



System i

可用性

通过基于任务的方法实现高可用性

V6R1





System i

可用性

通过基于任务的方法实现高可用性

V6R1

注意

在使用本资料及其支持的产品之前，请务必阅读第 161 页的『声明』中的信息。

本版本适用于 IBM i5/OS V6R1M0（产品号 5761-SS1）及所有后续发行版和修订版，直到在新版本中另有声明为止。本版本不能在所有精简指令集计算机（RISC）机型上运行，也不能在 CISC 机型上运行。

© Copyright International Business Machines Corporation 1998, 2008. All rights reserved.

目录

通过基于任务的方法实现高可用性	1
规划高可用性解决方案	1
规划应用程序弹性	2
标出弹性应用程序	2
支持集群的应用程序的 i5/OS 体系结构	2
编写高可用性集群应用程序	3
使应用程序具有弹性	3
重新启动高可用性集群应用程序	4
调用集群资源组出口程序	5
应用程序 CRG 注意事项	6
管理应用程序 CRG 对 IP 地址的接管	6
示例: 应用程序集群资源组故障转移操作	7
示例: 应用程序出口程序	8
规划数据弹性	46
确定要变为弹性数据的数据	47
规划切换式磁盘	47
切换式磁盘的硬件需求	47
切换式磁盘的软件需求	48
切换式磁盘的通信需求	48
规划跨站点镜像	48
规划地理镜像	49
规划城域镜像	52
规划全局镜像	55
规划逻辑复制	58
确定要用于逻辑复制的系统	58
集群中间件 IBM 业务合作伙伴和可用的集 群产品	58
逻辑复制的日志规划	59
逻辑复制的备份规划	59
逻辑复制的性能规划	59
规划环境弹性	59
规划集群管理域	59
规划受监视资源项 (MRE)	60
规划集群	60
集群的硬件需求	60
集群的软件需求	61
集群的通信需求	61
为集群提供网络	62
技巧: 集群通信	62
集群的性能规划	63
规划多发版集群	65
集群的性能规划	65
集群的规划核对表	65
规划 FlashCopy	68
FlashCopy 的硬件需求	68
FlashCopy 的软件需求	68
FlashCopy 的通信需求	69
用于高可用性的安全性规划	69
分发集群范围的信息	69
将集群与防火墙配合使用时的注意事项	69
在所有节点上维护用户概要文件	70
配置高可用性	70
场景: 配置高可用性	70
场景: 逻辑分区之间的切换式磁盘	71
场景: 在系统间使用切换式磁盘	72
场景: 切换式磁盘与地理镜像配合	73
场景: 跨站点镜像与地理镜像配合	75
场景: 跨站点镜像与城域镜像配合	76
场景: 跨站点镜像与全局镜像配合	78
对高可用性设置 TCP/IP	79
设置 TCP/IP 配置属性	80
启动 INETD 服务器	80
配置集群	80
创建集群	81
使节点能够被添加至集群	81
添加节点	82
启动节点	82
将节点添加至设备域	82
创建集群资源组 (CRG)	83
启动 CRG	86
指定消息队列	86
执行切换	88
配置节点	88
启动节点	88
使节点能够被添加至集群	89
添加节点	89
将节点添加至设备域	90
配置 CRG	90
启动 CRG	90
创建集群资源组 (CRG)	91
配置集群管理域	94
创建集群管理域	94
将节点添加至集群管理域	95
启动集群管理域	96
同步受监视资源	96
添加受监视资源项	97
配置切换式磁盘	97
创建独立磁盘池	98
启动镜像保护	98
停止镜像保护	99
添加磁盘机或磁盘池	99
评估当前配置	100
使磁盘池变为可用状态	101
配置跨站点镜像	102
配置地理镜像	102
配置城域镜像会话	103
配置全局镜像会话	103
管理高可用性	104
场景: 管理高可用性解决方案	104
场景: 在高可用性环境中执行备份	104
场景: 在地理镜像环境中执行备份	104
场景: 执行 FlashCopy	105

方案: 在高可用性环境中升级操作系统	105	管理城域镜像会话	144
示例: 升级操作系统	106	暂挂城域镜像会话	144
方案: 使设备具备高可用性	108	恢复城域镜像会话	145
管理集群	108	删除城域镜像会话	145
调整集群的集群版本	109	显示城域镜像属性	145
删除集群	110	管理全局镜像	146
显示集群配置	110	暂挂全局镜像会话	146
保存和恢复集群配置	110	恢复全局镜像会话	146
监视集群状态	111	删除全局镜像会话	146
指定消息队列	112	更改全局镜像会话属性	147
集群取消配置核对表	113	管理 FlashCopy	147
管理节点	114	配置 FlashCopy 会话	147
显示节点属性	114	更新 FlashCopy	147
停止节点	114	重新连接 FlashCopy	148
除去节点	114	拆离 FlashCopy	148
从设备域中除去节点	115	删除 FlashCopy	148
管理集群资源组 (CRG)	115	通过 FlashCopy 恢复数据	149
显示 CRG 状态	115	更改 FlashCopy 属性	149
停止 CRG	116	诊断高可用性解决方案	149
删除 CRG	117	集群故障诊断	149
创建可切换设备	117	确定是否存在集群问题	150
更改 CRG 的恢复域	118	收集集群的恢复信息	151
管理故障转移中断事件	118	常见集群问题	151
管理集群管理域	121	分区错误	153
停止集群管理域	122	确定主集群分区和辅助集群分区	154
删除集群管理域	122	将分区的节点更改为“已失败”	155
更改集群管理域的属性	123	分区集群管理域	155
管理受监视资源项	123	技巧: 集群分区	155
管理切换式磁盘	138	集群恢复	156
使磁盘池变为不可用状态	138	从集群作业故障恢复	156
使硬件可切换	138	恢复被破坏的集群对象	157
停顿独立磁盘池	140	在系统完全崩溃后恢复集群	157
恢复独立磁盘池	140	在发生灾难后恢复集群	158
管理跨站点镜像	140	从备份磁盘恢复集群	158
管理地理镜像	140	诊断跨站点镜像	158
暂挂地理镜像	140	地理镜像消息	158
恢复地理镜像	141	附录. 声明 161	
拆离镜像副本	142	编程接口信息	162
重新连接镜像副本	143	商标	162
取消配置地理镜像	143	条款和条件	163
更改地理镜像属性	144		

通过基于任务的方法实现高可用性

用于配置和管理 i5/OS® 高可用性的基于任务的方法允许您根据业务需要配置和管理定制高可用性解决方案。图形界面和命令行界面用于配置和管理高可用性解决方案。

基于解决方案的方法使用“高可用性解决方案管理器”图形界面，其中预定义解决方案是使用有限用户输入自动配置的，而基于任务的方法有所不同，它使得经验丰富的用户能够定制和实现个性化解决方案。但是，要使用此方法创建和管理高可用性解决方案，用户需要深入了解他们的高可用性需要，并且要熟悉一些界面。

“集群资源服务”图形界面

“集群资源服务”界面允许您配置和管理集群技术，它们是高可用性解决方案不可或缺的部分。要使用此界面，必须先安装 IBM® System i™ High Availability Solutions Manager (iHASM) 许可程序 5761-HAS。您可通过此界面执行下列功能：

- 创建和管理集群
- 创建和管理节点
- 配置和管理集群资源组
- 配置和管理集群管理域
- 创建和管理受监视资源
- 监视集群以了解与集群有关的事件，如集群分区和故障转移
- 在计划内中断（例如，安排好的系统维护）时执行手工切换

