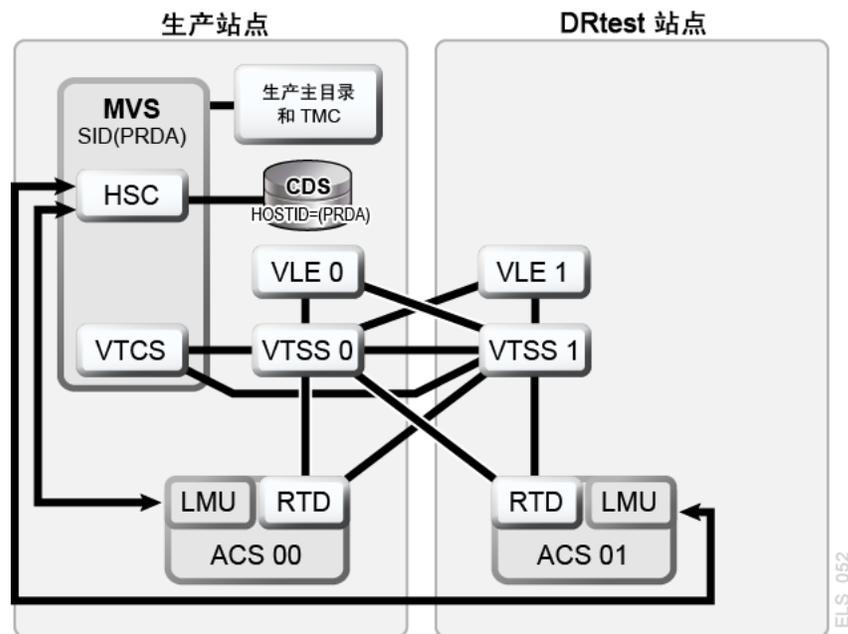


方案 5：生产站点和测试站点，其中每个站点均有 ACS 和 VLE

在方案 5 中，生产站点和测试站点均有一个 ACS，但测试站点没有用于测试的“备用”VTSS。在正常操作中，生产站点在这两个站点的 VTSS 中写入并访问 VTV，且输出 VTV 始终会立即迁移并进行双工处理，一个复制到 ACS 中的 MVC，另一个复制到 VLE 中的 VMVC。在此配置中，您必须要求迁移至测试站点的零个、一个或多个 VTSS，并将这些 VTSS 在生产系统中变为脱机，使测试能够接管所需的 VTSS 资源。此外，测试站点的一个或多个 LPAR 充当移位后的生产系统，与实际生产系统并行运行。ACS 和 VLE 在生产系统中均处于联机状态。

图 7.9 “VLE 和 ACS 配置—运行 DRTEST 实用程序之前”显示了运行 DRTEST 实用程序之前方案 5 的系统。

图 7.9. VLE 和 ACS 配置—运行 DRTEST 实用程序之前



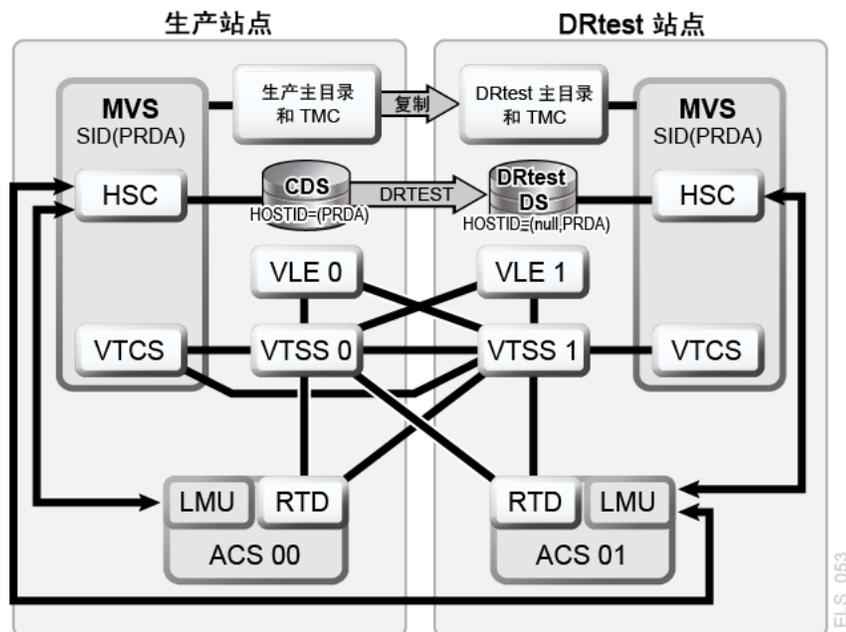
方案 5 的示例 JCL

仅限 CREATE 步骤，先前已运行 PRIMEPRD:

```
//DRTCREAT EXEC PGM=SLUADMIN, PARM='MIXED'
//STEPLIB DD DSN=h1q.SEALINK, DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSNEW1 DD DSN=h1q.DRTEST.SLSCNTL, DISP=(NEW, CATLG, DELETE),
// UNIT=SYSDA, SPACE=(CYL, x)
//SLSNEW2 DD DSN=h1q.DRTEST.SLSCNTL2, DISP=(NEW, CATLG, DELETE),
// UNIT=SYSDA, SPACE=(CYL, x)
//SLSNEW3 DD DSN=h1q.DRTEST.SLSSTBY, DISP=(NEW, CATLG, DELETE),
// UNIT=SYSDA, SPACE=(CYL, x)
//SLSIN DD *
DRTEST CREATE NOUPDPRD +
DRACS(01) DRVTSS(VTSS1) HOST(MVS1, MVS2) STORMNGR(VLE1)
```

图 7.10 “VLE 和 ACS 配置—运行 DRTEST 实用程序之后” 显示了运行 DRTEST 实用程序之后方案 5 的系统。

图 7.10. VLE 和 ACS 配置—运行 DRTEST 实用程序之后



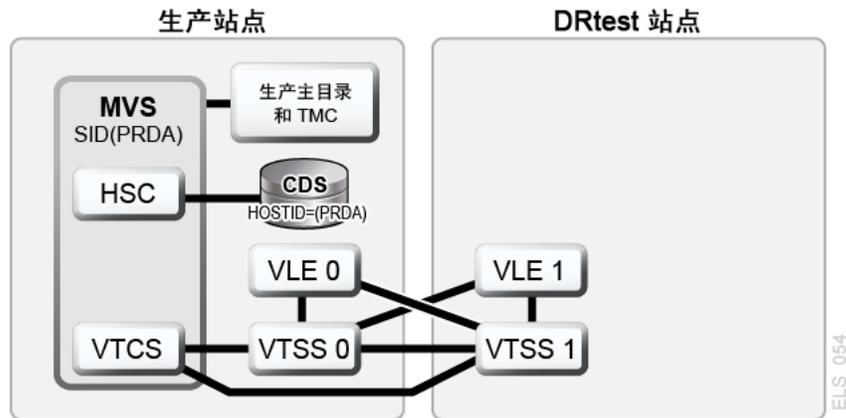
方案 6：生产站点和测试站点，其中每个站点只有 VLE

在方案 6 中，每个站点均有一个 VTSS 与 VLE 相连。测试站点的 VTSS 不是备用 VTSS，它将在生产站点的正常操作过程中使用。输出 VTV 始终会立即迁移并进行双工处理以分隔 VMVC，每个 VLE 中分别放一个 VMVC。

在此配置中，您必须要求迁移至测试站点的零个、一个或多个 VTSS，并将这些 VTSS 在生产系统中变为脱机，使测试能够接管所需的 VTSS 资源。此外，测试站点的一个或多个 LPAR 充当移位后的生产系统，与实际生产系统并行运行。这两个 VLE 在生产系统中均处于联机状态。

图 7.11 “仅 VLE 配置—运行 DRTEST 实用程序之前”显示了运行 DRTEST 实用程序之前方案 6 的系统。

图 7.11. 仅 VLE 配置—运行 DRTEST 实用程序之前



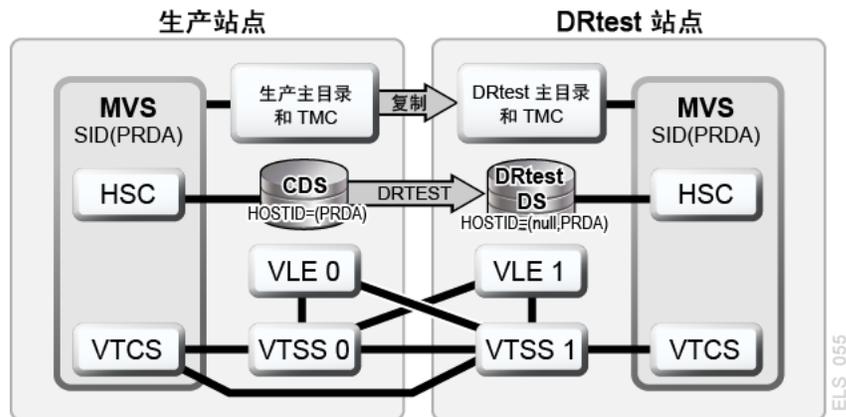
方案 6 的示例 JCL

仅限 CREATE 步骤，先前已运行 PRIMEPRD：

```
//DRTCREAT EXEC PGM=SLUADMIN, PARM='MIXED'
//STEPLIB DD DSN=h1q.SEALINK, DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSNEW1 DD DSN=h1q.DRTEST.SLSCNTL, DISP=(NEW, CATLG, DELETE),
// UNIT=SYSDA, SPACE=(CYL, x)
//SLSNEW2 DD DSN=h1q.DRTEST.SLSCNTL2, DISP=(NEW, CATLG, DELETE),
// UNIT=SYSDA, SPACE=(CYL, x)
//SLSNEW3 DD DSN=h1q.DRTEST.SLSSTBY, DISP=(NEW, CATLG, DELETE),
// UNIT=SYSDA, SPACE=(CYL, x)
//SLSIN DD *
DRTEST CREATE NOUPDPRD +
STORMNGR(VLE1) DRVTSS(VTSS1) HOST(MVS1,MVS2)
```

图 7.12 “仅 VLE 方案—运行 DRTEST 实用程序之后” 显示了运行 DRTEST 实用程序之后方案 6 的系统。

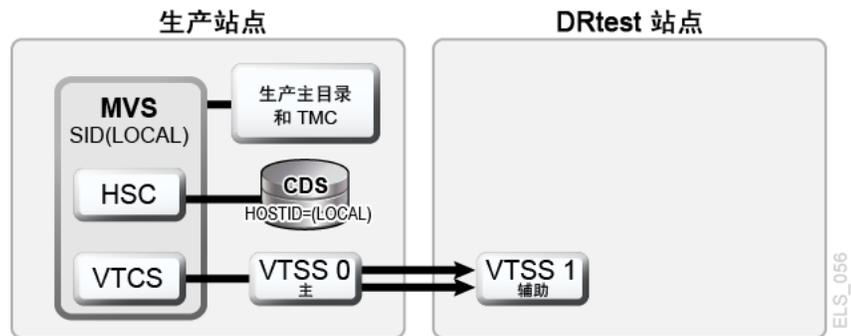
图 7.12. 仅 VLE 方案—运行 DRTEST 实用程序之后



方案 7: 生产站点和 DR 测试站点的群集 VTSS (无磁带)

如图 7.13 “主/辅助群集 VTSS 无磁带配置—运行 DRTEST 实用程序之前” 中所示, 在正常操作中, 方案 7 是用于生产和 DR 测试站点 DR 的群集 VTSS (无磁带) 配置。VTSS0 是生产站点的主配置; VTSS1 是 DR 测试站点的辅助配置。

图 7.13. 主/辅助群集 VTSS 无磁带配置—运行 DRTEST 实用程序之前



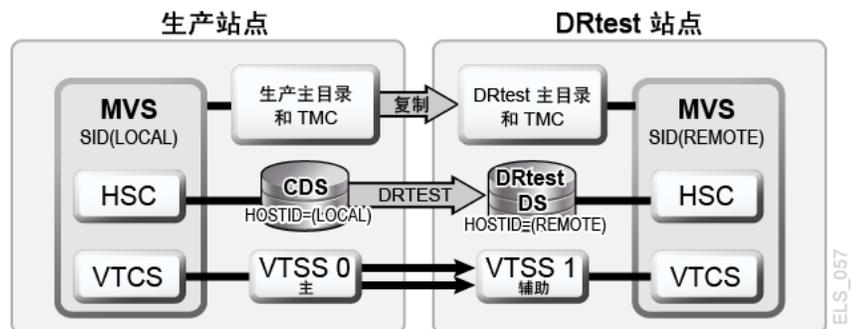
方案 7 的示例 JCL

仅限 CREATE 步骤, 先前已运行 PRIMEPRD:

```
//DRTCREAT EXEC PGM=SLUADMIN, PARM='MIXED'
//STEPLIB DD DSN=h1q.SEALINK, DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSNEW1 DD DSN=h1q.DRTEST.SLSCNTL, DISP=(NEW, CATLG, DELETE),
// UNIT=SYSDA, SPACE=(CYL, x)
//SLSNEW2 DD DSN=h1q.DRTEST.SLSCNTL2, DISP=(NEW, CATLG, DELETE),
// UNIT=SYSDA, SPACE=(CYL, x)
//SLSNEW3 DD DSN=h1q.DRTEST.SLSSTBY, DISP=(NEW, CATLG, DELETE),
// UNIT=SYSDA, SPACE=(CYL, x)
//SLSIN DD *
DRTEST CREATE NOUPDPRD +
DRVTTSS(VTSS1) SHARE HOST(MVS1, MVS2)
```

如果要使用 DR 测试站点进行测试会怎样? 图 7.14 “主/辅助群集 VTSS 无磁带配置—测试期间” 显示了 DR 测试期间方案 7 的系统。

图 7.14. 主/辅助群集 VTSS 无磁带配置—测试期间



第 8 章 在 VSM 环境中创建系统恢复点

正如“[定义恢复点目标 \(Recovery Point Objective, RPO\)](#)”中所述，成功 DR 解决方案的关键要点之一是能够建立系统检查点，确保可以使用一组一致的数据作为 DR 基线。

对于 VSM 环境，有效的 DR 基线具备以下特点：

- 保护指定的 DR 位置中的所有关键业务数据。
- 已捕获元数据（CDS、MVS 目录、TMC）的安全副本。
- 在声明灾难（实际灾难或测试灾难）时保证元数据副本有效。

VTCS 支持通过以下函数创建 DR 基线：

- *DRMONitr* 实用程序监视并确保关键 DR 数据达到其指定的恢复位置。它允许作业流处理停止等待数据达到其目的地。处理完所有数据之后，该实用程序将结束。*DRMONitr* 实用程序可作为作业步骤运行。该作业步骤完成后，系统将保证已处理所有受监视的数据，并在指定的 DR 位置保护其安全。
- *DRCHKPT* 实用程序用于确保通过 CDS 元数据访问的数据在规定期限内保持有效。这可保证 CDS 备份在规定期限内保持有效，因此您可以将 VSM 系统恢复回到 DR 基线。*DRCHKPT* 实用程序在活动的 CDS 中设置日期/时间戳，用于建立可以从中恢复 MVC 内容的恢复点。从此恢复时间点开始，数据内容将在未来一段时间内得到保证。如果没有 *DRCHKPT* 实用程序，则无法使用 CDS 备份恢复回到 DR 基线，这是因为 CDS（MVC 上的 VTV 位置）中的元素可能不再有效。

有关更多信息，请参见《[ELS Command, Control Statement, and Utility Reference](#)》。

另外，还要注意以下几点：

- 对于 VMVC，具有 *EJECT* 参数的 *MVCDRAIN* 会物理删除 VTV。

注意：

如果您使用 *DRCHKPT* 实用程序和/或 *CONFIG GLOBAL PROTECT* 参数来保护 VMVC 的 CDS 备份内容，指定 *MVCDR EJECT* 将验证 CDS 备份的 VMVC 内容。

- 对于 VMVC 和 MVC，不带 *EJECT* 参数的 *MVCDRAIN* 不会删除 VTV，但会更新 CDS 记录以便不在 VMVC/MVC 上显示 VTV。