“磁盘管理”界面

“磁盘管理”界面允许您配置和管理独立磁盘池，它们是实现一些数据弹性技术所需的。根据实现的数据弹性技术的类型，使用其中一些功能时可能有一些安装需求：

- 创建磁盘池
- 使磁盘池变为可用状态
- 使磁盘池变为不可用状态
- 配置地理镜像
- 配置城域镜像
- 配置全局镜像

命令行界面

命令行界面允许您使用 CL 命令执行许多不同高可用性任务。对于每个与集群有关的任务，已标出相应的 CL 命令。

相关信息

IBM System i High Availability Solutions Manager (iHASM) 命令

规划高可用性解决方案

配置 i5/OS 高可用性解决方案之前，需要适当规划以确保满足解决方案的所有需求。

每个高可用性技术都有配置特定解决方案前应该满足的最低需求。除了这些需求之外，还需要确定应使哪些资源成为弹性资源。应评估这些资源（如应用程序、数据和设备）以确定它们是否应具备高可用性。如果它们需要高可用性，那么在为高可用性配置解决方案之前应对该环境进行所有必要的更改。例如，您可能有些数据驻留在 SYSBAS 中，它们应具备高可用性。配置解决方案之前，应将该数据移至独立磁盘池。应用程序可能还需要更改以启用高可用性。

规划应用程序弹性

应用程序弹性是高可用性环境中的其中一个关键元素。如果您计划编写并在集群中使用高可用性应用程序，那么应该了解这些应用程序具有特定的可用性规范。

通过利用环境中的弹性应用程序，可以在另一个集群节点上重新启动应用程序，而不必重新配置客户机。此外，与该应用程序相关联的数据将在切换或故障转移后可用。这意味着，当应用程序及其数据从主节点切换到备份节点时，应用程序用户只会遇到极为短暂的工作中断，甚至不会发生工作中断情况。用户不需要知道应用程序和数据已移至后端。

为了在集群中实现应用程序弹性，必须使用符合特定可用性规范的应用程序。应用程序必须具备某些特征才能成为可切换的应用程序，从而始终可供集群中的应用程序用户使用。由于存在这些要求，因此，在集群中使用可切换的应用程序时，可选择下列选项：

1. 购买支持集群的软件应用程序

支持集群的软件产品符合某些高可用性要求。

2. 编写或更改您自己的应用程序，使其具备高可用性

- 1 独立软件供应商和应用程序员可以对应用程序进行定制，使它们在 i5/OS 高可用性环境中可切换。

有了弹性应用程序后，必须在集群中对其进行管理。

相关信息

高可用性和集群

标出弹性应用程序

并非每个应用程序都能让您受益于集群的可用性。

应用程序必须有弹性，才能利用集群提供的切换和故障转移功能。应用程序弹性允许应用程序在备份节点上重新启动而不必重新配置使用该应用程序的客户机。因此，应用程序必须满足特定需求才能完全利用集群提供的功能。

支持集群的应用程序的 i5/OS 体系结构

高可用性应用程序为最终用户提供了附加值，此类应用程序是指发生计划内或计划外运行中断时保持可用状态的应用程序。

i5/OS 提供了应用程序弹性体系结构，此体系结构支持各种程度的高可用性应用程序。位于此范围高端的应用程序具备高可用性特征，并且能够自动实现高可用性环境，您通过高可用性管理界面管理这些应用程序。

这些应用程序具有下列特征：

- 这些应用程序可以在主节点变为不可用时切换到备份集群节点。
- 这些应用程序在“弹性定义”和“状态数据区”中定义了弹性环境，以允许集群管理应用程序自动配置和激活该应用程序。

- 这些应用程序通过用于处理集群相关事件的应用程序 CRG 出口程序提供应用程序弹性，从而利用 i5/OS 集群资源服务的功能。
- 这些应用程序提供了应用程序重新启动功能，用于将用户重新定位到应用程序菜单屏幕或之后的位置。

具有更严格的可用性和重新启动特征的应用程序具有下列特征：

- 应用程序 CRG 出口程序处理集群事件（操作码）的能力更为强大，改善了应用程序弹性。
- 应用程序提供了更高水平的应用程序重新启动支持。对于以主机为中心的应用程序而言，落实控制或检查点功能会将用户重新定位到事务边界。对于以客户机为中心的应用程序而言，用户将经历无缝的故障转移，并且最大程度地减少了服务中断情况。

编写高可用性集群应用程序

高可用性应用程序是对集群环境中的系统运行中断具有弹性的应用程序。

您可以实现多种水平的应用程序可用性：

1. 如果发生应用程序错误，那么该应用程序将在同一节点上重新启动自身并排除该错误的任何潜在原因，例如控制数据损坏。您可以将该应用程序视为仅仅是第一次启动。
2. 该应用程序执行一定程度的检查点重新启动处理。您可以将该应用程序视为接近故障点。
3. 如果发生系统运行中断，那么该应用程序将在备份服务器上重新启动。您可以将该应用程序视为仅仅是第一次启动。
4. 如果发生系统运行中断，那么该应用程序将在备份服务器上重新启动并在各个服务器上执行一定程度的检查点重新启动处理。您可以将该应用程序视为接近故障点。
5. 如果发生系统运行中断，那么对该应用程序及其相关联的数据进行协调式故障转移，使它们切换到集群中的其他一个或多个节点。您可以将该应用程序视为仅仅是第一次启动。
6. 如果发生系统运行中断，那么对该应用程序及其相关联的数据进行协调式故障转移，使它们切换到集群中的其他一个或多个节点。该应用程序在各个服务器上执行一定程度的检查点重新启动处理。您可以将该应用程序视为接近故障点。

注： 在上面的第 1 到第 4 种情况下，数据恢复工作由您负责。

使应用程序具有弹性：

了解如何使应用程序具有弹性。

弹性应用程序具有下列特征：

- 此应用程序可以在此节点或另一个节点上重新启动
- 客户机可以通过 IP 地址访问此应用程序
- 此应用程序没有状态，或者状态信息已知
- 与此应用程序相关联的数据在切换后可用

使应用程序具有弹性（即不受集群环境中系统运行中断的影响）的三个基本要素是：

应用程序本身

应用程序对错误或系统运行中断的容忍程度以及应用程序重新启动其自身时的透明度如何？

应用程序可以使用集群功能来处理这些问题。

相关联的数据

发生运行中断时，是否会影响任何相关联的数据的可用性？

可以将关键数据存储在切换式磁盘中，以使这些数据在系统运行中断期间保持可用状态。此外，利用集群功能的集群中间件 IBM 业务合作伙伴复制产品也可以处理此问题。

控制能力与管理

定义支持数据可用性和应用程序可用性的环境的容易程度如何？

IBM System i High Availability Solutions Manager (即 iHASM, 许可程序号为 5761-HAS) 提供了多个界面来配置和管理高可用性解决方案和技术。iHASM 许可程序提供了下列界面:

“高可用性解决方案管理器”图形界面

此图形界面允许您从 i5/OS 支持的多个高可用性解决方案中进行选择。此界面验证所选解决方案的所有技术需求, 配置所选解决方案及相关技术, 并简化了对该解决方案中所有高可用性技术的管理。

“集群资源服务”图形界面

此图形界面使有经验的用户能够更为灵活地定制高可用性解决方案。它允许您配置和管理集群技术, 例如 CRG。如果高可用性解决方案使用了某些独立磁盘池, 那么您还可以使用此界面对这些独立磁盘池进行配置。

IBM System i 高可用性解决方案管理器命令

这些命令提供了类似的功能, 但通过命令行界面运行。

IBM System i 高可用性解决方案管理器 (iHASM) API

这些 API 允许您使用独立磁盘池的新功能。

此外, 您还可以通过使用了集群 API 并且也将弹性应用程序与弹性数据相结合的第三方集群管理界面来完成工作。

相关信息

高可用性管理

重新启动高可用性集群应用程序:

要重新启动应用程序, 该应用程序必须知道它在故障转移或切换发生时所处的状态。

状态信息随应用程序的不同而有所变化; 因此, 应用程序必须确定所需的信息。即使没有任何状态信息, 也可以在 PC 上重新启动该应用程序。但是, 您必须恢复您在该应用程序中所处的位置。

可以通过多种方法为备份系统保存应用程序状态信息。每个应用程序都需要确定最适合于该应用程序的方法。

- 应用程序可以将所有状态信息传送到发出请求的客户机系统。发生切换或故障转移时, 该应用程序将使用客户机上存储的状态在新服务器中恢复状态。可以使用“分发信息”API 或“集群散列表”API 实现此功能。
- 应用程序可以通过实时方式复制状态信息, 例如作业信息以及其他与该应用程序相关联的控制结构。对于这些结构中的每一项更改, 该应用程序都将该更改发送至备份系统。
- 应用程序可以将与其相关联的持久状态信息存储在集群资源组中该应用程序的出口程序数据部分中。此方法假定只需要少量的状态信息。您可以使用“更改集群资源组”(QcstChangeClusterResourceGroup) API 来完成此任务。
- 应用程序可以将状态信息存储在数据对象中, 然后将该数据对象随该应用程序的数据一起复制至备份系统。
- 应用程序可以将状态信息存储在同时包含应用程序数据的可切换 IASP 中包含的数据对象中。
- 应用程序可以存储有关客户机的状态信息。
- 不保存任何状态信息, 在这种情况下, 您需要执行恢复。

注：如果应用程序使用了某种形式的检查点重新启动处理功能，那么可以减少所需保存的信息量。在这种情况下，只在预先确定的应用程序检查点保存状态信息。执行重新启动时，将回到上一个已知检查点，这与数据库的落实控制处理的工作方式类似。

调用集群资源组出口程序：

在集群环境的不同阶段，将调用集群资源组出口程序。

此程序建立在集群中实现资源弹性所必需的环境。此出口程序对于弹性设备 CRG 而言是可选的，但对于其他 CRG 类型而言是必需的。如果使用了集群资源组出口程序，那么发生集群范围事件时将调用该出口程序，这些事件包括：

- 节点意外地离开集群
- 节点作为“关闭集群节点”（QcstEndClusterNode）API 或“除去集群节点项”（QcstRemoveClusterNodeEntry）API 调用结果而离开集群
- 集群作为“删除集群”（QcstDeleteCluster）API 调用结果而被删除
- 节点作为“启动集群节点”（QcstStartClusterNode）API 调用结果而被激活
- 与分区节点的通信恢复

出口程序完成下列过程：

- 在指定的激活组或者调用者的激活组（*CALLER）中运行。
- 如果出口程序发生了未处理的异常或者被取消，那么将忽略重新启动参数。
- 提供取消处理程序。

当集群资源组 API 运行时，将使用“创建集群资源组”（QcstCreateClusterResourceGroup）API 指定的用户概要文件从独立作业中调用出口程序。这个独立作业由该 API 在出口程序被调用时自动创建。如果数据 CRG 的出口程序未成功完成或者异常关闭，那么将使用操作码“撤销”在恢复域中所有活动节点上调用集群资源组出口程序。此操作码允许撤销任何未完成的活动并恢复集群资源组的原始状态。

假定设备 CRG 发生了不成功的切换。切换回所有设备后，如果所有设备都曾经在原始主节点上成功地联机，那么集群功能将使用操作码“启动”在原始主节点上调用出口程序。

如果应用程序 CRG 的出口程序未成功完成或者异常关闭，并且该 CRG 处于活动状态，那么集群资源服务将尝试重新启动该应用程序。将使用操作码“重新启动”来调用集群资源组出口程序。如果无法在指定的最大尝试次数内重新启动该应用程序，那么将使用操作码“故障转移”来调用集群资源组出口程序。仅当使用操作码“启动”来调用出口程序时，才会将重新启动计数复位，这可能是启动 CRG、故障转移或切换的结果。

当集群资源组启动时，在主节点上调用的应用程序 CRG 出口程序直到该应用程序本身关闭或出错后，才会将控制返回给集群资源服务。应用程序 CRG 进入活动状态后，如果集群资源服务必须将某些事件通知应用程序 CRG 出口程序，那么将在另一个作业中启动该出口程序的另一个实例。可能会返回除“启动”或“重新启动”以外的任何操作码。

调用集群资源组出口程序时，会将一组参数传递给该程序，这组参数标识正在处理的集群事件、集群资源的当前状态以及集群资源的预期状态。

有关集群资源组出口程序的完整资料，包括使用每个操作码时要传递给出口程序的信息，请参阅集群 API 文档中的集群资源组出口程序。在 QUSRTOOL 库中提供了样本源代码，可作为您编写出口程序的基础。请参阅 QATTSYSC 文件中的 TCSTAPPEXT 成员。

应用程序 CRG 注意事项

应用程序集群资源组用于管理应用程序弹性。

管理应用程序 CRG 对 IP 地址的接管:

可以使用集群资源服务来管理应用程序 CRG 对 IP 地址的接管。也可以进行手工管理。

可以通过两种方法管理与应用程序 CRG 相关联的应用程序接管 IP 地址。最简单的方法（即缺省方法）是让集群资源服务管理接管 IP 地址。此方法指示集群资源服务在恢复域中的所有节点上创建接管 IP 地址，这包括以后添加到恢复域中的节点。如果选择此方法，那么该接管 IP 地址在恢复域中的任何节点上当前都不能处于已定义状态。

另一种方法是，您自己管理接管 IP 地址。此方法指示集群资源服务不要执行任何用于配置接管 IP 地址的步骤，即配置工作由用户负责。在这种情况下，必须先要在恢复域中除复制节点以外的所有节点上添加接管 IP 地址，然后才能启动集群资源组。对于任何要添加至活动 CRG 的恢复域的节点而言，在添加前，必须配置接管 IP 地址。

相关概念

第 7 页的『示例: 应用程序集群资源组故障转移操作』

本示例说明一个故障转移场景的工作方式。其他故障转移场景的工作方式可能有所不同。

多个子网: 尽管缺省情况下所有恢复域节点在同一个子网中，但可以让应用程序接管 IP 地址跨多个子网工作。在恢复域中的节点分布在多个子网中的情况下，要配置应用程序接管 IP 地址，需要启用切换环境。

跨子网启用应用程序切换:

通常，集群功能要求应用程序集群资源组的恢复域中所有集群节点驻留在同一个 LAN 中，即使用相同的子网寻址方法。集群资源服务支持用户在配置应用程序 CRG 时配置接管 IP 地址。

用于将所配置应用程序接管 IP 地址从恢复域中一个节点切换到另一节点的底层网络协议是地址解析协议（ARP）。但是，可以对恢复域进行扩展，使其包含驻留在其他由商用路由器分隔的 LAN 中的集群节点。通过使用虚拟 IP 地址支持并在集群节点上使用路由信息协议（RIP）以及在网络中使用商用路由器，可以实现此扩展。

要启用切换环境，需要执行下列手工配置步骤。必须在恢复域中所有节点上按照这组指示信息执行操作，并且必须在集群中其他将要成为给定应用程序 CRG 的恢复域节点的节点上重复这些操作。

1. 选择应用程序 CRG 要使用的接管 IP 地址。

- 为了避免发生冲突，此地址不应与集群节点或路由器使用的任何其他现有地址重叠。例如，如果选择 19.19.19.19，那么请确保 19.0.0.0（19.19.0.0）不是系统路由表已知的路径。
- 添加接管接口，例如 19.19.19.19。创建此接口时，请指定线路描述 *VIRTUALIP，子网掩码 255.255.255.255（主机路由），最大传输单元 1500（范围 576-16388 内的任何数值）以及自动启动参数 *NO。此接管地址（例如 19.19.19.19）必须已作为 *VIRTUALIP 地址存在，这样才能在下一步中将其标识为“相关联的本地接口”。但是，此地址不必处于活动状态。

2. 使预期接管 IP 地址与创建集群或将节点添加至集群时用于集群通信的两个 IP 地址中的一个或全部相关联。

- 例如，这意味着使 19.19.19.19 接管地址成为以太网总线中用于本地集群功能的集群节点 IP 地址的“相关联的本地接口”。必须对每个集群节点上的每个集群地址执行此操作。

注: 要使用 CFGTCP 进行此更改，必须关闭那些集群地址。

3. 创建集群以及任何 CRG。对于应用程序 CRG，请对“配置接管 IP 地址”字段指定 QcstUserCfgsTakeoverIpAddr。请不要启动任何应用程序 CRG。