有关更多信息，请参见《[ELS Command, Control Statement, and Utility Reference](#)》。

检查点示例

下面，我们将讨论以下示例：

- "示例 1：本地 MVC 副本和远程 MVC 副本"
- "示例 2：使用 CONFIG RECLAIM PROTECT"

示例 1：本地 MVC 副本和远程 MVC 副本

在此示例中：

- *DRMONitr* 和 *DRCHKPT* 实用程序可确保 DR 数据达到其恢复位置，且关联的元数据（CDS 备份）可根据需要检索 VTV 数据。
- 本地站点与 VTSS 和 ACS (ACS 00) 一起显示，远程站点仅与 ACS (ACS 01) 一起显示，如图 8.1 “VSM 系统恢复点示例（本地和远程）” 中所示。

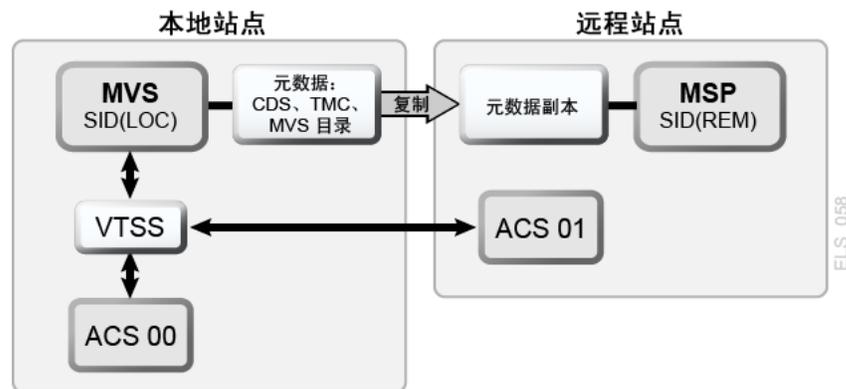
该示例是一个简单的 DR 策略，其中远程站点的关键数据副本和元数据每天均会得到保护。远程 VTV 副本是指定的 DR 副本。

生产作业完成后，将安排以下作业：

- 监视远程副本是否已完成 (*DRMONitr*)。
- 对 CDS 执行检查点操作 (*DRCHKPT*)。
- 备份元数据（CDS、TMC、MVS 目录）并安全保存在远程站点。请注意，元数据备份是 DR 的关键所在，我们假设这些备份已转到“已知”位置，或其位置已妥善记录。

这提供了一个每天同步的 DR 检查点。如果已声明 DR，则磁带环境将恢复回到该检查点，且作业从此已知状态重新运行。

图 8.1. VSM 系统恢复点示例（本地和远程）



要使用图 8.1 “VSM 系统恢复点示例（本地和远程）” 中所示的配置运行此示例，请执行以下操作：

1. 创建以下策略语句。

```
MGMT NAME(DR)  MIGPOL(LOCAL,REMOTE) IMMDELAY(0)
STOR NAME(LOCAL) ACS(00)
STOR NAME(REMOTE) ACS(01)
```

注:

对于有效的 DR 环境，您可能还需要考虑使用 *MIGRSEL* 和 *MIGRVTV* 语句，以便您能够尽早确保 DR 副本的安全。

2. 为了确保在远程位置保护关键数据的安全，需运行以下示例 *DRMONitr* 作业步骤。

```
//MONITOR EXEC PGM=SLUADMIN, PARM='MIXED'
//STEPLIB DD DSN=h1q.SEALINK, DISP=SHR
//SYSIN DD UNIT=SYSDA, SPACE=(TRK,1)
//*
//SYSPRINT DD SYSOUT=*
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
DRMON MGMT(DR) STOR(REMOTE) MAXAGE(24) TIMEOUT(30)
```

在此示例中，*DRMONitr* 实用程序将一直等待，直到 24 小时内的管理类 DR 的所有 VTV 副本均传送到远程 ACS 为止。如果运行时间（或等待时间）超过 30 分钟，则该实用程序将设置为中止。

3. 所有 VTV 副本传送到远程 ACS 之后（用 RC 0 表示），*DRCHKPT* 将运行以设置恢复点，如下例中所示。

```
//CHKPT EXEC PGM=SLUADMIN, PARM='MIXED'
//STEPLIB DD DSN=h1q.SEALINK, DISP=SHR
//SYSPRINT DD SYSOUT=*
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
DRCHKPT SET
```

在此示例中，*DRCHKPT* 实用程序在活动的 CDS 中设置时间戳或恢复点。从此恢复时间点开始，MVC 副本内容将在未来一段时间内得到保证。（例如，在另一个 *CHKPT* 实用程序运行之前。）

4. 在活动的 CDS 中设置恢复点之后，应立即进行 CDS 备份，如下例中所示。

```
//CHKPT EXEC PGM=SLUADMIN, PARM='MIXED'
//STEPLIB DD DSN=h1q.SEALINK, DISP=SHR
//SYSIN DD UNIT=SYSDA, SPACE=(TRK,1)
//*
//SLSCNTL DD DSN=h1q.DBSEPRM, DISP=SHR
```

```
//SLSBKUP DD DSN=h1q.DBSEPRM.BKUP,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
BACKUP OPTION(COPY)
```

进行备份之后，系统将保证 MVC 内容或元数据在未来某个时间点内有效（直到设置了后续恢复点或检查点为止）。

至此，整个过程已完成。如果 DR 声明本地生产站点不再可用，则会出现以下情况之一：

- MVC 和所有其他关键数据（例如，元数据副本）将传送给另一具有本地生产站点镜像的设备。

或者

- 远程位置将构建本地生产站点的副本。

系统将恢复元数据（CDS、TMC、MVS 目录）。重新启动磁带环境后，所有一切均可从 DR 同步点开始继续（前滚）。

示例 2：使用 CONFIG RECLAIM PROTECT

在此示例中，CDS 每 24 小时备份一次。CDS 备份中的 MVC 内容或 CDS 元数据必须保持有效，直到进行后续 CDS 备份为止。

此示例显示了设置为 28 小时的 MVC 保护。有关 *CONFIG RECLAIM PROTECT* 参数的更多信息，请参见《*ELS Command, Control Statement, and Utility Reference*》。

1. 设置 *CONFIG GLOBAL PROTECT = 28*。
2. 第 1 天备份 CDS。
 - 此备份后排空/回收的任何 MVC 在 28 小时内无法覆写。
 - 现在，在下次 CDS 备份之前，第 1 天的 CDS 备份是恢复点。
3. 第 2 天备份 CDS。
 - 此备份后排空/回收的任何 MVC 在 28 小时内无法覆写。
 - 现在，在下次 CDS 备份之前，第 2 天的 CDS 备份是恢复点。

第 9 章 将 VLE 用于灾难恢复

将 VLE (Virtual Library Extension) 用作灾难恢复解决方案，这为执行 DR 测试以及从业务中断事件恢复提供了一种简化且无中断的方法。

系统像管理磁带库 (ACS) 一样管理 VLE。但是，由于 VLE 使用磁盘存储（而非磁带存储），且在其内容中维护 VTV 的内部库存，因此它具有实际磁带库所无法提供的功能：

- VLE 是一款“无磁带”解决方案，可避免出现介质管理问题。
- 数据通过使用 IP 发送给 VLE，无需通道扩展。
- 与挂载和读取 MVC 磁带相比，VLE 可以通过使用其内部数据库在短短几秒钟内执行 MVC 审计。

本章介绍如何在简单的双站点环境中使用 VLE。但值得注意的是，该解决方案支持任意数量的站点，且每个站点支持任意数量的 VLE。此外，其中某一站点可以是仅 DR 站点，仅在 DR 测试或声明的灾难期间运行 MVS LPAR。

接下来的过程使用以下环境：其中具有 SITE1 和 SITE2 这两个站点。每个站点都有一个 VSM 和一个 VLE。在此示例中，SITE2 称为“仅 DR 站点”，但 SITE2 也可以是定义为 SITE1 镜像的生产站点。

注：

SITE2 的 VLE 缓冲区大小必须足够大，足以保存迁移的生产数据以及 DR 测试期间创建的数据。

正常的生产模式

在正常的生产过程中，策略是在 SITE1 中定义的，旨在将数据的一个副本迁移到 SITE1 的本地 VLE，并将另一个副本迁移到 SITE2 的远程 VLE。如果需要，您还可以创建其他副本，其中包括位于另一个 VLE 中的两个副本以及磁带副本。

下面显示了在 SITE1 定义的策略示例：

SMC 定义用于将 VLEPROD 的 *MGMTCLAS* 名称分配给带有高级限定词 "PAYROLL" 的数据集。

```
POLICY NAME(VLEPOL) MEDIA(VIRTUAL) MGMT(VLEMGMT) + SUBP(VIRTSCR)
TAPEREQ DSN(PAYROLL.*) POLICY(VLEPOL)
```

HSC *POOLPARM/VOLPARM* 定义用于定义生产卷：

```
POOLPARM TYPE(MVC) NAME(LOCAL)
VOLPARM VOLSER(VLL000-VLL099)
POOLPARM TYPE(MVC) NAME(VAULT1)
VOLPARM VOLSER(VLV000-VLV099)
POOLPARM TYPE(SCRATCH) NAME(VIRTSCR)
VOLPARM VOLSER(V000000-V99999) MEDIA(VIRTUAL)
```

注：

请注意，池 LOCAL 和 VAULT1 中的 MVC 分别是 SITE1 和 SITE2 VLE 中的 VMVC（虚拟 MVC），没有与之相关联的介质类型。

VTCS *STORCLAS* 和 *MGMTCLAS* 用于定义 VTCS 策略：

```
STOR NAME(VLE1) STORMNGR(SITE1VLE) MVCPOOL(LOCAL)
STOR NAME(VLE2) STORMNGR(SITE2VLE) MVCPOOL(VAULT1)
MGMT NAME(VLEMGMT) DELSCR(YES) MIGPOL(VLE1, VLE2)
```

使用以高级限定词 "PAYROLL" 开头的数据集运行作业时，SMC 使用 *TAPEREQ* 和 *POLICY* 将 VLEPROD 的 *MGMTCLAS* 分配给挂载请求。VTCS 在池 LOCSCR 中选择虚拟暂存卷（范围为 V00000-V99999），并为其分配 VLEPROD *MGMTCLAS*。该卷卸载后，一个副本将迁移到本地 VLE (STORMNGR SITE1VLE)，另一个副本则迁移到远程 VLE (STORMNGR SITE2VLE)。

使用 VLE 运行 DR 测试

在 SITE2 设置 DR 测试的过程非常简便和快速，且对 SITE1 的限制最少。

基本步骤包括：

1. 在 SITE2 创建一个仅包含基本配置数据的新 CDS。
2. 将 SITE1 VMVC 标记为 *READONLY* 以避免冲突。
3. 在 SITE2VLE 中执行虚拟生产 MVC 的审计。此步骤使用现有的虚拟元数据填充 CDS。根据 VLE 中的 VTV 数量，此步骤可能需要几分钟到小于一小时的时间完成。
4. 使用一系列与生产卷不重叠的 VTV 和 MVC，运行 DR 测试工作负荷。

本节的其余部分详细说明如何在 DR 站点定义参数，并介绍您必须采取哪些步骤才能确保生产 VMVC 的内容不会在测试期间发生变化。

1. 创建 DR 测试 CDS。
 - a. 使用 *LIBGEN/SLICREAT* 过程在 SITE2 创建 CDS。请注意，即使您已在 SITE2 运行生产工作，也要创建此 CDS。新 CDS 仅包含来自 SITE1 的 DR 数据。另请注意，即使配置不包含物理磁带，您也必须在 *LIBGEN* 宏中至少定义一个 ACS。
 - b. 运行 *SET VOLPARM* 实用程序为 DR 测试定义卷：

```

POOLPARM TYPE(MVC) NAME(VAULT1)
VOLPARM VOLSER(VLV000-VLV099)
POOLPARM TYPE(EXTERNAL) NAME(PRODVTVS)
VOLPARM VOLSER(V00000-V99999) MEDIA(VIRTUAL)
POOLPARM TYPE(MVC) NAME(DRMVC)
VOLPARM VOLSER(VLT000-VLT099)
POOLPARM TYPE(SCRATCH) NAME(VIRTSCR)
VOLPARM VOLSER(VT0000-VT9999) MEDIA(VIRTUAL)

```

请注意，前两个池定义由 SITE1 创建的、将在 SITE2 用作测试输入的卷。池类型 EXTERNAL 指示这些卷不属于暂存子池的一部分。最后两个池是将在 SITE2 用作测试输出的本地池。

- c. 定义 DR 测试要使用的 VTCS *MGMTCLAS* 和 *STORCLAS*:

```

STOR NAME(DRVLE) STORMNGR(SITE2VLE) MVCPOOL(DRMVC)
MGMT NAME(VLEMGMT) DELSCR(YES) MIGPOL(DRVLE)

```

- d. 请注意，由于 SITE2 DR 系统中的 *MGMTCLAS* 和暂存子池与生产策略具有相同的名称（但其定义不同），因此您现在可以对 SITE2 DR 测试使用与 SITE1 生产环境中相同的 SMC *POLICY* 和 *TAPEREQ* 语句。

- e. 在 DR 测试 LPAR 中初启 HSC/VTCS。

2. 将生产 MVC 标记为 *READONLY*。

- a. 这是整个过程的关键步骤，必须同时对 SITE1 的生产 CDS 和 SITE2 的 DR 测试 CDS 执行。请注意，在生产 CDS 中将 MVC 定义为 *READONLY* 之后，您可以继续运行正常处理，其中包括：

RECLAIM。自动回收不会选择处于 *READONLY* 状态的 MVC。

SCRATCH。尽管 VTV 会在生产 CDS 中更新为处于暂存状态且可能会被重用，但 VLE 只读虚拟 MVC 中的副本不受影响。

正常处理将附加到 VMVC 上的 VTV，或将其覆写。新 VTV 版本将迁移到新 VMVC，而 VLE 只读虚拟 MVC 中的副本不受影响。

注：

不过请注意，不能针对这些 MVC 运行 *DRAIN* 实用程序，因为这会删除虚拟 MVC 元数据的 VLE 副本。

- b. 使用实用程序函数 *ACTMVCN*，选择生产站点中使用生产 CDS 的生产 MVC。此实用程序会生成控制语句，从而在所选 MVC 上设置 *READONLY* 标志。此外，它还会生成在测试完成之后禁用 *READONLY* 标志的控制语句。在 *ACTMVCN* 控制语句中使用 *ALL* 关键字，可确保选择所有 MVC 进行 *READONLY* 处理，这样可在不影响 DR 测试的情况下对生产系统执行自动回收。此外，您还应使用 *SLUSMAUD DD* 语句，为测试中要使用的 VMVC 生成 *AUDIT* 语句。

请注意，您可以根据需要在生产站点运行 *ACTMVCN* 实用程序来创建生产更新，并在 DR 站点对 CDS 的镜像副本运行该实用程序来创建 DR 测试 CDS 更新。以下是运行此实用程序的 JCL 示例：

```
//ACTMVCN JOB (ACCT), 'ACTMVCN', NOTIFY=&SYSUID
//ACTMVC1 EXEC PGM=SLUADMIN, PARM='MIXED'
//STEPLIB DD DSN=hlq.SEALINK, DISP=SHR
//SLSPRINT DD SYSOUT=*
//* NOTE: CDS DD statements are optional if running at the production
//* site with an active HSC LPAR.
//SLSCNTL DD DSN=hlq.DBASEPRM, DISP=SHR
//SLSCNTL2 DD DSN=hlq.DBASESEC, DISP=SHR
//SLSSTBY DD DSN=hlq.DBASESBY, DISP=SHR
//* NOTE: MVCMAINT READONLY(ON) STATEMENTS
//SLUMVON DD DSN=hlq.SLUMVON, DISP=(NEW, CATLG, DELETE),
// SPACE=(CYL, 1)
//* NOTE: MVCMAINT READONLY(OFF) STATEMENTS
//SLUMVOF DD DSN=hlq.SLUMVOF, DISP=(NEW, CATLG, DELETE),
// SPACE=(CYL, 1)
//* NOTE: AUDIT MVC(VVVVVV) STATEMENTS
//SLUSMAUD DD DSN=hlq.SLUSMAUD, DISP=(NEW, CATLG, DELETE),
// SPACE=(CYL, 1)
//* NOTE: THE FOLLOWING SELECTS ALL "NON-EMPTY" VMVCS
//SLSIN DD *
ACTMVCN ALL MVCPOOL(VAULT1)
/*
```