4. 通过依次使用 CFGTCP 中的“配置 TCP/IP 应用程序”（选项 20）、“配置 RouteD”（选项 2）和“更改 RouteD 属性”（选项 1），确保“提供”设置为 *YES。否则，将其设置为 *YES 并在每个集群节点上启动或重新启动 ROUTED（RIP 或 RIP-2）。
 - 如果当前正在运行使用本地端口的 ROUTED，那么 NETSTAT 选项 3 将显示该 ROUTED。ROUTED 必须处于运行状态并在 CRG 恢复域中每个集群节点上发布路径（提供 = *YES）。
5. 确保网络中所有用于对恢复域 LAN 进行互连的商用路由器都接受并发布 RIP 的主机路径。
 - 这不一定是路由器的缺省设置。语言随路由器制造商的不同而有所变化，但应该设置“RIP 接口”。
 - 这也适用于指向系统的路由器接口以及路由器之间的接口。

注：在此配置中，请不要使用 System i 机器作为路由器。请使用 IBM 或其他公司推出的用于进行路由的商用路由器。无法将 System i 路由功能配置为处理此功能。
6. 在其中一个集群节点上手工激活接管地址：
 - a. 等待 5 分钟，允许 RIP 传播路径。
 - b. 从 CRG 恢复域中的所有节点以及从 LAN 中将要使用此地址的所选客户机 Ping 接管地址。
 - c. 再次确保接管地址已关闭。

（集群将在 CRG 启动时在所指定主节点上启动此地址。）
7. 启动应用程序 CRG。
 - 集群将在指定的首选节点上启动接管地址，RIP 将在整个恢复域中发布路径。RIP 可能需要 5 分钟时间来在整个域中更新路径。RIP 功能与“启动 CRG”功能无关。

要点:

- 如果未对应用程序 CRG 恢复域中的所有集群节点执行以上过程，那么该集群在切换过程中将挂起。
- 即使未执行故障转移以切换到副本节点，也最好在副本节点上执行此过程，以防日后将其更改为备份节点。
- 如果要使用多个虚拟 IP 地址，那么每个虚拟 IP 地址都要求与独立的应用程序 CRG 和独立的 IP 地址相关联。此地址可以是同一物理适配器上的另一个逻辑 IP 地址，也可以是另一个物理适配器。并且，您必须慎防路由表包含不明确的内容。可以通过执行下列操作来实现此目标：
 - 对于每个虚拟 IP 地址，在路由表中添加 *DFTRROUTE。
 - 要使用多个 IP 地址，请使用 CFGTCP（选项 2）。
 - 设置访问所选路由器所选的所有参数，其中包括下一跃；但是，应该将“首选绑定接口”设置为与此路由所代表的虚拟 IP 地址相关联的本地系统 IP 地址。

示例: 应用程序集群资源组故障转移操作:

本示例说明一个故障转移场景的工作方式。其他故障转移场景的工作方式可能有所不同。

当弹性应用程序的集群资源组由于超出重试限制或作业被取消而进行故障转移时，将发生下列情况:

- 使用操作码“故障转移”在 CRG 恢复域中的所有活动节点上调用集群资源组出口程序。这表明集群资源服务正在准备对该应用程序进行故障转移，以将它的访问点切换到第一个备份节点。
- 集群资源服务关闭主节点上的接管因特网协议（IP）连接。
- 集群资源服务在第一个备份节点（新的主节点）上启动接管 IP 地址。
- 集群资源服务提交一个作业，该作业使用操作码“启动”仅在新的主节点上调用集群资源组出口程序。此操作将重新启动该应用程序。

相关概念

第 6 页的『管理应用程序 CRG 对 IP 地址的接管』

可以使用集群资源服务来管理应用程序 CRG 对 IP 地址的接管。也可以进行手工管理。

示例: 应用程序出口程序:

此代码示例包含应用程序集群资源组出口程序。

您可以在 QUSRTOOL 库中找到此代码示例。

注: 只要使用这些代码示例, 即表示您同意第 159 页的『代码许可证和免责声明』中的条款。

```
/*-----*/
/*
/* Library:   QUSRTOOL
/* File:     QATTSYSC
/* Member:   TCSTAPPEXT
/* Type:     ILE C
/*
/* Description:
/* This is an example application CRG exit program which gets called for
/* various cluster events or cluster APIs. The bulk of the logic must
/* still be added because that logic is really dependent upon the unique
/* things that need to be done for a particular application.
/*
/* The intent of this example to to provide a shell which contains the
/* basics for building a CRG exit program. Comments throughout the example
/* highlight the kinds of issues that need to be addressed by the real
/* exit program implementation.
/*
/* Every action code that applies to an application CRG is handled in this
/* example.
/*
/* The tcstdtaara.h include is also shipped in the QUSRTOOL library. See
/* the TCSTDTAARA member in the QATTSYSC file.
/*
/* Change log:
/* Flag Reason Ver Date User Id Description
/* -----
/* ... D98332 v5r1m0 000509 ROCH Initial creation.
/* $A1 P9950070 v5r2m0 010710 ROCH Dataarea fixes
/* $A2 D99055 v5r2m0 010913 ROCH Added CancelFailover action code
/* $A3 D98854 v5r2m0 010913 ROCH Added VerificationPhase action code
/* $A4 P9A10488 v5r3m0 020524 ROCH Added example code to wait for data
/* CRGs on switchover action code
/*
/*-----*/

/*-----*/
/*
/* Header files
/*
/*-----*/
#include /* Useful when debugging */
#include /* offsetof macro */
#include /* system function */
#include /* String functions */
#include /* Exception handling constants/structures */
#include /* Various cluster constants */
#include /* Structure of CRG information */
#include "qusrtool/qattsysc/tcstdtaara" /* QCSTHAAPPI/QCSTHAAPPO data areas*/
#include /* API to Retrieve contents of a data area */
#include /* API error code type definition */
#include /* mitime builtin */
#include /* waittime builtin */
```

```

/*-----*/
/*
/* Constants
/*
/*-----*/
#define UnknownRole -999
#define DependCrgDataArea "QCSTHAAPPO"
#define ApplCrgDataArea "QCSTHAAPPI"
#define Nulls 0x00000000000000000000

/*-----*/
/*
/* The following constants are used in the checkDependCrgDataArea()
/* function. The first defines how long to sleep before checking the data
/* area. The second defines that maximum time to wait for the data area
/* to become ready before failing to start the application when the Start
/* CRG function is being run. The third defines the maximum wait time for
/* the Initiate Switchover or failover functions.
/*
/*-----*/
#define WaitSecondsIncrement 30
#define MaxStartCrgWaitSeconds 0
#define MaxWaitSeconds 900

/*-----*/
/*
/* As this exit program is updated to handle new action codes, change the
/* define below to the value of the highest numbered action code that is
/* handled.
/*
/*-----*/
#define MaxAc 21

/*-----*/
/*
/* If the exit program data in the CRG has a particular structure to it,
/* include the header file for that structure definition and change the
/* define below to use that structure name rather than char.
/*
/*-----*/
#define EpData char

/*-----*/
/*
/* Change the following define to the library the application resides in
/* and thus where the QCSTHAAPPO and QCSTHAAPPI data areas will be found.
/*
/*-----*/
#define ApplLib "QGPL"

/*-----*/
/*
/* Prototypes for internal functions.
/*
/*-----*/
static int getMyRole(Qcst_EXTP0100_t *, int, int);
#pragma argopt(getMyRole)
static int doAction(int, int, int, Qcst_EXTP0100_t *, EpData *);
#pragma argopt(doAction)
static int createCrg(int, int, Qcst_EXTP0100_t *, EpData *);
static int startCrg(int, int, Qcst_EXTP0100_t *, EpData *);
static int restartCrg(int, int, Qcst_EXTP0100_t *, EpData *);
static int endCrg(int, int, Qcst_EXTP0100_t *, EpData *);
static int verifyPhase(int, int, Qcst_EXTP0100_t *, EpData *);

```

```

static int deleteCrg(int, int, Qcst_EXTP0100_t *, EpData *);
static int memberIsJoining(int, int, Qcst_EXTP0100_t *, EpData *);
static int memberIsLeaving(int, int, Qcst_EXTP0100_t *, EpData *);
static int switchPrimary(int, int, Qcst_EXTP0100_t *, EpData *);
static int addNode(int, int, Qcst_EXTP0100_t *, EpData *);
static int rmvNode(int, int, Qcst_EXTP0100_t *, EpData *);
static int chgCrg(int, int, Qcst_EXTP0100_t *, EpData *);
static int deleteCrgWithCmd(int, int, Qcst_EXTP0100_t *, EpData *);
static int undoPriorAction(int, int, Qcst_EXTP0100_t *, EpData *);
static int endNode(int, int, Qcst_EXTP0100_t *, EpData *);
static int chgNodeStatus(int, int, Qcst_EXTP0100_t *, EpData *);
static int cancelFailover(int, int, Qcst_EXTP0100_t *, EpData *);
static int newActionCode(int, int, Qcst_EXTP0100_t *, EpData *);
static int undoCreateCrg(int, int, Qcst_EXTP0100_t *, EpData *);
static int undoStartCrg(int, int, Qcst_EXTP0100_t *, EpData *);
static int undoEndCrg(int, int, Qcst_EXTP0100_t *, EpData *);
static int undoMemberIsJoining(int, int, Qcst_EXTP0100_t *, EpData *);
static int undoMemberIsLeaving(int, int, Qcst_EXTP0100_t *, EpData *);
static int undoSwitchPrimary(int, int, Qcst_EXTP0100_t *, EpData *);
static int undoAddNode(int, int, Qcst_EXTP0100_t *, EpData *);
static int undoRmvNode(int, int, Qcst_EXTP0100_t *, EpData *);
static int undoChgCrg(int, int, Qcst_EXTP0100_t *, EpData *);
static int undoCancelFailover(int, int, Qcst_EXTP0100_t *, EpData *);
static void bldDataAreaName(char *, char *, char *);
#pragma argopt(bldDataAreaName)
static int checkDependCrgDataArea(unsigned int);
#pragma argopt(checkDependCrgDataArea)
static void setApp1CrgDataArea(char *);
#pragma argopt(setApp1CrgDataArea)
static void cancelHandler(_CNL_Hndlr_Parms_T *);
static void unexpectedExceptionHandler(_INTRPT_Hndlr_Parms_T *);
static void endApplication(unsigned int, int, int, Qcst_EXTP0100_t *, EpData *);
#pragma argopt(endApplication)

/*-----*/
/*
/* Some debug routines
/*
/*-----*/
static void printParms(int, int, int, Qcst_EXTP0100_t *, EpData *);
static void printActionCode(unsigned int);
static void printCrgStatus(int);
static void printRcvyDomain(char *,
                           unsigned int,
                           Qcst_Rcvy_Domain_Array1_t *);
static void printStr(char *, char *, unsigned int);

/*-----*/
/*
/* Type definitions
/*
/*-----*/

/*-----*/
/*
/* This structure defines data that will be passed to the exception and
/* cancel handlers. Extend it with information unique to your application.*/
/*
/*-----*/
typedef struct {
    int *retCode;           /* Pointer to return code
    EpData *epData;        /* Exit program data from the CRG
    Qcst_EXTP0100_t *crgData; /* CRG data
    unsigned int actionCode; /* The action code
    int role;              /* This node's recovery domain role
    int priorRole;         /* This node's prior recovery domainrole

```



```

} volatile HandlerDataT;

/*-----*/
/*
/* Function pointer array for handling action codes. When the exit program*/
/* is updated to handle new action codes, add the new function names to */
/* this function pointer array. */
/*
/*-----*/
static int (*fcn[MaxAc+1]) (int role,
                           int priorRole,
                           Qcst_EXTP0100_t *crgData,
                           EpData *epData) = {
    newActionCode, /* 0 - currently reserved */
    createCrg,    /* 1 */
    startCrg,    /* 2 */
    restartCrg,  /* 3 */
    endCrg,      /* 4 */
    verifyPhase, /* 5 - currently reserved */
    newActionCode, /* 6 - currently reserved */
    deleteCrg,   /* 7 */
    memberIsJoining, /* 8 */
    memberIsLeaving, /* 9 */
    switchPrimary, /* 10 */
    addNode,     /* 11 */
    rmvNode,    /* 12 */
    chgCrg,     /* 13 */
    deleteCrgWithCmd, /* 14 */
    undoPriorAction, /* 15 */
    endNode,    /* 16 */
    newActionCode, /* 17 - applies only to a device CRG */
    newActionCode, /* 18 - applies only to a device CRG */
    newActionCode, /* 19 - applies only to a device CRG */
    chgNodeStatus, /* 20 */
    cancelFailover /* 21 */
};

/*-----*/
/*
/* Function pointer array for handling prior action codes when called with */
/* the Undo action code. When the exit program is updated to handle */
/* Undo for new action codes, add the new function names to this function */
/* pointer array. */
/*
/*-----*/
static int (*undoFcn[MaxAc+1]) (int role,
                                int priorRole,
                                Qcst_EXTP0100_t *crgData,
                                EpData *epData) = {
    newActionCode, /* 0 - currently reserved */
    undoCreateCrg, /* 1 */
    undoStartCrg, /* 2 */
    newActionCode, /* 3 */
    undoEndCrg, /* 4 */
    newActionCode, /* 5 - no undo for this action code */
    newActionCode, /* 6 - currently reserved */
    newActionCode, /* 7 */
    undoMemberIsJoining, /* 8 */
    undoMemberIsLeaving, /* 9 */
    undoSwitchPrimary, /* 10 */
    undoAddNode, /* 11 */
    undoRmvNode, /* 12 */
    undoChgCrg, /* 13 */
    newActionCode, /* 14 */
    newActionCode, /* 15 */
};

```

```

newActionCode,      /* 16 */
newActionCode,      /* 17 - applies only to a device CRG */
newActionCode,      /* 18 - applies only to a device CRG */
newActionCode,      /* 19 - applies only to a device CRG */
newActionCode,      /* 20 */
undoCancelFailover /* 21 */
};

/*****
*/
/* This is the entry point for the exit program. */
/*
*/
/*****
void main(int argc, char *argv[]) {

    HandlerDataT hdldata;

/*****
*/
/*-----*/
/*
*/
/* Take each of the arguments passed in the argv array and cast it to
*/
/* the correct data type.
*/
/*
*/
/*****
*/
/*-----*/
int *retCode      = (int *)argv[1];
unsigned int *actionCode = (unsigned int *)argv[2];
EpData *epData    = (EpData *)argv[3];
Qcst_EXTP0100_t *crgData = (Qcst_EXTP0100_t *)argv[4];
char *formatName  = (char *)argv[5];

/*****
*/
/*-----*/
/*
*/
/* Ensure the format of the data being passed is correct. */
/* If not, a change has been made and this exit program needs to be
*/
/* updated to accommodate the change. Add appropriate error logging for
*/
/* your application design.
*/
/*
*/
/*****
*/
/*-----*/
if (0 != memcmp(formatName, "EXTP0100", 8))
    abort();

/*****
*/
/*-----*/
/*
*/
/* Set up the data that will be passed to the exception and cancel
*/
/* handlers.
*/
/*
*/
/*****
*/
/*-----*/
hdldata.retCode      = retCode;
hdldata.epData       = epData;
hdldata.crgData      = crgData;
hdldata.actionCode   = *actionCode;
hdldata.role         = UnknownRole;
hdldata.priorRole    = UnknownRole;
_VBDY(); /* force changed variables to home storage location */
/*****
*/
/*-----*/
/*
*/
/* Enable an exception handler for any and all exceptions.
*/
/*
*/
/*****
*/