3. 在生产站点，运行 *MVCMAINT* 实用程序函数，将 VMVC 标记为 *READONLY*。

```
//RONLYON EXEC PGM=SLUADMIN, PARM='MIXED'
//STEPLIB DD DSN=hlq.SEALINK, DISP=SHR
//SLSPRINT DD SYSOUT=*
//* NOTE: EXEC MVCMAINT TO SET READONLY(ON). Output of
//* ACTMVCN utility.
//SLSIN DD DSN=hlq.SLUMVON, DISP=SHR
```

4. 在 DR 站点初启 HSC/VTCS。
5. 使用新创建的 SITE2 CDS 以及 *ACTMVCN* 实用程序的输出，在 SITE2 VLE 中运行生产 VMVC 的 MVC 审计。此步骤将填充包含 VTV 与 VMVC 之间关系的 CDS 元数据。

```
//AUDIT EXEC PGM=SLUADMIN
//STEPLIB DD DSN=hlq.SEALINK, DISP=SHR
//SLSPRINT DD SYSOUT=*
//* NOTE: AUDIT CONTROL STATEMENTS FROM ACTMVCN UTILITY
```

```
//SLSIN DD DSN=h1q.SLUSMAUD,DISP=SHR
```

(可选) 您可以使用 LCM 或其他选择要撤回的 VTV 的方法, 将 DR 测试中要使用的 VTV 撤回到 VTSS 缓冲区。但是, 由于从 VLE 缓冲区中撤回的速度相对较快, 因此没有必要执行此步骤。

6. 使用 *ACTMVCN READONLY(ON)* 的输出运行 *MVCMaint* 实用程序, 在 DR CDS 上的 SITE2 将生产 VMVC 设置为 *READONLY*。

```
//RONLYON EXEC PGM=SLUADMIN,PARM='MIXED'
//STEPLIB DD DSN=h1q.SEALINK,DISP=SHR
//SLSPRINT DD SYSOUT=*
//* NOTE: EXEC MVCMaint TO SET READONLY(ON). Output of
//* ACTMVCN utility.
//SLSIN DD DSN=h1q.SLUSMVON,DISP=SHR
```

7. 可选: 在开始 DR 测试之前, 您可能需要运行 *VTVRPT* 和 *MVCRPT*, 以验证 DR 测试 CDS 的内容。
8. 运行 DR 测试工作负荷。
 - a. 初启 SMC。如果在 *MGMTCLAS* 和暂存子池中使用与生产系统相同的名称, 则可使用生产 *TAPEREQ* 和 *POLICY* 语句。建议 (并非强制) 您对 DR 测试 TapePlex 使用不同的 TapePlex 名称。
 - b. 使用 SMC 和新 HSC/VTCS CDS 运行 DR 测试工作负荷。
 - c. 对于 DR 测试期间更新生产 VTV 卷, 这一点并无限制。生产 VTV 中的数据可能会附加到 (*DISP=MOD*) 或被 (*DISP=OLD*) 覆写。这些更新不会影响 *READONLY* 生产虚拟 MVC 上 VTV 副本的内容, 因此不影响数据的生产副本。

使用 VLE 运行 DR 测试之后清除

DR 测试完成后, 清理的目的在于将元数据从 VTSS 和 VLE 中删除, 以便下次 DR 测试不会看到此数据。请注意, DR 测试 HSC/VTCS 必须保持活动状态, 直到清理完成为止。步骤包括:

1. 运行 *SCRATCH* 实用程序函数, 暂存测试期间通过 VTSS 和 VLE DR 测试 VMVC 创建的所有 VTV。在 DR 测试 *MGMTCLAS* 中指定 *DELSCR(YES)* 参数后, 运行暂存实用程序时, 将导致 VTV 同时从缓冲区和 VLE 元数据中删除。

```
//SCRATCH EXEC PGM=SLUADMIN
//STEPLIB DD DSN=h1q.SEALINK,DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
SCRATCH VOL(VT0000-VT9999)
```

请注意，如果已使用 *DISP=MOD* 或 *DISP=OLD* 修改任何生产 VTV，则这些 VTV 将保留在缓冲区和 VLE 中。

通过在测试之后将 VTV 暂存在 DR 测试子池中，您可以最大程度地减少清除 VTSS 所需的时间以及测试完成后留在 VLE 中的数据量。

2. 将 VTSS 迁移至 0。

```
//MIGRT00 EXEC PGM=SLUADMIN
//STEPLIB DD DSN=h1q.SEALINK,DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
MIGRATE VTSS(DRVTSS) THRESHLD(0)
```

仅当 DR 测试的输出涉及新版本的生产 VTV 时，才需要执行此步骤。

3. 验证 DR VTSS 现在是否为空。

```
//AUDVTSS EXEC PGM=SLUADMIN
//STEPLIB DD DSN=h1q.SEALINK,DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
AUDIT VTSS(DRVTSS)
```

请注意，如果您在 DR 测试期间修改了生产 VTV，则此数据和元数据的副本将保留在 VLE 中，供 DR 测试 MVC 池使用（VLT000-VLT099、VTV V00000-V99999）。在下次 DR 测试期间，这些 VMVC 将从磁带的逻辑起点开始写入，且其中包含的任何数据均会从 VLE 中删除。由于新的 DR 测试 CDS 不识别此数据，因此不会影响下次 DR 测试。

4. 在生产站点，使用 *ACTMVCGN* 实用程序在测试开始时创建的 *READONLY(OFF)* 控制卡，将生产 VMVC 返回到可写状态。

```
//RDONLYOF EXEC PGM=SLUADMIN,PARM='MIXED'
//STEPLIB DD DSN=h1q.SEALINK,DISP=SHR
//SLSPRINT DD SYSOUT=*
/* NOTE: EXEC MVCMAINT TO SET READONLY(OFF)
//SLSIN DD DSN=h1q.SLUSMVOF,DISP=SHR
```

使用 VLE 实现业务连续性

当 SITE1 出现故障且需要 SITE2 接管 SITE1 的工作负荷时，该过程与 DR 测试过程基本相同。

如果 SITE1 出现故障时正好在执行 DR 测试，请遵循以上过程清除 DR 测试并停止 DR 测试。

要开始在 SITE2 中运行 SITE1 的工作负荷，请遵循上述过程开始 DR 测试。当然，由于没有要更新的“生产”CDS，因此您将忽略在生产 CDS 中将生产 VMVC 标记为 *READONLY* 的步骤。但是，您将使用生产 CDS 的镜像副本为 VLE 中的生产 MVC 生成 *MVCMaint READONLY* 控制卡。

此外，您还需要使用 DR 测试策略将正在创建的 VTV 和输出 VMVC 隔离至单独的范围中，避免出现任何可能会损坏生产数据的情况，直到验证业务连续性为止。

注：

如果针对磁带数据执行 *DISP=MOD* 处理的生产作业没有既定的同步点，则出现故障时可能无法预测 VTV 的内容。StorageTek 建议审核所有灾难恢复过程，确保磁带数据的同步点可预测。

附录 A. 群集 VTSS 示例

"[使用群集 VTSS 配置](#)"提供了 VTSS 群集的基础知识，而本附录提供了以下群集示例：

- "单向群集 VTSS"
- "双向群集 VTSS"
- "扩展群集"
- "带有 TCP/IP CLINK 的 VSM5 到 VSM5 群集"
- "带有 TCP/IP CLINK 和交叉连接 VLE 的 VSM5 到 VSM6 群集"
- "带有 TCP/IP CLINK 的 VSM6 到 VSM6"无磁带"群集"
- "应该使用单向还是双向群集？"

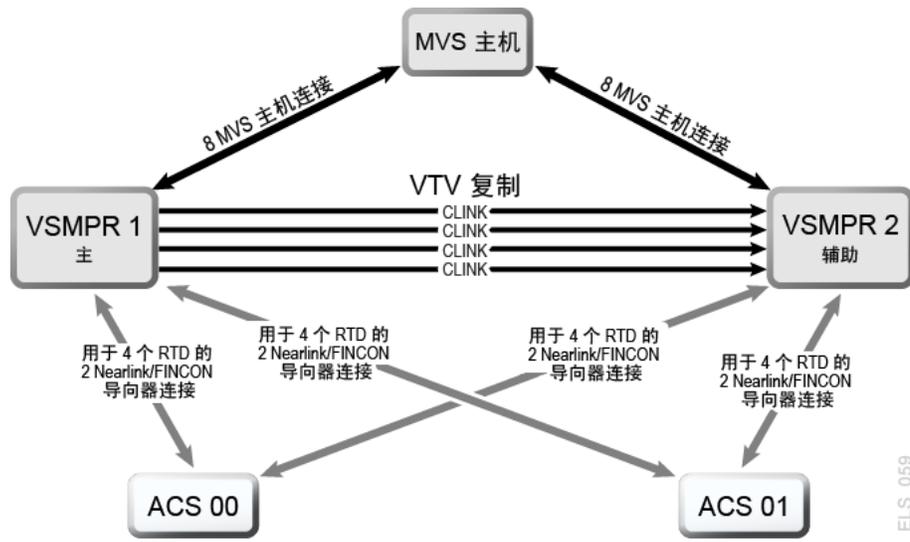
单向群集 VTSS

[图 A.1 "双 ACS 单向群集 VTSS 配置"](#) 显示了单向群集 VTSS 双 ACS 系统的示例。请注意，在此示例中，FICON 端口提供 CLINK 连接。在此示例中，只有一个 MVS 主机，但是它产生大量的关键数据，您必须使用刚刚购买的两个全新 VSM4 来保护这些数据。

VTSS1 是主 VTSS，它使用群集链路 (CLINK) 连接到辅助 VTSS (VTSS2)。如果 VTV 的管理类指定了复制，则当 VTV 到达 VTSS1 中时，它将复制到 VTSS2，并且也会立即迁移（使用 KEEP）。

结果，您提高了数据可用性（每个 VTSS 中都有 VTV 的副本，以防一个发生故障）和数据保护（VTV 也在备用磁带上，以防两个 VTSS 都脱机）。因此，群集 VTSS 对于业务连续性和业务恢复来说是一个出色的解决方案。

图 A.1. 双 ACS 单向群集 VTSS 配置



现在来看一下这种群集配置的硬件。图 A.2 “带有 8 个 VCF 卡、8 个主机端口、8 个 RTD 的 FICON 导向器和 4 个 CLINK 端口的 VSM4” 显示了带有 8 个 VCF 卡的 VSM4 的 CONFIG 通道接口标识符。在这种配置中，您已分配：

- 8 个主机端口
- RTD 的 4 个端口。RTD 端口全部连接到 FICON 导向器，每个导向器都连接到 RTD，因此在每个端口上显示两个 RTD 的 *CHANIF* 标识符。这样允许后端连接到 8 个 RTD，尽管每个端口/导向器一次只能有一个 RTD 处于活动状态。
- 使 CLINK 连接构成单向 VTSS 群集的 4 个端口，以及主机连接的 8 个端口。有两个配置完全相同的 VSM4 (VTSS1 和 VTSS2) 构成群集 VTSS，如图 A.2 “带有 8 个 VCF 卡、8 个主机端口、8 个 RTD 的 FICON 导向器和 4 个 CLINK 端口的 VSM4” 中所示。

图 A.2. 带有 8 个 VCF 卡、8 个主机端口、8 个 RTD 的 FICON 导向器和 4 个 CLINK 端口的 VSM4



您现在已看到示例单向群集是什么样子，并且也已看到所需的 VCF 卡端口配置。现在来看一下“[配置和管理单向群集 VTSS 系统](#)”。

配置和管理单向群集 VTSS 系统

要配置和管理图 6.1 “单向群集 VTSS” 中所示的单向群集系统，请执行以下操作：

1. 确保您的系统需要群集 VTSS。
2. 使用 *CONFIG* 创建 *CLUSTER* 和 *CLINK* 语句以定义 VTSS 群集及其连接。

以下示例显示了使用 *CONFIG JCL* 定义两个 VSM4 (VTSS1 和 VTSS2) 的单向群集，如图 6.1 “单向群集 VTSS” 中所示。

请注意：

- *CLUSTER* 语句将群集定义为由 VTSS1 和 VTSS2 组成。
- *CLINK* 语句定义仅使用 **VTSS1** 的发送 (Nearlink 模式) 端口以单向方式启用群集，其中 VTSS1 是主 VTSS，VTSS2 是辅助 VTSS。

```
//CREATECFG EXEC PGM=SWSADMIN,PARM='MIXED'
//STEPLIB DD DSN=h1q.SLSLINK,DISP=SHR
//SLSCNTL DD DSN=FEDB.VSMLMULT.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=FEDB.VSMLMULT.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=FEDB.VSMLMULT.DBASESTBY,DISP=SHR
//CFG22202 DD DSN=FEDB.VSMLMULT.CFG22202,DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
CONFIG RESET CDSLEVEL(V62ABOVE)
GLOBAL MAXVTV=65000 MVCFREE=60 VTVATTR=SCRATCH RECALWER=YES
```

```

LOCKSTR=STK_VTCS_LOCKS VTPAGE=LARGE
RECLAIM THRESHLD=70 MAXMVC=30 START=40 CONMVC=5
VTVVOL LOW=905000 HIGH=999999 SCRATCH
VTVVOL LOW=C00000 HIGH=C25000 SCRATCH
VTVVOL LOW=RMM000 HIGH=RMM020 SCRATCH
MVCVOL LOW=N25980 HIGH=N25989
MVCVOL LOW=N35000 HIGH=N35999
VTSS NAME=VTSS1 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
RTD NAME=PR11A00 DEVNO=1A00 CHANIF=0C
RTD NAME=PR11A01 DEVNO=1A01 CHANIF=0D
RTD NAME=PR11A02 DEVNO=1A02 CHANIF=0K
RTD NAME=PR11A03 DEVNO=1A03 CHANIF=0L
RTD NAME=PR12A08 DEVNO=2A08 CHANIF=1C
RTD NAME=PR12A09 DEVNO=2A09 CHANIF=1D
RTD NAME=PR12A0A DEVNO=2A0A CHANIF=1K
RTD NAME=PR12A0B DEVNO=2A0B CHANIF=1L
VTD LOW=9900 HIGH=99FF
VTSS NAME=VTSS2 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
RTD NAME=PR23A00 DEVNO=3A00 CHANIF=0C
RTD NAME=PR23A01 DEVNO=3A01 CHANIF=0D
RTD NAME=PR23A02 DEVNO=3A02 CHANIF=0K
RTD NAME=PR23A03 DEVNO=3A03 CHANIF=0L
RTD NAME=PR24A08 DEVNO=4A08 CHANIF=1C
RTD NAME=PR24A09 DEVNO=4A09 CHANIF=1D
RTD NAME=PR24A0A DEVNO=4A0A CHANIF=1K
RTD NAME=PR24A0B DEVNO=4A0B CHANIF=1L
VTD LOW=9900 HIGH=99FF
CLUSTER NAME=CLUSTER1 VTSSs(VTSS1,VTSS2)
CLINK VTSS=VTSS1 CHANIF=0G
CLINK VTSS=VTSS1 CHANIF=00
CLINK VTSS=VTSS1 CHANIF=1G
CLINK VTSS=VTSS1 CHANIF=10

```

3. 在 *CONFIG GLOBAL* 语句中指定条件性复制设置。

```
CONFIG GLOBAL REPLICAT=CHANGED
```

在此示例中，*CONFIG GLOBAL REPLICAT=CHANGED* 指定：

- 仅当 VTV 已更新且辅助 VTSS 上不存在完全相同的副本时才复制 VTV。
- 使用 *MIGPOL* 参数，按您在第 5 步中创建的存储类以双工方式将 VTV 迁移到 ACS 00 和 01。

要无条件地复制 VTV，请指定 *CONFIG GLOBAL REPLICAT=ALWAYS*。

4. 创建用于指定 VTV 复制的一个管理类以及用于迁移（以双工方式）复制的 VTV 的两个存储类。

```
MGMT NAME(VSMREPL) REPLICAT(YES) MIGPOL(REPLSTR1,REPLSTR2)
```