```

```

/*-----*/
#pragma exception_handler(unexpectedExceptionHandler, hdlData, \
                        _C1_ALL, _C2_ALL, _CTLA_INVOKE )

/*-----*/
/*
/* Enable a cancel handler to recover if this job is canceled.
/*
/*
/*-----*/
#pragma cancel_handler(cancelHandler, hdlData)

/*-----*/
/*
/* Extract the role and prior role of the node this exit program is
/* running on. If the cluster API or event changes the recovery domain
/* (node role or membership status), the new recovery domain's offset is
/* passed in Offset_Rcvy_Domain_Array and the offset of the recovery
/* domain as it looked prior to the API or cluster event is passed in
/* Offset_Prior_Rcvy_Domain_Array. If the recovery domain isn't changed,
/* only Offset_Rcvy_Domain_Array can be used to address the recovery
/* domain.
/*
/*
/*-----*/
hdlData.role = getMyRole(crgData,
                        crgData->Offset_Rcvy_Domain_Array,
                        crgData->Number_Nodes_Rcvy_Domain);
if (crgData->Offset_Prior_Rcvy_Domain_Array)
    hdlData.priorRole =
        getMyRole(crgData,
                  crgData->Offset_Prior_Rcvy_Domain_Array,
                  crgData->Number_Nodes_Prior_Rcvy_Domain);
else
    hdlData.priorRole = hdlData.role;
_VBDY(); /* force changed variables to home storage location */

/*-----*/
/*
/* Enable the following to print out debug information.
/*
/*
/*-----*/
/*
    printParms(*actionCode, hdlData.role, hdlData.priorRole, crgData,
epData);
*/

/*-----*/
/*
/* Do the correct thing based upon the action code. The return code
/* is set to the function result of doAction().
/*
/*
/*-----*/
*retCode = doAction(*actionCode,
                    hdlData.role,
                    hdlData.priorRole,
                    crgData,
                    epData);

```

```

/*-----*/
/*
/* The exit program job will end when control returns to the operating
/* system at this point.
/*
/*
/*-----*/
return;

#pragma disable_handler /* unexpectedExceptionHandler */
#pragma disable_handler /* cancelHandler */
} /* end main() */

/*****/
/*
/* Get the role of this particular node from one of the views of the
/* recovery domain.
/*
/*
/* APIs and cluster events which pass the updated and prior recovery domain*/
/* to the exit program are:
/*
/* QcstAddNodeToRcvyDomain
/* QcstChangeClusterNodeEntry
/* QcstChangeClusterResourceGroup
/* QcstEndClusterNode (ending node does not get the prior domain)
/* QcstInitiateSwitchOver
/* QcstRemoveClusterNodeEntry (removed node does not get the prior domain)
/* QcstRemoveNodeFromRcvyDomain
/* QcstStartClusterResourceGroup (only if inactive backup nodes are
/* reordered)
/*
/* a failure causing failover
/* a node rejoining the cluster
/* cluster partitions merging
/*
/*
/* All other APIs pass only the updated recovery domain.
/*
/*
/*****/
static int getMyRole(Qcst_EXT_P0100_t *crgData, int offset, int
count) {

    Qcst_Rcvy_Domain_Array1_t *nodeData;
    unsigned int iter = 0;

/*-----*/
/*
/* Under some circumstances, the operating system may not be able to
/* determine the ID of this node and passes *NONE. An example of such a
/* circumstance is when cluster resource services is not active on a
/* node and the DLTCRG CL command is used.
/*
/*
/*-----*/
    if (0 == memcmp(crgData->This_Nodes_ID, QcstNone,
sizeof(Qcst_Node_Id_t)))
        return UnknownRole;

/*-----*/
/*
/* Compute a pointer to the first element of the recovery domain array.
/*
/*
/*-----*/
    nodeData = (Qcst_Rcvy_Domain_Array1_t *)((char *)crgData +

```

```

offset);

/*-----*/
/*
/* Find my node in the recovery domain array. I will not be in the
/* prior recovery domain if I am being added by the Add Node to Recovery
/* Domain API.
/*
/*
/*-----*/
while ( 0 != memcmp(crgData->This_Nodes_ID,
                  nodeData->Node_ID,
                  sizeof(Qcst_Node_Id_t))
        &&
        iter < count
    ) {
    nodeData++;
    iter++;
}

if (iter < count)
    return nodeData->Node_Role;
else
    return UnknownRole;
} /* end getMyRole() */

/*****
/*
/* Call the correct function based upon the cluster action code. The
/* doAction() function was split out from main() in order to clarify the
/* example. See the function prologues for each called function for
/* information about a particular cluster action.
/*
/*
/* Each action code is split out into a separate function only to help
/* clarify this example. For a particular exit program, some action codes
/* may perform the same function in which case multiple action codes could
/* be handled by the same function.
/*
/*
*****/
static int doAction(int actionCode,
                  int role,
                  int priorRole,
                  Qcst_EXTP0100_t *crgData,
                  EpData *epData) {

/*-----*/
/*
/* For action codes this exit program knows about, call a function to
/* do the work for that action code.
/*
/*
/*-----*/

if (actionCode <= MaxAc )
    return (*fcn[actionCode]) (role, priorRole, crgData, epData);
else

/*-----*/
/*
/* IBM has defined a new action code in a new operating system release
/* and this exit program has not yet been updated to handle it. Take a
/* default action for now.
/*
/*

```

```

/*-----*/
    return newActionCode(role, priorRole, crgData, epData);
} /* end doAction() */

/*****/
/* */
/* Action code = QcstCrgAcInitialize */
/* */
/* The QcstCreateClusterResourceGroup API was called. A new cluster */
/* resource group object is being created. */
/* */
/* Things to consider: */
/* - Check that the application program and all associated objects are on */
/* the primary and backup nodes. If the objects are not there, */
/* consider sending error/warning messages or return a failure return */
/* code. */
/* - Check that required data or device CRGs are on all nodes in the */
/* recovery domain. */
/* - Perform any necessary setup that is required to run the */
/* the application on the primary or backup nodes. */
/* - If this CRG is enabled to use the QcstDistributeInformation API, */
/* the user queue needed by that API could be created at this time. */
/* */
/*****/
static int createCrg(int role,
                    int doesNotApply,
                    Qcst_EXTP0100_t *crgData,
                    EpData *epData) {

    return QcstSuccessful;
} /* end createCrg() */

/*****/
/* */
/* Action code = QcstCrgAcStart */
/* */
/* The QcstStartClusterResourceGroup API was called. A cluster resource */
/* group is being started. */
/* The QcstInitiateSwitchOver API was called and this is the second action */
/* code being passed to the exit program. */
/* The fail over event occurred and this is the second action code being */
/* passed to the exit program. */
/* */
/* A maximum wait time is used when checking to see if all dependent CRGs */
/* are active. This is a short time if the CRG is being started because of */
/* the QcstStartClusterResourceGroup API. It is a longer time if it is */
/* because of a failover or switchover. When failover or switchover are */
/* being done, it make take a while for data or device CRGs to become */
/* ready so the wait time is long. If the Start CRG API is being used, the */
/* dependent CRGs should already be started or some error occurred, the */
/* CRGs were started out of order, etc. and there is no need for a long */
/* wait. */
/* */
/* Things to consider: */
/* - If this node's role is primary, the application should be started. */
/* This exit program should either call the application so that it runs */
/* in this same job or it should monitor any job started by this */
/* exit program so the exit program knows when the application job */
/* ends. By far, the simplest approach is run the application in this */
/* job by calling it. */
/* Cluster Resource Services is not expecting this exit program to */
/* return until the application finishes running. */
/* - If necessary, start any associated subsystems, server jobs, etc. */
/* - Ensure that required data CRGs have a status of active on all nodes */
/* in the recovery domain. */

```

```

/*                                                                 */
/*****                                                             */
static int startCrg(int role,
                    int doesNotApply,
                    Qcst_EXTP0100_t *crgData,
                    EpData *epData) {

    unsigned int maxWaitTime;

    /* Start the application if this node is the primary          */
    if (role == QcstPrimaryNodeRole) {

/*-----*/
        /* Determine if all CRGs that this application CRG is dependent upon
        /* are ready. If the check fails, return from the Start action code.
        /* Cluster Resource Services will change the state of the CRG to
        /* Inactive.
        /*
        /*
/*-----*/
        if (crgData->Cluster_Resource_Group_Status ==
QcstCrgStartCrgPending)
            maxWaitTime = MaxStartCrgWaitSeconds;
        else
            maxWaitTime = MaxWaitSeconds;
        if (QcstSuccessful != checkDependCrgDataArea(maxWaitTime))
            return QcstSuccessful;

/*-----*/
        /* Just before starting the application, update the data area to
        /* indicate the application is running.
        /*
        /*
/*-----*/
        setApp1CrgDataArea(App1_Running);

/*-----*/
        /* Add logic to call application here. It is expected that control
        /* will not return until something causes the application to end: a
        /* normal return from the exit program, the job is canceled, or an
        /* unhandled exception occurs. See the cancelHandler() function for
        /* some common ways this job could be canceled.
        /*
        /*
/*-----*/

/*-----*/
        /* After the application has ended normally, update the data area to
        /* indicate the application is no longer running.
        /*
        /*
/*-----*/
        setApp1CrgDataArea(App1_Ended);
    }
    else

/*-----*/