注：

- 请注意 *GLOBAL REPLICAT*（指定何时可以执行复制）与 *MGMTclas REPLICAT(YES)*（表示当 *GLOBAL REPLICAT* 条件表明到时间时着手复制）之间的交互。
- 管理类 *VSMREPL* 并未指定立即迁移策略。VTV 复制自动强制执行立即迁移。复制完成后，会将该管理类中的 VTV 添加到 VTSS 上的立即迁移队列中。请注意，双工并不是复制 VTV 的要求。有关更多信息，请参见“[群集 VTSS 配置的工作原理](#)”。

5. 为包含复制的已迁移 VTV 的 MVC 创建存储类。

```
STOR NAME(REPLSTR1) ACS(00) MEDIA(STK1R) MIRATE(RECEIVER)
STOR NAME(REPLSTR2) ACS(01) MEDIA(STK1R) MIGRATE(RECEIVER)
```

在此示例中，*STORclas* 语句定义了在第 4 步的 *MIGPOL* 参数中引用的存储类 *REPLSTR1* 和 *REPLSTR2*。另请注意，存储类的 *MIGRATE* 参数指定接收复制的 VTV 的 VTSS（在本例中为 *VTSS2*，即辅助 VTSS）执行向两个 ACS 的迁移。采用这种方式可确保辅助 VTSS 起到迁移引擎的作用。

6. 使用 *MGMTDEF* 命令装入 *MGMTclas* 和 *STORclas* 控制语句。

```
MGMTDEF DSN(hsc.parms)
```

7. 创建 *TAPEREQ* 语句，以将关键数据发送到 VSM 并为数据分配管理类 *VSMREPL*。

```
TAPEREQ DSN(*.PAYROLL.***) MEDIA(VIRTUAL) MGMT(VSMREPL)
```

在此示例中，*TAPEREQ* 语句指定：

- 将包含 HLQ 掩码 **.PAYROLL.*** 的数据集发送到 VSM。
- 分配您在第 4 步中创建的管理类 *VSMREPL*。

注意：

要复制 VTV，*VTSS1* 和 *VTSS2* 必须都变为联机到 VTCS，以便 VTCS 可以向两个 VTSS 发送控制命令。有关更多信息，请参见“[群集 VTSS 配置的工作原理](#)”。

您也可以通过 SMC *TAPEREQ* 语句或者 SMC DFSMS ACS 例程使用专用设备组替换将复制作业发送到 VSM。有关更多信息，请参见《*SMC Configuration and Administration Guide*》。

8. 检查 HSC PARMLIB 选项，以确保启用了子类型 28 记录。

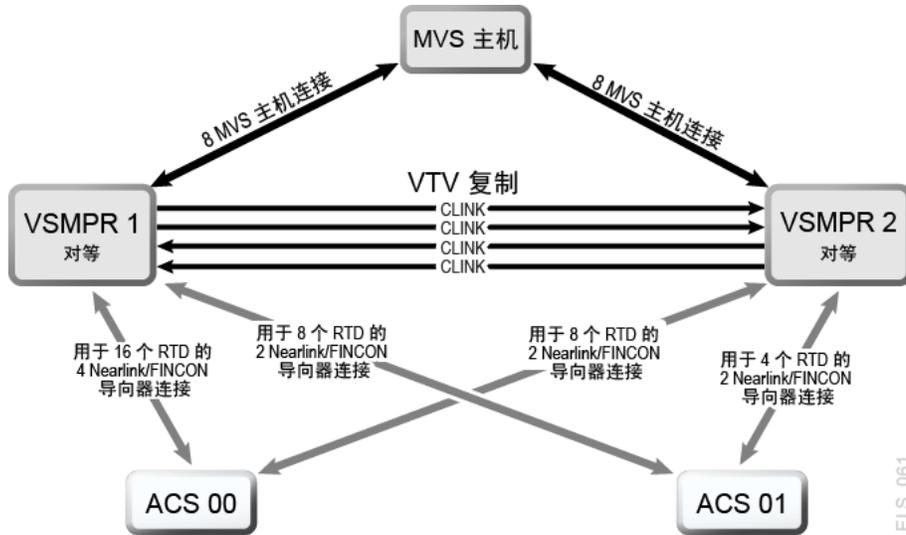
如果启用了该选项，VTSS 群集将针对每次执行的复制编写一条子类型 28 记录。

双向群集 VTSS

图 A.3 “双 ACS 双向群集 VTSS 配置” 显示了双向群集 VTSS 双 ACS 系统的示例。请注意，在此示例中，FICON 端口提供 CLINK 连接。

该系统非常类似于单向群集的示例，只是又多了一步：有两个 MVS 主机共享一个 CDS，图中的一切都是交叉连接。这些站点互相镜像，以实现最佳数据可用性和数据保护。要使其成为双向群集，必须使用 *CLINK* 语句将两个 VTSS 配置为对等设备。

图 A.3. 双 ACS 双向群集 VTSS 配置



注：

- 双向群集要求运行 VTCS 6.1 及更高版本。您无法在低于 VTCS 6.1 的发行版上配置双向群集。
- 显示的这种配置具有最多启用总共 16 个并发 NearLink I/O 传输的功能。这些可以分布于多达 14 个 NearLink 端口上的多个目标，每个端口最多总共 2 个并发 NearLink I/O 传输。这项功能要求运行 VTSS 微代码 D02.06.00.00 或更高版本。

图 A.4 “带有 8 个 VCF 卡、8 个主机端口、24 个 RTD 的 FICON 导向器和 4 个 CLINK 的 VSMPR1-VSM5” 显示了图 A.3 “双 ACS 双向群集 VTSS 配置” 中所示 VSMPR1 的 *CONFIG* 通道接口标识符。在这种配置中，您已分配：

- 8 个主机端口
- RTD 的 6 个端口。RTD 端口全部连接到 FICON 导向器，每个导向器都连接到 4 个 RTD，因此在每个端口上显示全部 4 个 RTD 的 CHANIF 标识符。这样允许后端连接到 ACS00 的 24 个 RTD，尽管每个端口/导向器一次只能有一个 RTD 处于活动状态。
- 使用 FICON 导向器的 4 个端口。两个是发起方的 Nearlink，两个是终止方的主机模式，用于使 CLINK 连接构成双向 VTSS 群集。

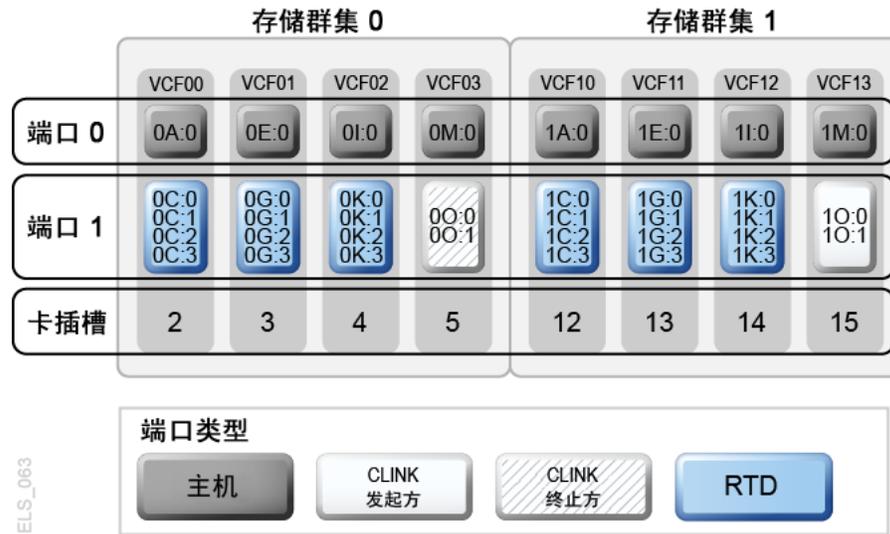
图 A.4. 带有 8 个 VCF 卡、8 个主机端口、24 个 RTD 的 FICON 导向器和 4 个 CLINK 的 VSMR1-VSM5



图 A.5 “带有 8 个 VCF 卡、8 个主机端口、24 个 RTD 的 FICON 导向器和 4 个 CLINK 的 VSMR2-VSM5” 显示了 VSMR1 的 *CONFIG* 通道接口标识符，VSMR1 是双向群集中的 VSM5，它带有 8 个 VCF 卡并启用了最多 32 个 RTD 功能。在这种配置中，您已分配：

- 8 个主机端口
- RTD 的 6 个端口。RTD 端口全部连接到 FICON 导向器，每个导向器都连接到 4 个 RTD，因此在每个端口上显示全部 4 个 RTD 的 *CHANIF* 标识符。这样允许后端连接到 24 个 RTD，尽管每个端口/导向器一次只能有一个 RTD 处于活动状态。

图 A.5. 带有 8 个 VCF 卡、8 个主机端口、24 个 RTD 的 FICON 导向器和 4 个 CLINK 的 VSMR2-VSM5



- 使用 FICON 导向器的 4 个端口。两个是发起方的 Nearlink，两个是终止方的主机模式，用于使 CLINK 连接构成双向 VTSS 群集。

注意：

如图 6.3 “双向群集 VTSS 的 ESCON/FICON CLINK” 中所示，每个 CLINK 必须连接到每个 VTSS 上的同一个存储群集，这是一项要求。未能以这种方式配置会产生复制、通道和通信错误！所以如图中所示，VSMR1 上的 Nearlink 端口（CLINK 发起方）在存储群集 0 上，VSMR2 上的主机端口（CLINK 终止方）也在存储群集 0 上。对于数据在另一个方向流动的 CLINK 连接同样如此；它们都在存储群集 1 上。

配置和管理双向群集系统

要配置和管理图 A.3 “双 ACS 双向群集 VTSS 配置” 中所示的双向群集系统，请执行以下操作：

1. 确保您的系统满足安装 ELS 中所述的群集 VTSS 要求。
2. 使用 CONFIG 创建 CLUSTER 和 CLINK 语句以定义 VTSS 群集及其连接。

以下示例显示了使用 CONFIG JCL 定义两个 VSM4（VSMR1 和 VSMR2）的双向群集，如图 A.3 “双 ACS 双向群集 VTSS 配置” 中所示。

- 请注意，CLUSTER 语句将群集定义为由 VSMR1 和 VSMR2 组成。
- CLINK 语句定义使用两个 VTSS 的发送（Nearlink 模式）端口以双向方式启用群集，对于每个 CLINK 的发送和接收端口，它们使用每个 VTSS 上的同一个存储群集连接。

```
//CREATECF EXEC PGM=SWSADMIN, PARM='MIXED'
//STEPLIB DD DSN=h1q.SLSLINK, DISP=SHR
```

```
//SLSCNTL DD DSN=FEDB.VSMLMULT.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=FEDB.VSMLMULT.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=FEDB.VSMLMULT.DBASETBY,DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
CONFIG RESET CDSLEVEL(V61ABOVE)
GLOBAL MAXVTV=32000 MVCFREE=40 VTVATTR=SCRATCH RECALWER=YES LOCKSTR=VTCS_LOCKS
REPLICAT=ALWAYS VTPAGE=LARGE SYNCHREP=YES MAXRTDS=32
RECLAIMTHRESHLD=70 MAXMVC=40 START=35
RECLAIMTHRESHLD=70 MAXMVC=40 START=35
VTVVOL LOW=905000 HIGH=999999 SCRATCH
VTVVOL LOW=C00000 HIGH=C25000 SCRATCH
VTVVOL LOW=RMM000 HIGH=RMM020 SCRATCH
MVCVOL LOW=N25980 HIGH=N25989
MVCVOL LOW=N35000 HIGH=N35999
VTSS NAME=VSMR1 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
RTD NAME=VPR12A00 DEVNO=2A00 CHANIF=0C:0
RTD NAME=VPR12A01 DEVNO=2A01 CHANIF=0C:1
RTD NAME=VPR12A02 DEVNO=2A02 CHANIF=0C:2
RTD NAME=VPR12A03 DEVNO=2A03 CHANIF=0C:3
RTD NAME=VPR12A04 DEVNO=2A04 CHANIF=0G:0
RTD NAME=VPR12A05 DEVNO=2A05 CHANIF=0G:1
RTD NAME=VPR12A06 DEVNO=2A06 CHANIF=0G:2
RTD NAME=VPR12A07 DEVNO=2A07 CHANIF=0G:3
RTD NAME=VPR12A08 DEVNO=2A08 CHANIF=0K:0
RTD NAME=VPR12A09 DEVNO=2A09 CHANIF=0K:1
RTD NAME=VPR12A0A DEVNO=2A0A CHANIF=0K:2
RTD NAME=VPR12A0B DEVNO=2A0B CHANIF=0K:3
RTD NAME=VPR13A00 DEVNO=3A00 CHANIF=1C:0
RTD NAME=VPR13A01 DEVNO=3A01 CHANIF=1C:1
RTD NAME=VPR13A02 DEVNO=3A02 CHANIF=1C:2
RTD NAME=VPR13A03 DEVNO=3A03 CHANIF=1C:3
RTD NAME=VPR13A04 DEVNO=3A04 CHANIF=1G:0
RTD NAME=VPR13A05 DEVNO=3A05 CHANIF=1G:1
RTD NAME=VPR13A06 DEVNO=3A06 CHANIF=1G:2
RTD NAME=VPR13A07 DEVNO=3A07 CHANIF=1G:3
RTD NAME=VPR13A08 DEVNO=3A08 CHANIF=1K:0
RTD NAME=VPR13A09 DEVNO=3A09 CHANIF=1K:1
RTD NAME=VPR13A0A DEVNO=3A0A CHANIF=1K:2
RTD NAME=VPR13A0B DEVNO=3A0B CHANIF=1K:3
VTD LOW=9900 HIGH=99FF
VTSS NAME=VSMR2 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
RTD NAME=VPR22B00 DEVNO=2B00 CHANIF=0C:0
RTD NAME=VPR22B01 DEVNO=2B01 CHANIF=0C:1
RTD NAME=VPR22B02 DEVNO=2B02 CHANIF=0C:2
```

```

RTD NAME=VPR22B03 DEVNO=2B03 CHANIF=0C:3
RTD NAME=VPR22B04 DEVNO=2B04 CHANIF=0G:0
RTD NAME=VPR22B05 DEVNO=2B05 CHANIF=0G:1
RTD NAME=VPR22B06 DEVNO=2B06 CHANIF=0G:2
RTD NAME=VPR22B07 DEVNO=2B07 CHANIF=0G:3
RTD NAME=VPR22B08 DEVNO=2B08 CHANIF=0K:0
RTD NAME=VPR22B09 DEVNO=2B09 CHANIF=0K:1
RTD NAME=VPR22B0A DEVNO=2B0A CHANIF=0K:2
RTD NAME=VPR22B0B DEVNO=2B0B CHANIF=0K:3
RTD NAME=VPR23B00 DEVNO=3B00 CHANIF=1C:0
RTD NAME=VPR23B01 DEVNO=3B01 CHANIF=1C:1
RTD NAME=VPR23B02 DEVNO=3B02 CHANIF=1C:2
RTD NAME=VPR23B03 DEVNO=3B03 CHANIF=1C:3
RTD NAME=VPR23B04 DEVNO=3B04 CHANIF=1G:0
RTD NAME=VPR23B05 DEVNO=3B05 CHANIF=1G:1
RTD NAME=VPR23B06 DEVNO=3B06 CHANIF=1G:2
RTD NAME=VPR23B07 DEVNO=3B07 CHANIF=1G:3
RTD NAME=VPR23B08 DEVNO=3B08 CHANIF=1K:0
RTD NAME=VPR23B09 DEVNO=3B09 CHANIF=1K:1
RTD NAME=VPR23B0A DEVNO=3B0A CHANIF=1K:2
RTD NAME=VPR23B0B DEVNO=3B0B CHANIF=1K:3
VTD LOW=9900 HIGH=99FF
CLUSTER NAME=CLUSTER1 VTSSs(VSMR1,VSMR2)
CLINK VTSS=VSMR1 CHANIF=00:0
CLINK VTSS=VSMR1 CHANIF=00:1
CLINK VTSS=VSMR2 CHANIF=10:0
CLINK VTSS=VSMR2 CHANIF=10:1

```

3. 在 *CONFIG GLOBAL* 语句中指定条件性复制设置。

```
CONFIG GLOBAL REPLICAT=CHANGED
```

与单向群集的示例一样，在此示例中，也使用 *CONFIG GLOBAL REPLICAT=CHANGED*。

4. 创建用于指定 VTV 复制的一个管理类以及用于迁移（以双工方式）复制的 VTV 的两个存储类。

```
MGMT NAME(VSMREPL) REPLICAT(YES) MIGPOL(REPLSTR1,REPLSTR2)
```

在此示例中，仅当 VTV 已更改且群集中的另一个 VTSS 中不存在 VTV 时才复制 VTV。按您将在第 5 步中创建的存储类以双工方式迁移到 ACS 01 和 00。

5. 为包含复制的已迁移 VTV 的 MVC 创建存储类。

```
STOR NAME(REPLSTR1) ACS(01) MEDIA(STK1R) MIRATE(EITHER)
```

```
STOR NAME(REPLSTR2) ACS(00) MEDIA(STK1R) MIGRATE(EITHER)
```

在此示例中，*STORclas* 语句定义了在第 4 步的 *MIGPOL* 参数中引用的存储类 *REPLSTR1* 和 *REPLSTR2*。另请注意，为了优化 VTSS 和 RTD 资源，存储类的 *MIGRATE* 参数允许从任一 VTSS 迁移。这是双向或对等 VTSS 群集的典型策略。

6. 使用 *MGMTDEF* 命令装入 *MGMTclas* 和 *STORclas* 控制语句。

```
MGMTDEF DSN(hsc.parms)
```

7. 创建 *TAPEREQ* 语句，以将关键数据发送到 VSM 并为数据分配管理类 *VSMREPL*。

```
TAPEREQ DSN(*.PAYROLL.***) MEDIA(VIRTUAL) MGMT(VSMREPL)
```

在此示例中，*TAPEREQ* 语句指定：

- 将包含 HLQ 掩码 **.PAYROLL.*** 的数据集发送到 VSM。
- 分配您在第 4 步中启用的管理类 *VSMREPL*。

注意：

要复制 VTV，*VSMR1* 和 *VSMR2* 必须都变为联机到 VTCS，以便 VTCS 可以向两个 VTSS 发送控制命令。有关更多信息，请参见[“群集 VTSS 配置的工作原理”](#)。

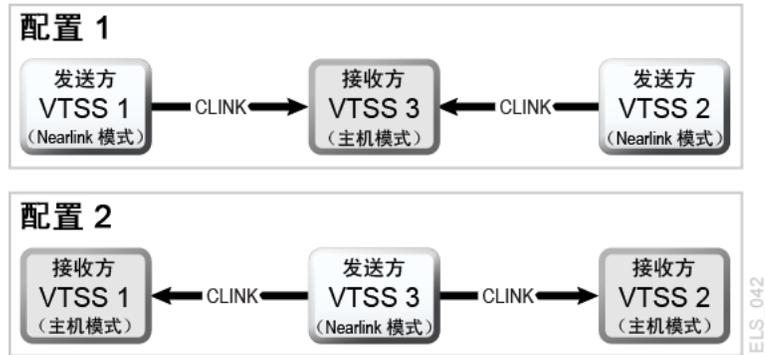
- 您也可以通 *SMC TAPEREQ* 语句或者 ELS 用户出口使用专用设备组替换将复制作业发送到 VSM。如果替换了一个跨越所有对等 VTSS 中所有 VTD 的专用设备组，则当群集中的一个对等 VTSS 脱机时，VTCS 可以继续正确影响分配。
 - 对于 SMC，管理类名称（如果在 StorageTek DFSMS 界面中分配了该名称）在分配时可用。因此，在界面中分配的专用设备组不再需要只包含属于群集的 VTSS。只要专用设备组包含位于全功能群集中主 VTSS 上的某些磁带机，SMC 便有足够的信息使分配指向主 VTSS 上的磁带机（如果启用了管理类指定复制）。
8. 检查 HSC PARMLIB 选项，以确保启用了子类型 28 记录。

如果启用了该选项，VTSS 群集将针对每次执行的复制编写一条子类型 28 记录。

扩展群集

扩展群集 (Extended Clustering, EC) 允许三个或更多 VTSS 由单个 Tapeplex (1 个 CDS) 配置中的 CLINK 进行连接，如图 A.6 “基本扩展群集配置” 中的示例中所示。

图 A.6. 基本扩展群集配置



配置和管理包含 3 个 VTSS 的群集系统

如图 A.6 “基本扩展群集配置” 中所示，配置 1 显示 2 个 VTSS 复制到单个“收集器”VTSS，这是最实用的配置，因为包含多个 VSM 的主位置可以将 VTV 馈送到包含单个收集器 VSM 的辅助位置。同步和异步复制均可在每个发送方 VTSS 上使用。每个 VTSS 必须连接了等效（相似型号）的 RTD。如下方配置 1 的 CONFIG 语句中所示：

- *CLUSTER* 语句定义为群集配置的所有 VTSS 名称。
- *CLINK* 语句定义发送 VTSS 上的 Nearlink 端口位置及其 PARTNER 或目标 VTSS。

```

/CREATECFG EXEC PGM=SLUADMIN, PARM='MIXED'
//STEPLIB DD DSN=h1q.SEALINK, DISP=SHR
//SLSCNTL DD DSN=h1q.DBASEPRM, DISP=SHR
//SLSCNTL2 DD DSN=h1q.DBASESEC, DISP=SHR
//SLSSTBY DD DSN=h1q.DBASETBY, DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
CONFIG RESET CDSLEVEL(V62ABOVE)
GLOBAL MAXVTV=65000 MVCFREE=60 VTVATTR=SCRATCH RECALWER=YES
LOCKSTR=STK_VTCS_LOCKS VTVPAGE=LARGE
RECLAIM THRESHLD=70 MAXMVC=30 START=40 CONMVC=5
VTSS NAME=VTSS1 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
RTD NAME=PA11A00 DEVNO=1A00 CHANIF=0C
RTD NAME=PA11A01 DEVNO=1A01 CHANIF=0D
RTD NAME=PA11A02 DEVNO=1A02 CHANIF=0K
RTD NAME=PA11A03 DEVNO=1A03 CHANIF=0L
RTD NAME=PA12A08 DEVNO=2A08 CHANIF=1C
RTD NAME=PA12A09 DEVNO=2A09 CHANIF=1D
RTD NAME=PA12A0A DEVNO=2A0A CHANIF=1K
RTD NAME=PA12A0B DEVNO=2A0B CHANIF=1L
VTD LOW=7900 HIGH=79FF
VTSS NAME=VTSS2 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
  
```

```

RTD NAME=PA23A00 DEVNO=3A00 CHANIF=0C
RTD NAME=PA23A01 DEVNO=3A01 CHANIF=0D
RTD NAME=PA23A02 DEVNO=3A02 CHANIF=0K
RTD NAME=PA23A03 DEVNO=3A03 CHANIF=0L
RTD NAME=PA24A08 DEVNO=4A08 CHANIF=1C
RTD NAME=PA24A09 DEVNO=4A09 CHANIF=1D
RTD NAME=PA24A0A DEVNO=4A0A CHANIF=1K
RTD NAME=PA24A0B DEVNO=4A0B CHANIF=1L
VTD LOW=8900 HIGH=89FF
VTSS NAME=VTSS3 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
RTD NAME=PA33A00 DEVNO=3A00 CHANIF=0C
RTD NAME=PA33A01 DEVNO=3A01 CHANIF=0D
RTD NAME=PA33A02 DEVNO=3A02 CHANIF=0K
RTD NAME=PA33A03 DEVNO=3A03 CHANIF=0L
RTD NAME=PA34A08 DEVNO=4A08 CHANIF=1C
RTD NAME=PA34A09 DEVNO=4A09 CHANIF=1D
RTD NAME=PA34A0A DEVNO=4A0A CHANIF=1K
RTD NAME=PA34A0B DEVNO=4A0B CHANIF=1L
VTD LOW=9900 HIGH=99FF
CLUSTER NAME=CLUSTER1 VTSS(VTSS1,VTSS2,VTSS3)
CLINK VTSS=VTSS1 CHANIF=0G PART=VTSS3
CLINK VTSS=VTSS2 CHANIF=0G PART=VTSS3

```

如图 A.6 “基本扩展群集配置” 中所示，配置 2 显示单个复制 VTSS 连接到 2 个接收方 VTSS。请注意，此处未使用“收集器”一词，因为 VTV 仅复制到一个 VTSS（VTSS1 或 VTSS2），并且接收方 VTSS 不可配置。这是一个要理解的非常重要的概念，因为当前没有任何管理类参数用于选择特定 VTSS 来指引 VTV。这种配置对于 VTV 必须最终处于特定辅助位置的主位置和辅助位置环境中的实施没有用处，可能会使扩展双向配置不合需要。同步和异步复制均可在发送方 VTSS 上使用。每个 VTSS 必须连接了等效（相似型号）的 RTD。

当决定在扩展群集环境中实施双向复制时，配置 2 变得最为重要。需要进行双向复制，然后在一个方向上使用“多个 VTSS 对一个 VTSS”配置，在另一个方向上使用“VTSS 对”配置，其中“VTSS 对”是在复制的 VTV 必须位于的两个 VTSS 之间配置的。

如下方配置 2 的 *CONFIG* 语句中所示：

- CLUSTER 语句定义为群集配置的所有 VTSS 名称。
- CLINK 语句定义发送 VTSS 上的 Nearlink 端口位置及其 PARTNER 或目标 VTSS。

```

//CREATCFG EXEC PGM=SLUADMIN,PARM='MIXED'
//STEPLIB DD DSN=h1q.SEALINK,DISP=SHR
//SLSCNTL DD DSN=h1q.DBASEPRM,DISP=SHR

```

```
//SLSCNTL2 DD DSN=h1q.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=h1q.DBASETBY,DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
CONFIG RESET CDSLEVEL(V62ABOVE)
GLOBAL MAXVTV=65000 MVCFREE=60 VTVATTR=SCRATCH RECALWER=YES
LOCKSTR=STK_VTCS_LOCKS VTVPAGE=LARGE
RECLAIM THRESHLD=70 MAXMVC=30 START=40 CONMVC=5
VTSS NAME=VTSS1 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
RTD NAME=PA11A00 DEVNO=1A00 CHANIF=0C
RTD NAME=PA11A01 DEVNO=1A01 CHANIF=0D
RTD NAME=PA11A02 DEVNO=1A02 CHANIF=0K
RTD NAME=PA11A03 DEVNO=1A03 CHANIF=0L
RTD NAME=PA12A08 DEVNO=2A08 CHANIF=1C
RTD NAME=PA12A09 DEVNO=2A09 CHANIF=1D
RTD NAME=PA12A0A DEVNO=2A0A CHANIF=1K
RTD NAME=PA12A0B DEVNO=2A0B CHANIF=1L
VTD LOW=7900 HIGH=79FF
VTSS NAME=VTSS2 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
RTD NAME=PA23A00 DEVNO=3A00 CHANIF=0C
RTD NAME=PA23A01 DEVNO=3A01 CHANIF=0D
RTD NAME=PA23A02 DEVNO=3A02 CHANIF=0K
RTD NAME=PA23A03 DEVNO=3A03 CHANIF=0L
RTD NAME=PA24A08 DEVNO=4A08 CHANIF=1C
RTD NAME=PA24A09 DEVNO=4A09 CHANIF=1D
RTD NAME=PA24A0A DEVNO=4A0A CHANIF=1K
RTD NAME=PA24A0B DEVNO=4A0B CHANIF=1L
VTD LOW=8900 HIGH=89FF
VTSS NAME=VTSS3 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
RTD NAME=PA33A00 DEVNO=3A00 CHANIF=0C
RTD NAME=PA33A01 DEVNO=3A01 CHANIF=0D
RTD NAME=PA33A02 DEVNO=3A02 CHANIF=0K
RTD NAME=PA33A03 DEVNO=3A03 CHANIF=0L
RTD NAME=PA34A08 DEVNO=4A08 CHANIF=1C
RTD NAME=PA34A09 DEVNO=4A09 CHANIF=1D
RTD NAME=PA34A0A DEVNO=4A0A CHANIF=1K
RTD NAME=PA34A0B DEVNO=4A0B CHANIF=1L
VTD LOW=9900 HIGH=99FF
CLUSTER NAME=CLUSTER1 VTSS(VTSS1,VTSS2,VTSS3)
CLINK VTSS=VTSS3 CHANIF=0G PART=VTSS1
CLINK VTSS=VTSS3 CHANIF=0G PART=VTSS2
```

带有 TCP/IP CLINK 的 VSM5 到 VSM5 群集

图 A.7. 带有 TCP/IP CLINK 的群集 VSM5

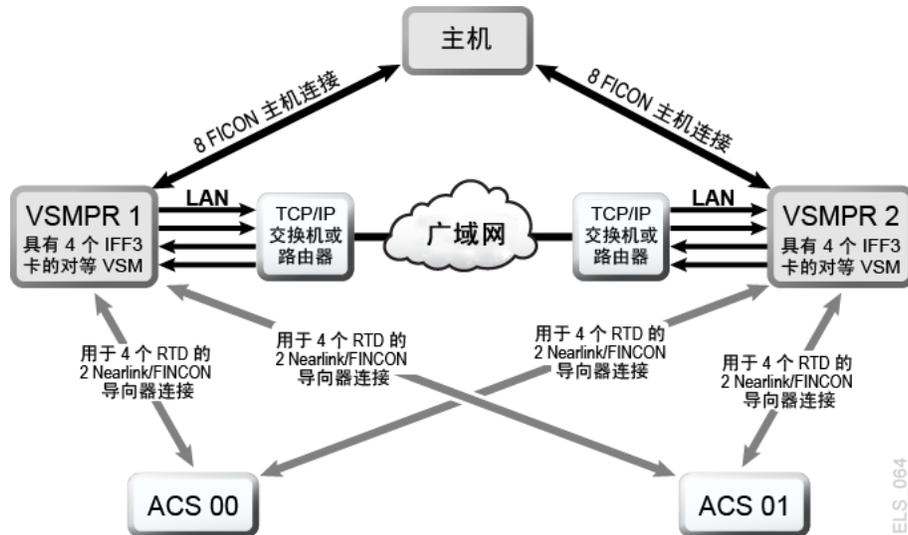


图 A.7 “带有 TCP/IP CLINK 的群集 VSM5” 显示了带有 TCP/IP CLINK 的 VSM5 到 VSM5 群集示例。

在图 A.7 “带有 TCP/IP CLINK 的群集 VSM5” 中，假定为了实现冗余，您将每个 VSM5 上单独 IFF 卡上的目标用于本机 IP，如表 A.1 “VSMPR1 的 CLINK IPIF 值” 和表 A.2 “VSMPR2 的 CLINK IPIF 值” 中所示。

表 A.1. VSMPR1 的 CLINK IPIF 值

| IFF 卡 | 目标编号 | 示例 IP | 相应的 CLINK IPIF |
|-------|------|-----------|----------------|
| IFF0 | 目标 0 | 128.0.1.1 | 0A:0 |
| IFF1 | 目标 0 | 128.0.2.1 | 0I:0 |
| IFF2 | 目标 0 | 128.0.3.1 | 1A:0 |
| IFF3 | 目标 0 | 128.0.4.1 | 1I:0 |

表 A.2. VSMPR2 的 CLINK IPIF 值

| IFF 卡 | 目标编号 | 示例 IP | 相应的 CLINK IPIF |
|-------|------|-----------|----------------|
| IFF0 | 目标 0 | 128.0.1.2 | 0A:0 |
| IFF1 | 目标 0 | 128.0.2.2 | 0I:0 |
| IFF2 | 目标 0 | 128.0.3.2 | 1A:0 |
| IFF3 | 目标 0 | 128.0.4.2 | 1I:0 |