```

```

    /*
    /* On backup or replicate nodes, mark the status of the application in */
    /* the data area as not running. */
    /*
    /*-----*/
    setApp1CrgDataArea(App1_Ended);

    return QcstSuccessful;
} /* end startCrg()
    */

/*****/
/*
/* Action code = QcstCrgAcRestart */
/*
/* The previous call of the exit program failed and set the return */
/* code to QcstFailWithRestart or it failed due to an exception and the */
/* exception was allowed to percolate up the call stack. In either */
/* case, the maximum number of times for restarting the exit program has */
/* not been reached yet. */
/*
/* This action code is passed only to application CRG exit programs which */
/* had been called with the Start action code. */
/*
/*****/
static int restartCrg(int role,
                    int doesNotApply,
                    Qcst_EXTP0100_t *crgData,
                    EpData *epData) {

/*-----*/
/*
/* Perform any unique logic that may be necessary when restarting the */
/* application after a failure and then call the startCrg() function to */
/* do the start functions. */
/*
/*-----*/

    return startCrg(role, doesNotApply, crgData, epData);
} /* end restartCrg()

/*****/
/*
/* Action code = QcstCrgAcEnd */
/*
/* The end action code is used for one of the following reasons: */
/* - The QcstEndClusterResourceGroup API was called. */
/* - The cluster has become partitioned and this node is in the secondary */
/* partition. The End action code is used regardless of whether the */
/* CRG was active or inactive. Action code dependent data of */
/* QcstPartitionFailure will also be passed. */
/* - The application ended. Action code dependent data of */
/* QcstResourceEnd will also be passed. All nodes in the recovery */
/* domain will see the same action code (including the primary). */
/* - The CRG job has been canceled. The exit program on this node will */
/* be called with the End action code. QcstMemberFailure will be */
/* passed as action code dependent data. */
/*
/*
/*
/*
/*****/

```



```

/* Things to consider: */
/* - If the CRG is active, the job running the application is canceled */
/* and the IP takeover address is ended AFTER the exit program is */
/* called. */
/* - If subsystems or server jobs were started as a result of the */
/* QcstCrgAcStart action code, end them here or consolidate all logic */
/* to end the application in the cancelHandler() since it will be */
/* invoked for all Cluster Resource Services APIs which must end the */
/* application on the current primary. */
/* */
/*****/
static int endCrg(int role,
                 int priorRole,
                 Qcst_EXTP0100_t *crgData,
                 EpData *epData) {

/*-----*/
/*
/* End the application if it is running on this node.
/*
/*
/*-----*/
    endApplication(QcstCrgAcRemoveNode, role, priorRole, crgData,
epData);

    return QcstSuccessful;
} /* end endCrg() */

/*****/
/*
/* Action code = QcstCrgAcVerificationPhase
/*
/*
/* The verification phase action code is used to allow the exit program to
/* do some verification before proceeding with the requested function
/* identified by the action code depended data. If the exit program
/* determines that the requested function cannot proceed it should return
/* QcstFailWithOutRestart.
/*
/*
/*
/* NOTE: The exit program will NOT be called with Undo action code.
/*
/*
/*****/
static int verifyPhase(int role,
                     int doesNotApply,
                     Qcst_EXTP0100_t *crgData,
                     EpData *epData) {

/*-----*/
/*
/* Do verification
/*
/*
/*-----*/
    if (crgData->Action_Code_Dependent_Data == QcstDltCrg) {
        /* do verification */
        /* if ( fail ) */
        /* return QcstFailWithOutRestart */
    }

    return QcstSuccessful;
} /* end verifyPhase() */

/*****/

```

```

/* */
/* Action code = QcstCrgAcDelete */
/* */
/* The QcstDeleteClusterResourceGroup or QcstDeleteCluster API was called. */
/* A cluster resource group is being deleted while Cluster Resource */
/* Services is active. */
/* If the QcstDeleteCluster API was used, action code dependent data of */
/* QcstDltCluster is passed. */
/* If the QcstDeleteCluster API was used and the CRG is active, the exit */
/* program job which is still active for the Start action code is canceled*/
/* after the Delete action code is processed. */
/* */
/* Things to consider: */
/* - Delete application programs and objects from nodes where they are */
/* no longer needed such as backup nodes. Care needs to be exercised */
/* when deleting application objects just because a CRG is being */
/* deleted since a particular scenario may want to leave the */
/* application objects on all nodes. */
/* */
/*****/
static int deleteCrg(int role,
                    int doesNotApply,
                    Qcst_EXTP0100_t *crgData,
                    EpData *epData) {

    return QcstSuccessful;
} /* end deleteCrg()
    */

/*****/
/* */
/* Action code = QcstCrgAcReJoin */
/* */
/* One of three things is occurring- */
/* 1. The problem which caused the cluster to become partitioned has been */
/* corrected and the 2 partitions are merging back together to become */
/* a single cluster. Action code dependent data of QcstMerge will be */
/* passed. */
/* 2. A node which either previously failed or which was ended has had */
/* cluster resource services started again and the node is joining the */
/* cluster. Action code dependent data of QcstJoin will be passed. */
/* 3. The CRG job on a particular node which may have been canceled or */
/* ended has been restarted. Action code dependent data of QcstJoin */
/* will be passed. */
/* */
/* Things to consider: */
/* - If the application replicates application state information to other*/
/* nodes when the application is running, this state information will */
/* need to be resynchronized with the joining nodes if the CRG is */
/* active. */
/* - Check for missing application objects on the joining nodes. */
/* - Ensure the required data CRGs are on the joining nodes. */
/* - If the application CRG is active, ensure the required data CRGs are */
/* active. */
/* */
/*****/
static int memberIsJoining(int role,
                          int priorRole,
                          Qcst_EXTP0100_t *crgData,
                          EpData *epData) {

/*-----*/
/* */
/* Ensure the data area status on this node starts out indicating */
/* the application is not running if this node is not the primary. */

```

```

/*                                                                 */
/*-----*/
if (role != QcstPrimaryNodeRole) {
    setApp1CrgDataArea(App1_Ended);
}

/*-----*/
/*                                                                 */
/* If a single node is rejoining the cluster, you may do a certain set of */
/* actions. Whereas if the nodes in a cluster which became partitioned */
/* are merging back together, you may have a different set of actions. */
/*                                                                 */
/*-----*/
if (crgData->Action_Code_Dependent_Data == QcstJoin) {
    /* Do actions for a node joining. */
}
else {
    /* Do actions for partitions merging. */
}

return QcstSuccessful;
} /* end memberIsJoining() */

/*****
/*                                                                 */
/* Action code = QcstCrgAcFailover */
/*                                                                 */
/* Cluster resource services on a particular node(s) has failed or ended */
/* for this cluster resource group. The Failover action code is passed */
/* regardless of whether the CRG is active or inactive. Failover can */
/* happen for a number of reasons: */
/*                                                                 */
/* - an operator canceled the CRG job on a node. Action code dependent */
/* data of QcstMemberFailure will be passed. */
/* - cluster resource services was ended on the node (for example, the */
/* QSYSWRK subsystem was ended with CRS still active). Action code */
/* dependent data of QcstNodeFailure will be passed. */
/* - the application for an application CRG has failed on the primary */
/* node and could not be restarted there. The CRG is Active. */
/* Action code dependent data of QcstApp1Failure will be passed. */
/* - the node failed (such as a power failure). Action code dependent */
/* data of QcstNodeFailure will be passed. */
/* - The cluster has become partitioned due to some communication failure */
/* such as a communication line or LAN failure. The Failover action */
/* code is passed to recovery domain nodes in the majority partition. */
/* Nodes in the minority partition see the End action code. Action */
/* code dependent data of QcstPartitionFailure will be passed. */
/* - A node in the CRG's recovery domain is being ended with the */
/* QcstEndClusterNode API. The node being ended will see the End Node */
/* action code. All other nodes in the recovery domain will see the */
/* Failover action code. Action code dependent data of QcstEndNode */
/* will be passed for the Failover action code. */
/* - An active recovery domain node for an active CRG is being removed */
/* from the cluster with the QcstRemoveClusterNodeEntry API. Action */
/* code dependent data of QcstRemoveNode will be passed. If an */
/* inactive node is removed for an active CRG, or if the CRG is */
/* inactive, an action code of Remove Node is passed. */
/*                                                                 */
/* The exit program is called regardless of whether or not the CRG is */
/* active. The exit program may have nothing to do if the CRG is not */
/* active. */
/*                                                                 */
/* If the CRG is active and the leaving member was the primary node, */