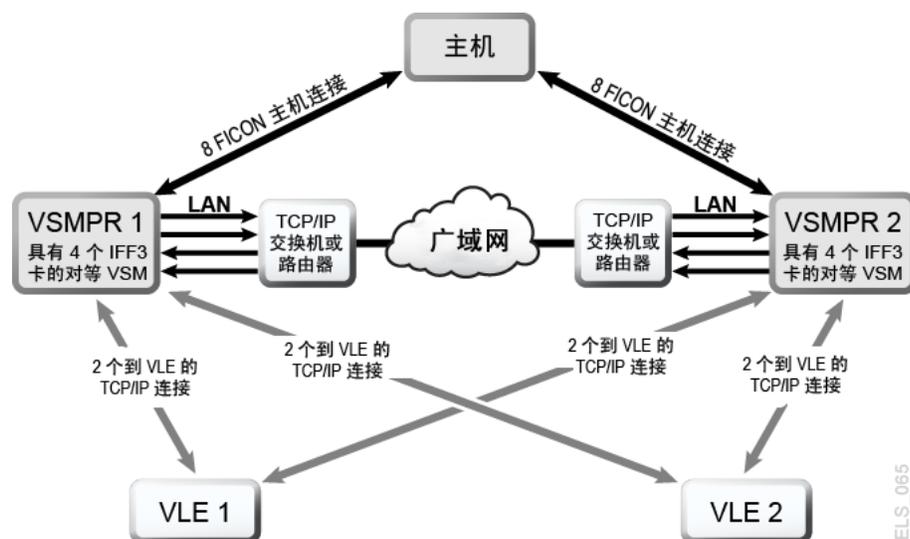
以下示例显示了使用 CONFIG JCL 以表 A.1 “VSMPR1 的 CLINK IPIF 值” 和表 A.2 “VSMPR2 的 CLINK IPIF 值” 中所示的值定义图 A.7 “带有 TCP/IP CLINK 的群集 VSM5” 中所示的配置。

```
//CREATECF EXEC PGM=SLUADMIN, PARM='MIXED'
//STEPLIB DD DSN=h1q.SEALINK, DISP=SHR
//SLSCNTL DD DSN=FEDB.VSMLMULT.DBASEPRM, DISP=SHR
//SLSCNTL2 DD DSN=FEDB.VSMLMULT.DBASESEC, DISP=SHR
//SLSSTBY DD DSN=FEDB.VSMLMULT.DBASETBY, DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
CONFIG CDSLEVEL(V61ABOVE)
GLOBAL MAXVTV=32000 MVCFREE=40 VTVATTR=SCRATCH RECALWER=YES
LOCKSTR=VTCS_LOCKS REPLICAT=ALWAYS VTVPAGE=LARGE INITMVC=YES
SYNCHREP=YES MAXRTDS=16 FASTMIGR=YES
RECLAIM THRESHLD=70 MAXMVC=40 START=35
VTSS NAME=VSMR1 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
VTD LOW=8900 HIGH=89FF
RTD NAME=VPR12A00 DEVNO=2A00 CHANIF=0C:0
RTD NAME=VPR12A01 DEVNO=2A01 CHANIF=0C:1
RTD NAME=VPR12A02 DEVNO=2A02 CHANIF=0C:2
RTD NAME=VPR12A03 DEVNO=2A03 CHANIF=0C:3
RTD NAME=VPR12A04 DEVNO=2A04 CHANIF=0G:0
RTD NAME=VPR12A05 DEVNO=2A05 CHANIF=0G:1
RTD NAME=VPR12A06 DEVNO=2A06 CHANIF=0G:2
RTD NAME=VPR12A07 DEVNO=2A07 CHANIF=0G:3
VTSS NAME=VSMR2 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
VTD LOW=9900 HIGH=99FF
RTD NAME=VPR22B00 DEVNO=2B00 CHANIF=0C:0
RTD NAME=VPR22B01 DEVNO=2B01 CHANIF=0C:1
RTD NAME=VPR22B02 DEVNO=2B02 CHANIF=0C:2
RTD NAME=VPR22B03 DEVNO=2B03 CHANIF=0C:3
RTD NAME=VPR22B04 DEVNO=2B04 CHANIF=0G:0
RTD NAME=VPR22B05 DEVNO=2B05 CHANIF=0G:1
RTD NAME=VPR22B06 DEVNO=2B06 CHANIF=0G:2
RTD NAME=VPR22B07 DEVNO=2B07 CHANIF=0G:3
CLUSTER NAME=CLUSTER1 VTSSs(VSMR1, VSMR2)
CLINK VTSS=VSMR1 IPIF=0A:0
CLINK VTSS=VSMR1 IPIF=0I:0
CLINK VTSS=VSMR1 IPIF=1A:0
CLINK VTSS=VSMR1 IPIF=1I:0
CLINK VTSS=VSMR2 IPIF=0A:0
CLINK VTSS=VSMR2 IPIF=0I:0
CLINK VTSS=VSMR2 IPIF=1A:0
CLINK VTSS=VSMR2 IPIF=1I:0
```

带有 TCP/IP CLINK 和交叉连接 VLE 的 VSM5 到 VSM6 群集

图 A.8 “带有 TCP/IP CLINK 和交叉连接 VLE 的 VSM5 到 VSM6 群集” 显示了带有 TCP/IP CLINK 的 VSM5 到 VSM6 群集示例，其中每个 VTSS 交叉连接到两个 VLE。

图 A.8. 带有 TCP/IP CLINK 和交叉连接 VLE 的 VSM5 到 VSM6 群集



在图 A.8 “带有 TCP/IP CLINK 和交叉连接 VLE 的 VSM5 到 VSM6 群集” 中，假定为了实现冗余，您将 VSM5 (VSMPR1) 的单独 IFF 卡上的目标用于本机 IP 和 VLE 连接，如表 A.3 “VSMPR1 的 CLINK IPIF 值” 和表 A.4 “VSMPR1 的 RTD IPIF 值” 中所示。

表 A.3. VSMPR1 的 CLINK IPIF 值

| IFF 卡 | 目标编号 | 示例 IP | 相应的 CLINK IPIF |
|-------|------|-----------|----------------|
| IFF0 | 目标 0 | 128.0.1.1 | 0A:0 |
| IFF1 | 目标 0 | 128.0.2.1 | 0I:0 |
| IFF2 | 目标 0 | 128.0.3.1 | 1A:0 |
| IFF3 | 目标 0 | 128.0.4.1 | 1I:0 |

表 A.4. VSMPR1 的 RTD IPIF 值

| IFF 卡 | 目标编号 | 示例 IP | 相应的 CLINK IPIF |
|-------|------|-----------|----------------|
| IFF0 | 目标 1 | 128.0.1.2 | 0A:1 |
| IFF1 | 目标 1 | 128.0.2.2 | 0I:1 |
| IFF2 | 目标 1 | 128.0.3.2 | 1A:1 |
| IFF3 | 目标 1 | 128.0.4.2 | 1I:1 |

以下示例显示了使用 CONFIG JCL 以表 A.3 “VSMPR1 的 CLINK IPIF 值” 和表 A.4 “VSMPR1 的 RTD IPIF 值” 中所示的值定义图 A.8 “带有 TCP/IP CLINK 和交叉连接 VLE 的 VSM5 到 VSM6 群集” 中所示的配置。

```
//CREATECF EXEC PGM=SLUADMIN, PARM='MIXED'
//STEPLIB DD DSN=h1q.SEALINK, DISP=SHR
//SLSCNTL DD DSN=FEDB.VSMLMULT.DBASEPRM, DISP=SHR
```

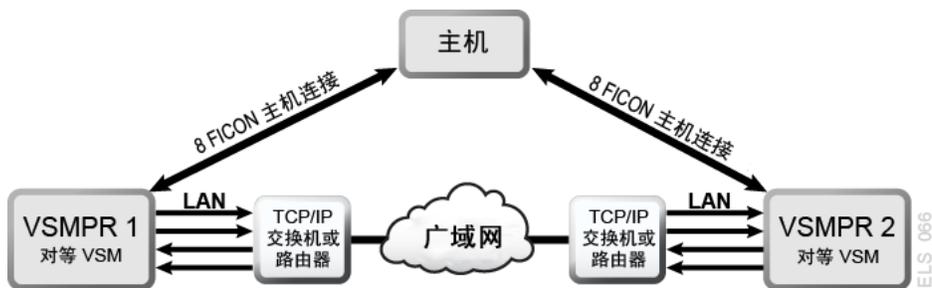
```
//SLSCNTL2 DD DSN=FEDB.VSMLMULT.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=FEDB.VSMLMULT.DBASEBY,DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
CONFIG CDSLEVEL(V61ABOVE)
GLOBAL MAXVTV=32000 MVCFREE=40 VTVATTR=SCRATCH RECALWER=YES
LOCKSTR=VTCS_LOCKS REPLICAT=ALWAYS VTPAGE=LARGE INITMVC=YES
SYNCHREP=YES MAXRTDS=16 FASTMIGR=YES
RECLAIM THRESHLD=70 MAXMVC=40 START=35
VTSS NAME=VSMPR1 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
VTD LOW=9900 HIGH=99FF
RTD NAME=VL1RTD1 STORMNGR=VLE1 IPIF=0A:1
RTD NAME=VL1RTD2 STORMNGR=VLE1 IPIF=0I:1
RTD NAME=VL2RTD1 STORMNGR=VLE2 IPIF=1A:1
RTD NAME=VL2RTD2 STORMNGR=VLE2 IPIF=1I:1
CLUSTER NAME=CLUSTER1 VTSSs(VSMPR1,VSMPR2)
CLINK VTSS=VSMPR1 IPIF=0A:0
CLINK VTSS=VSMPR1 IPIF=0I:0
CLINK VTSS=VSMPR1 IPIF=1A:0
CLINK VTSS=VSMPR1 IPIF=1I:0
CLINK VTSS=VSMPR2 IPIF=0A:0
CLINK VTSS=VSMPR2 IPIF=0I:0
CLINK VTSS=VSMPR2 IPIF=1A:0
CLINK VTSS=VSMPR2 IPIF=1I:0
```

请注意，在此示例中，VSM5 (VSMPR1) 的 *CLINK IPIF* 和 *RTD IPIF* 参数值必须与表 A.3 “VSMPR1 的 CLINK IPIF 值” 和表 A.4 “VSMPR1 的 RTD IPIF 值” 中所显示的值匹配，而 VSM6 (VSMPR2) 的 *CLINK IPIF* 和 *RTD IPIF* 值只需满足对这些值的 VTCS 限制，并且对每个 VTSS 必须唯一；它们并不对应于 VSM6 TCP/IP 端口上的实际值。

带有 TCP/IP CLINK 的 VSM6 到 VSM6“无磁带”群集

图 A.9 “带有 TCP/IP CLINK 的 VSM6 到 VSM6 无磁带群集” 显示了带有 TCP/IP CLINK 的“无磁带”VSM6 到 VSM6 群集示例。

图 A.9. 带有 TCP/IP CLINK 的 VSM6 到 VSM6 无磁带群集



以下示例显示了使用 CONFIG JCL 定义图 A.9 “带有 TCP/IP CLINK 的 VSM6 到 VSM6 无磁带群集” 中所显示的配置。

```
//CREATECF EXEC PGM=SLUADMIN,PARM='MIXED'
//STEPLIB DD DSN=h1q.SEALINK,DISP=SHR
```

```

//SLSCNTL DD DSN=FEDB.VSMLMULT.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=FEDB.VSMLMULT.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=FEDB.VSMLMULT.DBASETBY,DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
CONFIG CDSLEVEL(V61ABOVE)
GLOBAL MAXVTV=32000 MVCFREE=40 VTVATTR=SCRATCH RECALWER=YES
LOCKSTR=VTCS_LOCKS REPLICAT=ALWAYS VTVPAGE=LARGE INITMVC=YES
SYNCHREP=YES MAXRTDS=16 FASTMIGR=YES
RECLAIM THRESHLD=70 MAXMVC=40 START=35
VTSS NAME=VSMR1 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
VTD LOW=8900 HIGH=89FF
CLUSTER NAME=CLUSTER1 VTSSs(VSMR1,VSMR2)
CLINK VTSS=VSMR1 IPIF=0A:0
CLINK VTSS=VSMR1 IPIF=0A:1
CLINK VTSS=VSMR1 IPIF=0A:2
CLINK VTSS=VSMR1 IPIF=0A:3
CLINK VTSS=VSMR2 IPIF=0A:0
CLINK VTSS=VSMR2 IPIF=0A:1
CLINK VTSS=VSMR2 IPIF=0A:2
CLINK VTSS=VSMR2 IPIF=0A:3

```

请注意，在此示例中，两个 VSM6 的 *CLINK IPIF* 值只需满足对这些值的 VTCS 限制，并且对每个 VTSS 必须唯一；它们并不对应于 VSM6 TCP/IP 端口上的实际值。另请注意，由于群集无磁带，因此两个 VSM6 都没有 *CONFIG RTD* 语句。

应该使用单向还是双向群集？

您可以使用 *VTSSLST* 和 *VTSSSEL* 语句使双向群集变为单向群集。您为什么要这样做呢？如果您要切换主 VTSS 与辅助 VTSS 的角色，应该怎样做呢？首先执行“[配置和管理双向群集系统](#)”中所述过程中的前几步设置。完成第 5 步后，使用以下 *VTSSLST* 和 *VTSSSEL* 语句进行更改。

```

VTSSLST NAME(SITEA) VTSS(VSMR1)
VTSSSEL FUNCTION(SCRATCH) HOST(MVSA) VTSSLST(SITEA)
VTSSSEL FUNCTION(SPECIFIC) HOST(MVSA) VTSSLST(SITEA)

```

在此示例中：

- *VTSSLST* 语句定义仅包含 VSMR1 的 VTSS 列表 SITEA。
- *VTSSSEL* 语句指引从 MVSA 到 SITEA（仅包含 VSMR1）的暂存和特定 VTV 挂载，实际上会使其成为主 VTSS。

所以，该群集事实上是双向群集，但是借助 *VTSSLST* 和 *VTSSSEL* 语句，可以灵活地实际上使任一 VTSS 成为主 VTSS 而使另一 VTSS 成为辅助 VTSS，只需使用 *MGMTDEF* 命令装入相应的 *MGMTclas*、*STORclas*、*VTSSLST* 和 *VTSSSEL* 控制语句即可。

如果您要切换主 VTSS 与辅助 VTSS，应该怎样做呢？在这种情况下，请重写 *VTSSLST* 和 *VTSSSEL* 语句，以使 VSMR2 成为主 VTSS 而使 VSMR1 成为辅助 VTSS。

```
VTSSLST NAME(SITEB) VTSS(VSMR2)
VTSSSEL FUNCTION(SCRATCH) HOST(MVSB) VTSSLST(SITEB)
VTSSSEL FUNCTION(SPECIFIC) HOST(MVSB) VTSSLST(SITEB)
```

在此示例中：

- *VTSSLST* 语句定义仅包含 *VSMR2* 的列表 *SITEB*。
- *VTSSSEL* 语句指引从 *MVSB* 到 *SITEB*（仅包含 *VSMR2*）的暂存和特定 *VTV* 挂载，实际上会使其成为主 *VTSS*。

最后，如果该群集作为真正的双向群集效果更好，应该怎样做呢？在这种情况下，请删除 *VTSSLST* 和 *VTSSSEL* 语句并重新装入定义。

词汇表

注：

词汇表中含有 (I) 标记的条目均源自《IBM Dictionary of Computing》（《IBM 计算技术词典》）。

| | |
|---|--|
| 4410 | Oracle StorageTek 标准磁带库存储模块 (Library Storage Module, LSM)。 |
| 4480 | Oracle StorageTek 18 磁轨 1/2 英寸磁带传送装置。 |
| 4490 | 带 ESCON 支持的 Oracle StorageTek 36 磁轨长磁带传送装置。也称为 Silverton。 |
| 9310 | Oracle StorageTek 磁带库存储模块 (Library Storage Module, LSM)，标准 4410 LSM 的高性能版本。也称为 PowderHorn。 |
| 9360 | Oracle StorageTek 磁带库存储模块 (Library Storage Module, LSM)。也称为 WolfCreek。 |
| 9740 | Oracle StorageTek 磁带库存储模块 (Library Storage Module, LSM)。也称为 TimberWolf。 |
| Abnormal end of task, abend (任务异常终止) | 终止了计算机处理任务的软件或硬件问题。 |
| ACS | Automated Cartridge System (自动化磁带系统)，这是一个全自动化的磁带存储和检索磁带库子系统，包括一个或多个通过直通端口连接的磁带库存储模块 (Library Storage Module, LSM)。 |
| ACS library (ACS 磁带库) | 一个磁带库包含一个或多个自动化磁带系统 (Automated Cartridge System, ACS)、连接的盒式磁带机以及 ACS 中的磁带盒。 |
| ACSid | LIBGEN 过程中使用的一种方法，通过使用十进制数字 00 至 99 来标识 ACS。 |
| ACSLs | Automated Cartridge System Library Software，这是一个 Oracle StorageTek 磁带库控制软件，在基于 UNIX® 的磁带库控制系统中运行。 |
| address (地址) | 硬件 ID 或者数据的目标或起源的编码表示方式。 |
| allocation (分配) | 将资源分配到特定的任务。 |
| asynchronous transmission (异步传输) | 面向字符的数据传输（不同于 IBM 的数据块模式传输）。 |

| | |
|--|--|
| Automated Cartridge System Library Software (ACSLs) | 请参见 ACSLs 。 |
| Automated Cartridge System, ACS (自动化磁带系统) | 请参见 ACS 。 |
| automatic mode (自动模式) | LSM 与所有连接的主机之间的关系。以自动模式运行的 LSM 处理磁带时不需要操作员干预。这是经过联机修改的 LSM 的正常运行模式。相反的情况是 "manual mode" (手动模式)。请参见 "manual mode" (手动模式)。 |
| bar code (条形码) | 由一系列不同宽度的条形组成的代码。此代码出现在附加到盒式磁带脊上的外部标签上，等效于卷序列号 (volser)。机械手的机器视觉系统可读取此代码。 |
| BISYNC | Binary Synchronous Communications (二进制同步通信)。IBM 开发的一种早期低级协议，用于在同步通信链路上传输数据。这是一种数据传输形式，其中字符的同步由发送和接收站的计时信号控制。 |
| CAPid | CAPid 按 CAP 所在的 LSM 唯一定义 CAP 的位置。CAPid 的形式为 AA:LL:CC，其中 AA:LL 是 LSMid，CC 是两位数的 CAP 编号。 |
| Cartridge Access Port, CAP (磁带存取口) | 一种组件，允许在无人进入 LSM 的情况下，在 LSM 中插入或弹出多个磁带。 |
| cartridge drive, CD (盒式磁带机) | 一种硬件设备，其中包含两个或四个磁带传送装置以及关联的电源和气流。 |
| cartridge tape I/O driver (盒式磁带 I/O 驱动程序) | 向磁带子系统发出命令 (例如，读取、写入和倒带) 的操作系统软件。它是连接特定类型的控制单元的软件焦点。(例如，Oracle 的 StorageTek CARTLIB 产品。) |
| cartridge (盒式磁带) | 磁带的塑料外壳。它大约为 4 英寸 (100 毫米) 宽，5 英寸 (125 毫米) 长，1 英寸 (25 毫米) 厚。在传输中装载磁带时，磁带会自动穿线。塑料导销与磁带连接，用于自动穿线。磁带盒脊上有一个 OCR/条形码友标签，列出 VOLSER (磁带卷标识符)。 |
| cell (单元) | LSM 中存储单个磁带盒的插座。 |
| CGI | Common Gateway Interface (通用网关接口) |
| channel-to-channel, CTC (通道到通道) | 是指在通道到通道适配器的对侧上的程序之间的通信 (数据传输)。(I) |
| channel (通道) | 将主机和主存储与输入和输出设备的控制单元连接的设备。全双工通道有两个路径 (即 2 条线路，或者一条线路有两个频率的信号)。半双工通道需要一个端口接收，同时另一个端口传输。 |

| | |
|---|--|
| client link (客户机链路) | LCS 与客户机之间的通信链路。 |
| Client System Component, CSC (客户机系统组件) | 提供客户机计算系统的操作系统与 StorageTek 磁带库控制系统 (LCS) 之间的接口的软件。 |
| client system (客户机系统) | LCS 向其提供 StorageTek 自动化磁带系统接口的系统。 |
| client (客户机) | 磁带库控制系统提供的 ACS 服务的终极用户。 |
| client/server (客户机/服务器) | 分布式系统中的一种交互模型, 在该模型中, 位于一个站点的程序向位于另一站点的程序提供请求服务并等待响应。发出请求的程序称为客户机, 使请求得以满足的程序称为服务器。 |
| coaxial cable (同轴电缆) | 一种在网络数据传输中以同步通信方式传输的介质, 与双绞线相对, 后者是异步 RS-232 通信的主要介质。 |
| complex (复合磁带库) | 一种由其他系统 (具体是指 ACS 服务器系统和客户机系统) 组成的系统。 |
| connected mode (已连接模式) | 主机与 ACS 之间的关系。在这种模式下, 主机和某个 ACS 可以通信 (是指至少有一个到此 ACS 的站联机)。 |
| connection number (连接编号) | 通信路径在服务器上的唯一标识符。该编号按 TCP/IP 分配, 用于标识服务器节点与服务器上某个特定端口之间以及客户机节点与客户机上的某个特定端口之间的唯一连接。只要存在连接, 就会存在连接编号, 并且仅在存在连接时才存在连接编号。 |
| console (控制台) | 用于在系统上控制会话的主要 I/O 设备。 |
| control data set, CDS (控制数据集) | 主机软件用于控制自动化磁带库功能的数据集。也称为磁带库数据库。 |
| Control Path Adaptor, CPA (控制路径适配器) | 一种 Bus-Tech, Inc. 硬件设备, 允许在主机处理器的块多路复用器通道与局域网之间进行通信。 |
| Control Unit, CU (控制单元) | 一种基于微处理器的装置, 位于通道与 I/O 设备之间。它可将通道命令转换成设备命令, 并将设备状态发送给通道。 |
| coupling facility channel (耦合工具通道) | 一种高带宽光纤通道, 可提供耦合工具与直接与其连接的中央处理器复合磁带库之间的数据共享所需的高速连接。(I) |
| coupling facility (耦合工具) | 一种特殊的逻辑分区, 可在 sysplex 中提供高速缓存、列表处理和锁定功能。(I) |

| | |
|--|--|
| CTC | Channel-to-channel (通道到通道)。 |
| Data Path Adapter (数据路径适配器) | 一种硬件设备, 可以将客户机计算系统的数据协议转换为 StorageTek 控制单元或 IMU 的数据协议。例如, DEC 的 TC44-AA/BA STI-to-4400 ACS Interconnect。 |
| data set (数据集) | 视为一个单元的一组记录。 |
| data sharing (数据共享) | 并行子系统或应用程序直接访问和更改相同的数据同时维护数据完整性的能力。(I) |
| device number (设备编号) | 一个四位数的十六进制编号, 用于唯一标识与处理器连接的设备。 |
| device preferencing (设备首选) | 首选某个 36 磁轨传送装置类型而不选择其他 36 磁轨传送装置类型的过程。 |
| device separation (设备隔离) | 请参见 drive exclusion (磁带机排除) 。 |
| DFSMS | Data Facility Storage Management Subsystem (数据工具存储管理子系统)。 |
| direct access storage device, DASD (直接访问存储设备) | IBM 的磁盘磁带机存储设备术语。 |
| directed allocation (定向分配) | 请参见 "drive prioritization" (磁带机优先级确定)。 |
| disconnected mode (断开连接模式) | 主机与 ACS 之间的关系。在这种模式下, 主机和 ACS 不能通信 (没有到此 ACS 的联机站)。 |
| dotted-decimal notation (点分十进制表示法) | 32 位整数的语法表示形式, 该整数由四个以十进制表示的 8 位数字组成, 它们之间用句点 (点) 分隔。在 TCP/IP 描述中, 点分十进制表示法用于 Internet 地址。 |
| drive exclusion (磁带机排除) | (以前称为设备隔离) 是指根据 SMC 排除标准将磁带机从分配请求中排除的 SMC 功能。 |
| drive panel (磁带机面板) | 包含磁带传送装置的 LSM 挡板。T9840 传送装置的磁带机面板可以包含 10 个或 20 个传送装置。非 T9840 传送装置的磁带机面板最多可以包含 4 个传送装置。 |
| drive prioritization (磁带机优先级确定) | (以前称为定向分配) 是指根据包括卷位置在内的 SMC 分配标准影响特定磁带机选择的 SMC 功能。 |
| Dual LMU (双 LMU) | 提供冗余 LMU 能力的一种硬件/微代码功能。 |

| | |
|--|---|
| dump (转储) | 在时间 t 时主存储内容的打印表示形式。该表示形式用于调试目的。 |
| dynamic server switching (动态服务器切换) | 当活动服务器上发生系统故障时切换服务器处理器的功能。 |
| ECART | Enhanced Capacity Cartridge (增强容量盒式磁带)。 |
| Enhanced Capacity Cartridge (增强容量盒式磁带) | 长度为 1100 英尺的盒式磁带, 仅可在 36 磁轨传送装置 (即 4490、9490 和 9490EE) 上使用。 |
| Enterprise Systems Connection, ESCON (企业系统连接) | 一组提供使用光缆作为传输介质的动态连接的环境的产品和服务。(I) |
| error codes, EC (错误代码) | 通过消息显示的数字代码, 指示导致错误的问题的类型。 |
| error recovery procedures, ERP (错误恢复过程) | 设计用于帮助隔离错误以及在可能的情况下从设备错误进行恢复的过程。 |
| ESCON | Enterprise Systems Connection (企业系统连接)。 |
| esoteric name (专用设备组名称) | 分配给具有相同设备类型的传送装置的名称。 |
| Ethernet (以太网) | 一种 LAN 体系结构, 所用的总线拓扑允许各种计算机连接到常用的屏蔽同轴干线。以太网体系结构类似于 IEEE 802.3 标准。 |
| event control block, ECB (事件控制块) | 提供一个区域供在操作完成时存储完成代码。 |
| File Transfer Protocol, FTP (文件传输协议) | 一种 TCP/IP 命令, 提供一种在通过 TCP/IP 连接的计算机之间传输文件的方式。 |
| file (文件) | 视为一个单元的一组相关记录。 |
| foreign socket (外部套接字) | 面向 TCP/IP 连接的协议中的两个端点之一。指定可以连接到服务器的外部主机的地址。 |
| GB | 1,073,741,834 字节的存储 |
| Handshake (握手) | 由一个进程发送到另一个进程的控制流信号。 |
| helical cartridge (螺旋式磁带) | 一种高容量的螺旋式扫描磁带, 最多可保存 50GB 未压缩数据。这种磁带只能在 RedWood (SD-3) 传送装置上使用。 |

| | |
|--|---|
| host computer (主机) | 对计算机网络进行控制的计算机。 |
| HTTP | Hypertext Transfer Protocol (超文本传输协议)。 |
| ICRC | Improved Cartridge Recording Capacity (改进的磁带录制功能)。一种压缩精简功能, 可以提高 1/2 英寸磁带上数据存储量。 |
| IEEE 802.3 | 由 IEEE 制定并在全球范围内接受的一种标准, 适用于采用 CSMA/CD (Carrier Sense Multiple Access with Collision Detection, 带冲突检测的载波侦测多路访问) 的局域网。 |
| initial program load, IPL (初始程序装入) | 激活计算机复位的过程。 |
| Interactive Storage Management Facility (交互式存储管理工具) | 用于定义 DFSMS/MVS 存储组和类的一系列应用程序。 |
| Internet address (Internet 地址) | 用于指定网络或该网络上的主机以便进行 TCP/IP 通信的编码系统。标准 Internet 地址表示法为点分十进制格式。 |
| Internet | 使用 TCP/IP 的网络集合, 作为一个虚拟网络运行。 |
| Internet Protocol, IP (Internet 协议) | 消息的形式描述和两个网络用来交换消息的规则。 |
| ISMF | Interactive Storage Management Facility (交互式存储管理工具) |
| JES | Job entry subsystem (作业输入子系统)。(I) |
| JES2 | 一种 MVS 子系统, 用于将作业接收到系统中, 将其转换为内部格式, 选择它们进行执行, 处理其输出, 以及将其从系统中清除。在包含多个处理器的安装中, 每个 JES2 处理器都可以独立地控制其作业输入、调度和输出处理。另请参见 JES3。(I) |
| JES3 | 一种 MVS 子系统, 用于将作业接收到系统中, 将其转换为内部格式, 选择它们进行执行, 处理其输出, 以及将其从系统中清除。在包含多个松散耦合处理单元的联合体中, JES3 程序对处理器进行管理, 以便全局处理器对本地处理器进行集中控制, 通过一个公用的作业队列为其分配作业。另请参见 JES2。(I) |
| job control language, JCL (作业控制语言) | 一种面向问题的语言, 用于描述对操作系统的作业处理要求。 |
| LAN | Local Area Network (局域网), 小型 (局部) 地理区域中的网络 |
| LCS | Library Control System, LCS (磁带库控制系统) |

| | |
|--|--|
| LCS processor console (LCS 处理器控制台) | 磁带库控制系统处理器控制器用于控制 VM 操作系统 (对于基于 VM 的 LCS)。 |
| LCU | 请参见 "Library Control Unit" (磁带库控制单元)。 |
| LIBGEN | 将磁带库的配置定义到 VM/HSC 的过程。 |
| library cartridge transport (磁带库磁带传送装置) | 请参见 "transport" (传送装置)。 |
| library complex (复合磁带库) | 复合磁带库包含一个 HSC 控制数据集 (Control Data Set, CDS), 并且最多可包含 256 个自动磁带系统 (Automatic Cartridge System, ACS), 其中每个自动磁带系统最多可包含 24 个磁带库存储模块 (Library Storage Module, LSM)。 |
| library control component (磁带库控制组件) | ACS 中用于控制磁带的挂载和卸载的软件。 |
| library control platform (磁带库控制平台) | 为磁带库控制系统提供恰当环境的硬件和软件。 |
| library control processor (磁带库控制处理器) | 经过恰当配置、用于支持磁带库控制系统的运行的计算机硬件。 |
| Library Control Software (磁带库控制软件) | 一种磁带库控制组件、客户机系统界面以及磁带库工具。 |
| Library Control System, LCS (磁带库控制系统) | 磁带库控制平台和磁带库控制软件。 |
| Library Control Unit, LCU (磁带库控制单元) | LSM 中控制机械手移动的部分。 |
| library database (磁带库数据库) | 一种包含有关可移动介质卷的位置和状态的信息 (如单元位置、暂存状态等) 的文件或数据集。也称为控制数据集 (control data set, CDS)。 |
| library drive (磁带库磁带机) | ACS 中的一种不同于独立盒式磁带机的盒式磁带机。 |
| Library Management Unit, LMU (磁带库管理单元) | 请参见 LMU 。 |

| | |
|--|---|
| library mode (磁带库模式) | 4400 自动化磁带系统中 4480 磁带子系统的运行，与操作员将磁带插入传送装置的手动模式相对。请参见 "manual mode" (手动模式)。 |
| Library Storage Module, LSM (磁带库存储模块) | 请参见 LSM 。 |
| library (磁带库) | 请参见 TapePlex。 |
| LMU | Library Management Unit (磁带库管理单元)，对一个或多个 LSM/LCU 的活动进行协调的硬件或软件产品。 |
| local area network, LAN (局域网) | 请参见 LAN 。 |
| local port (本地端口) | 可供支持 TCP/IP 的主机处理器使用的众多应用程序或进程之一的目的地。 |
| local socket (本地套接字) | 支持 TCP/IP 的主机的网络地址和应用程序进程的特定端口的地址组合。 |
| logical port, LP (逻辑端口) | 请参见 LP 。 |
| LP | logical port (逻辑端口)，与客户机系统连接的 CLS 软件。CLSLP 是用于在客户机系统与 VM/HSC 之间传递数据的软件组件之一。 |
| LSM | Library Storage Module (磁带库存储模块)，标准 LSM (4410)，一种十二端结构，存储空间最多可容纳 6000 个左右的盒式磁带。它还包含一个可在存储单元与连接的传送装置之间移动磁带的独立式视觉辅助机械手。另请参见 PowderHorn、SL3000、SL8500 和 WolfCreek。 |
| LSM number (LSM 编号) | 一种用来标识 LSM 的方法。LSM 编号是在 LIBGEN 过程中定义 SLIACS 宏 LSM 参数的结果。此参数中列出的第一个 LSM 会获得 LSM 编号 00 (十进制)，列出的第二个 LSM 获得编号 01，依此类推，直到所有 LSM 都得到标识为止 (最大编号 99)。 |
| LSMId | LSMId 包含联接到 (并联) LSM 编号的 ACSId。 |
| manual mode (手动模式) | 在 ACS 之外操作磁带机。请参见 library mode (磁带库模式)。 |
| master LMU (主 LMU) | 在双 LMU 配置中当前控制 ACS 正常工作的 LMU。 |
| mixed configuration (混合配置) | 一种同时包含处于手动模式和磁带库模式的不同类型的磁带机的配置。 |
| modem (调制解调器) | 一种用于通过模拟传输工具传输数字化数据的设备。 |

| | |
|---|---|
| multi-client (多客户机) | 多个 (同构或异构) 客户机系统连接到一个 LCS 的环境。 |
| MVS system console (MVS 系统控制台) | MVS/CSC 通过 MVS 系统控制台提供了操作员界面。 |
| OCR label (OCR 标签) | Optical character recognition (光学字符识别) 标签。一种附加在磁带盒脊上的外部标签, 可人读, 也可机读。 |
| operating system, OS (操作系统) | 一种软件, 对有助于便利总体系统运营的程序执行进行控制。 |
| operator console (操作员控制台) | 在本文档中, 操作员控制台是指 MVS 客户机系统控制台。 |
| Pass-thru Port, PTP (直通端口) | 一种机制, 允许磁带在多 LSM ACS 中从一个 LSM 传递到另一个 LSM。 |
| physical port (物理端口) | 支持服务器/客户机链路所需的通信硬件。 |
| physical volume (物理卷) | 数据文件介质的物理绑定单元。请参见 "cartridge" (盒式磁带)。 |
| PowderHorn (9310) | 标准 LSM 的高性能版本。 |
| preconfigured package (预配置包) | 一种存储服务器包, 其中包括供应商提供的所有硬件、软件和配置参数设置。 |
| product change request, PCR (产品更改请求) | 对产品增强功能的请求。通常情况下, 此请求来自客户, 但也可能来自 Oracle。 |
| program temporary fix, PTF (临时修复程序) | 一种用于修复一个或一系列缺陷的软件版本。 |
| program update tape, PUT (程序更新磁带) | 包含 MVS/CSC 系统软件的更新或新版本的一个或多个磁带。 |
| protocol (协议) | 消息形式的形式描述以及两台或更多计算机交换这些消息时必须遵循的规则。 |
| recovery (恢复) | 自动或手动解决服务器系统中的问题的过程。 |
| reel-id (卷轴 ID) | 特定磁带卷的标识符。等效于卷序列号 (volume serial number, VOLSER)。 |
| Request (请求) | 一个术语, 指的是为执行磁带相关功能而向 4400 ACS 发出的命令。 |

| | |
|---|--|
| scratch tape subpool (暂存磁带子池) | 所有暂存磁带的子集。子池由具有类似物理特征的一组或多组卷序列号组成；这些物理特征包括卷类型（卷轴或磁带盒）、卷轴大小、长度、物理位置等。有些安装还可以按其他特征（如标签类型）进一步划分其暂存池。 |
| scratch tape (暂存磁带) | 因没有所有者而可以提供给任何用户的磁带。 |
| SD-3 | Oracle StorageTek 螺旋式磁带传送装置。也称为 RedWood。 |
| server (服务器) | ELS 磁带库控制系统，如 HSC。在 SMC 中，服务器是用到指定的 TAPEPLEX 的指定 SERVER 路径表示的。虽然在远程主机上 SMC HTTP 服务器组件是必需的中间件，但是考虑到 SMC 来说，服务器被视为在远程主机上运行的 ELS 磁带库控制系统。 |
| SL3000 | Oracle StorageTek SL3000 模块化磁带库，它提供了混合介质、逻辑和物理分区功能、高级管理以及高可用性。它支持包括大型机和开放系统的混合环境，可以从 200 个磁带插槽扩展到 6,000 个以内的磁带插槽。 |
| SL8500 | Oracle StorageTek SL8500 模块化磁带库，它提供了混合介质、逻辑和物理分区功能、高级管理以及大容量和高可用性。它支持包括大型机和开放系统在内的混合环境，可以从标准的 1,450 个磁带插槽扩展到 100,880 个磁带插槽的复杂配置。 |
| socket (套接字) | 网络上的某个唯一地址，加上节点地址，加上特定应用程序在特定网络上的 ID。TCP/IP 使用的一种抽象技术。 |
| standard capacity cartridge (标准容量盒式磁带) | 可在任何纵向传送装置（即 4480、4490、9490 或 9490EE）上使用的盒式磁带。 |
| standby LMU (备用 LMU) | 双 LMU 配置中的冗余 LMU，可以在主 LMU 出现故障时或者在操作员发出 SWITCH 命令时进行接管。 |
| standby (备用) | 已联机变化但连接到双 LMU ACS 的备用 LMU 的站的状态。 |
| station (站) | 主机计算机与 LMU 之间的硬件路径，VM/HSC 和 LMU 通过它来发送控制信息。 |
| Storage Management Component, SMC (存储管理组件) | IBM 的 z/OS 操作系统与 StorageTek 自动化磁带库控制系统（HSC 和 MVS/CSC）之间的软件接口。SMC 为 ELS 解决方案执行分配处理、消息处理和 SMS 处理。 |
| storage server (存储服务器) | 设计用于让异构计算机系统能够使用自动化磁带库服务的一组硬件和软件产品。 |
| switchover (切换) | 备用 LMU 承担主 LMU 功能。 |

| | |
|---|--|
| synchronous LAN (同步 LAN) | 基于同步通信构建的局域网。 |
| synchronous (同步) | 请参见 BISYNC 。 |
| Systems Network Architecture, SNA (系统网络体系结构) | 一项有关用于传输信息单元以及控制网络配置和操作的逻辑结构、格式、协议和操作顺序的说明。 |
| T10000A | 能够读写 120GB 或 500GB T10000A 磁带的 Oracle StorageTek T10000 A 大容量磁带传送装置。 |
| T10000B | 能够读写 240GB 或 1TB T10000B 磁带的 Oracle StorageTek T10000 B 大容量磁带传送装置。 |
| T10000C | Oracle StorageTek T10000 C 高速/大容量磁带机，本身提供高达 252 MB/s 的速度和 5 TB 容量，最适合用于数据量不断增长的数据中心。 |
| T10000D | Oracle StorageTek T10000D 高速/大容量磁带机，本身提供高达 252 MB/s 的速度和 8.5 TB 容量，最适合用于数据保留要求不断增长的数据中心。 |
| T9840A | 能够读写 9840A 磁带的以访问为中心的 Oracle StorageTek 磁带传送装置。 |
| T9840B | 能够读写 T9840B 磁带的以访问为中心的 Oracle StorageTek 磁带传送装置。 |
| T9840C | 能够读写 T9840C 磁带的以访问为中心的 Oracle StorageTek 磁带传送装置。 |
| T9840D | 能够读写 T9840D 磁带的以访问为中心的 Oracle StorageTek 磁带传送装置。 |
| T9940A | 能够读写 60GB T9940A 磁带的以容量为中心的 Oracle StorageTek 磁带传送装置。 |
| T9940B | 能够读写 200GB T9940B 磁带的以容量为中心的 Oracle StorageTek 磁带传送装置。 |
| tape drive (磁带机) | 一种磁带处理设备，在一个机柜中最多包含四个传送装置。一个磁带机可以视为单个传送装置。 |
| tape library management system, TLMS (磁带库管理系统) | 本文档中所用的 TLMS 是指任何磁带库管理系统，而不是指 CA-1。 |

| | |
|---|--|
| TapePlex | (原“磁带库”)，一种单一硬件配置，通常用单个 HSC 控制数据集 (Control Data Set, CDS) 表示。一个 TapePlex 可以包含多个自动化磁带系统 (Automated Cartridge System, ACS) 和虚拟磁带存储子系统 (Virtual Tape Storage Subsystem, VTSS)。 |
| TCP/IP | Transmission Control Protocol/Internet Protocol (传输控制协议/Internet 协议)。 |
| trace event type (跟踪事件类型) | 在启用跟踪的情况下通过系统跟踪的事件的类型。 |
| trace file (跟踪文件) | 包含对系统调试有用的信息的文件。 |
| transaction (事务处理) | 触发特定进程执行的一组特定输入。 |
| Transmission Control Protocol, TCP (传输控制协议) | 提供全双工流服务的一种网络间标准协议。 |
| transport (传送装置) | 用来穿进、定位磁带以及从磁带进行读写的一个机电设备。 |
| UCB | Unit Control Block (单元控制块)。 |
| userid (用户标识) | 有时也称为 VM 用户标识；用户标识是用来标识特定的“虚拟机”用户或客户机的名称。 |
| utility (实用程序) | 执行对计算机系统的主功能起辅助作用的功能的程序。 |
| virtual machine, VM (虚拟机) | 请参见 VM 。 |
| Virtual Storage Manager, VSM (虚拟存储管理器) | 一种存储解决方案，可虚拟化 VTSS 缓冲区中的卷和传输，从而提高介质和传输的使用率。 |
| virtual storage (虚拟存储) | 一项操作系统功能，它根据程序需要按段 (或页) 来分配主存储要求，从而使得在表面上看来存在无限的或虚拟的存储。 |
| Virtual Tape Control System, VTCS (虚拟磁带控制系统) | 虚拟存储管理器 (Virtual Storage Manager, VSM) 解决方案的主要主机代码。此代码在单独的地址空间中运行，但与 HSC 密切通信。 |
| Virtual Tape Storage Subsystem, VTSS (虚拟磁带存储子系统) | 包含虚拟卷 (VTV) 和虚拟磁带机 (VTD) 的 DASD 缓冲区。VTSS 是一个 StorageTek RAID 6 硬件设备，其中包含启用了传送装置仿真的微代码。RAID 设备可以从磁盘读取“磁带”数据或将“磁带”数据写入到磁盘中，也可以从真实磁带机 (real tape drive, RTD) 读取数据或将数据写入到 RTD 中。 |

| | |
|--|--|
| Virtual Telecommunications Access Method, VTAM (虚拟电信访问方法) | IBM 主机驻留通信软件，用作公共的通信接口。 |
| VM | Virtual Machine (虚拟机)，对计算机及其关联的设备的功能模拟。每个虚拟机都由合适的操作系统进行控制。 |
| VM/SP 或 VM/XA | IBM 公司的一种专用操作系统，其中主要包含两个重要组件 CP 和 CMS。 |
| volume serial number, VOLSER (卷序列号) | 物理卷的标识符。 |
| volume (卷) | 作为一个单元挂载或卸载的盒式磁带 (数据载体)。 |
| WolfCreek (9360) | 容量比标准 LSM 小的高性能 LSM。 |
| ZCART | 一种扩展增强的盒式磁带，它采用了更薄的介质，可以提供两倍于增强容量 (ECART) 盒式磁带的容量。这种盒式磁带的长度为 2200 英尺，只能在 TimberLine 9490EE 36 磁轨传送装置上使用。 |

索引

A

按存储类导出和导入 MVC, 36
按存储类导入和导出 MVC, 36
按管理类导出和导入 MVC, 33
按管理类导入和导出 MVC, 33

B

保管用于 DR 的 MVC, 43
保管用于 LTR 的 MVC, 52

C

处理临时中断, 21
CDRT
 DR 测试完成后进行清理, 97
 VTCS 策略, 94
 VTV 数据的来源, 88
 优化对测试资源和生产资源的访问, 95
 使用软件, 87
 元数据注意事项, 88
 操作方案, 98
 管理资源, 89
 运行 DR 测试, 95
CDRT 的 VTCS 策略, 94

D

带有 TCP/IP CLINK 的 VSM5 到 VSM5 群集示例, 137
带有 TCP/IP CLINK 的 VSM6 到 VSM6 无磁带群集示例, 140
带有 TCP/IP CLINK 和交叉连接 VLE 的 VSM5 到 VSM6 群集示例, 138
单向群集, 80
 VTSS 示例, 123
 配置和管理示例, 125
定义恢复点目标 (Recovery Point Objective, RPO), 20
定义恢复时间目标 (Recovery Time Objective, RTO), 20

E

ELS 外部保管

准备, 41

G

更新 VTCS CONFIG 面板以定义远程磁带库, 72
管理 CDRT 资源, 89
规划 DR 测试, 27
规划数据高可用性 (Data High Availability, D-HA), 23

J

检查点
 在 VSM 环境中创建, 111
 示例, 112
将 RPO 与同步点恢复联系起来, 22
将 VLE 用于灾难恢复, 115
将特定的卷弹出至本地 (基础) 保管库, 53

K

跨 Tapeplex 复制
 VTV 只读注意事项, 56
 定义策略, 60
 当远程站点没有 LPAR 时使用, 63
 灾难恢复测试, 67
 用于实现业务恢复, 66
 用于实现业务连续性, 65
 管理通过 CTR 复制的 VTV, 69
 说明, 55
 配置, 57
 配置和启动, 58
扩展群集, 82
扩展群集示例, 133

L

利用作业监视实施同步复制, 83

P

配置远程磁带库, 71

Q

群集 VTSS
 示例, 123
群集 VTSS 配置
 使用, 75
 使用 TCP/IP 连接形成群集, 85

- 单向, 80
- 双向, 81
- 同步或异步复制, 83
- 定义, 75
- 工作原理, 78
- 扩展群集, 82
- 要求, 76

S

- 实施同步复制, 83
- 使用群集 VTSS 配置, 75
- 数据高可用性 (Data High Availability, D-HA)
 - 规划, 23
 - 高可用性物理磁带, 23
 - 高可用性虚拟磁带, 24
- 双向群集, 81
 - VTSS 示例, 128
 - 转到单向群集示例, 141
 - 配置和管理示例, 130

T

- 通过 CDRT 进行 DR 测试, 29
- 通过 VSM 跨 Tapeplex 复制进行 DR 测试, 30
- 通过物理导出和导入进行 DR 测试, 28
- 同步点恢复, 21
- 同步复制
 - 实施, 83

V

- VLE
 - DR 测试完成后进行清理, 119
 - 正常的生产模式, 115
 - 用于实现业务连续性, 120
 - 用于灾难恢复, 115
 - 运行 DR 测试, 116
- VTSS 群集系统
 - 配置和管理示例, 134

X

- 系统恢复检查点
 - 在 VSM 环境中创建, 111
- 修改 SMC SCMD5 文件, 71

Y

- 异步复制

- 利用作业监视实施, 83
- 远程磁带库
 - MVS 池注意事项, 72
 - 修改 SMC SCMD5 文件, 71
 - 更新 VTCS CONFIG 面板以定义远程磁带库, 72
 - 配置, 71
- 远程磁带库的 MVC 池注意事项, 72

Z

- 灾难恢复
 - 介绍, 19
 - 使用 CTR 测试, 67
 - 使用 VLE 清除, 119
 - 使用 VLE 运行, 116
 - 使用跨 Tapeplex 复制, 55
 - 保管 MVC, 43
 - 保管用于 LTR 的 MVC, 52
 - 同步点恢复, 21
 - 处理临时中断, 21
 - 定义恢复时间目标 (Recovery Time Objective, RTO), 20
 - 定义恢复点目标 (Recovery Point Objective, RPO), 20
 - 将 RPO 与同步点恢复联系起来, 22
 - 当 DR 站点没有 LPAR 时进行测试, 68
 - 执行真正的灾难恢复, 26
 - 规划 DR 测试, 27
 - 通过 CDRT 进行 DR 测试, 29
 - 通过 VSM 跨 Tapeplex 复制进行 DR 测试, 30
 - 通过物理导出和导入进行 DR 测试, 28
 - 在 DR 解决方案中使用跨 TapePlex 复制, 55
 - 执行物理导出和导入
 - 介绍, 33
 - 执行真正的灾难恢复, 26
 - 准备进行 ELS 外部保管, 41