```

```

/* perform the functions necessary for failover to a new primary. */
/* */
/* The Action_Code_Dependent_Data field can be used to determine if: */
/* - the failure was due to a problem that caused the cluster to become */
/*   partitioned (all CRGs which had the partitioned nodes in the */
/*   recovery domain are affected) */
/* - a node failed or had cluster resource services ended on the node (all */
/*   CRGs which had the failed/ended node in the recovery domain are */
/*   affected) */
/* - only a single CRG was affected (for example a single CRG job was */
/*   canceled on a node or a single application failed) */
/* */
/* */
/* Things to consider: */
/* - Prepare the new primary node so the application can be started. */
/* - The application should NOT be started at this time. The exit */
/*   program will be called again with the QcstCrgAcStart action code if */
/*   the CRG was active when the failure occurred. */
/* - If the application CRG is active, ensure the required data CRGs are */
/*   active. */
/* */
/*****
static int memberIsLeaving(int role,
                          int priorRole,
                          Qcst_EXTP0100_t *crgData,
                          EpData *epData) {

/*-----*/
/* */
/* If the CRG is active, perform failover. Otherwise, nothing to do. */
/* */
/*-----*/
if (crgData->Original_Cluster_Res_Grp_Stat == QcstCrgActive) {

/*-----*/
/* */
/* The CRG is active. Determine if my role has changed and I am now */
/* the new primary. */
/* */
/*-----*/

if (priorRole != role && role == QcstPrimaryNodeRole) {

/*-----*/
/* */
/* I was not the primary but am now. Do failover actions but don't */
/* start the application at this time because this exit program will */
/* be called again with the Start action code. */
/* */
/*-----*/

/*-----*/

/*-----*/
/* */
/* Ensure the data area status on this node starts out indicating */
/* the application is not running. */
/* */
/*-----*/

setApplCrgDataArea(Appl_Ended);

```

```

/*-----*/
/*
/* If the application has no actions to do on the Start action code */
/* and will become active as soon as the takeover IP address is */
/* activated, then this code should be uncommented. This code will */
/* determine if all CRGs that this application CRG is dependent upon */
/* are ready. If this check fails, return failure from the action */
/* code. */
/*
/*
/*-----*/
/* if (QcstSuccessful != checkDependCrgDataArea(MaxWaitSeconds)) */
/* return QcstFailWithOutRestart; */

}
}

return QcstSuccessful;
} /* end memberIsLeaving() */

/*****
/*
/* Action code = QcstCrgAcSwitchover */
/*
/* The QcstInitiateSwitchOver API was called. The first backup node in */
/* the cluster resource group's recovery domain is taking over as the */
/* primary node and the current primary node is being made the last backup.*/
/*
/* Things to consider: */
/* - Prepare the new primary node so the application can be started. */
/* - The application should NOT be started at this time. The exit */
/* program will be called again with the QcstCrgAcStart action code. */
/* - The job running the application is canceled and the IP takeover */
/* address is ended prior to the exit program being called on the */
/* current primary. */
/* - Ensure required data or device CRGs have switched over and are */
/* active. */
/*
/*****
static int switchPrimary(int role,
                        int priorRole,
                        Qcst_EXTP0100_t *crgData,
                        EpData *epData) {

/*-----*/
/*
/* See if I am the old primary. */
/*
/*-----*/
if (priorRole == QcstPrimaryNodeRole) {

/*-----*/
/*
/* Do what ever needs to be done to cleanup the old primary before the */
/* switch. Remember that that job which was running the exit program */
/* which started the application was canceled already. */
/*
/* One example may be to clean up any processes holding locks on the */
/* database. This may have been done by the application cancel */
/* handler if one was invoked. */
/*
/*-----*/
}
}

```

```

/*-----*/
/*
/* I'm not the old primary. See if I'm the new primary.
/*
/*
/*-----*/
else if (role == QcstPrimaryNodeRole) {

/*-----*/
/*
/* Do what ever needs to be done on the new primary before the
/* application is started with the QcstCrgAcStart action code.
/*
/*
/*-----*/

/*-----*/
/*
/* Ensure the data area status on this nodes starts out indicating
/* the application is not running.
/*
/*
/*-----*/
setApp1CrgDataArea(App1_Ended);

/*-----*/
/*
/* If the application has no actions to do on the Start action code
/* and will become active as soon as the takeover IP address is
/* activated, then this code should be uncommented. This code will
/* determine if all CRGs that this application CRG is dependent upon
/* are ready. If this check fails, return failure from the action
/* code.
/*
/*
/*-----*/
/*
/* if (QcstSuccessful != checkDependCrgDataArea(MaxWaitSeconds))
/* return QcstFailWithOutRestart;
/*
}
else {

/*-----*/
/*
/* This node is one of the other backup nodes or it is a replicate
/* node. If there is anything those nodes must do, do it here. If
/* not, remove this else block.
/*
/*
/*-----*/

/*-----*/
/*
/* Ensure the data area status on this nodes starts out indicating
/* the application is not running.
/*
/*
/*-----*/
setApp1CrgDataArea(App1_Ended);
}

return QcstSuccessful;

```

```

} /* end switchPrimary() */

/*****
/*
/* Action code = QcstCrgAcAddNode
/*
/* The QcstAddNodeToRcvyDomain API was called. A new node is being added
/* to the recovery domain of a cluster resource group.
/*
/* Things to consider:
/* - A new node is being added to the recovery domain. See the
/* considerations in the createCrg() function.
/* - If this CRG is enabled to use the QcstDistributeInformation API,
/* the user queue needed by that API could be created at this time.
/*
*****/
static int addNode(int role,
                  int priorRole,
                  Qcst_EXTP0100_t *crgData,
                  EpData *epData) {

/*-----*/

/*
/* Determine if I am the node being added.
/*
/*-----*/

if (0 == memcmp(&crgData->This_Nodes_ID,
               &crgData->Changing_Node_ID,
               sizeof(Qcst_Node_Id_t)))
{
/*-----*/

/*
/* Set the status of the data area on this new node.
/*
/*-----*/

setApp1CrgDataArea(App1_Ended);

/*-----*/

/*
/* Create the queue needed by the Distribute Information API.
/*
/*-----*/

if (0 == memcmp(&crgData->DI_Queue_Name,
               Nulls,
               sizeof(crgData->DI_Queue_Name)))
{
}

return QcstSuccessful;
} /* end addNode()
*/

/*****

```

```

/* */
/* Action code = QcstCrgAcRemoveNode */
/* */
/* The QcstRemoveNodeFromRcvyDomain or the QcstRemoveClusterNodeEntry */
/* API was called. A node is being removed from the recovery domain of */
/* a cluster resource group or it is being removed entirely from the */
/* cluster. */
/* */
/* This action code is seen by: */
/* For the QcstRemoveClusterNodeEntry API: */
/* - If the removed node is active and the CRG is Inactive, all nodes in */
/* the recovery domain including the node being removed see this */
/* action code. The nodes NOT being removed see action code dependent */
/* data of QcstNodeFailure. */
/* - If the removed node is active and the CRG is Active, the node being */
/* removed sees the Remove Node action code. All other nodes in the */
/* recovery domain see an action code of Failover and action code */
/* dependent data of QcstNodeFailure. */
/* - If the node being removed is not active in the cluster, all nodes */
/* in the recovery domain will see this action code. */
/* For the QcstRemoveNodeFromRcvyDomain API: */
/* - All nodes see the Remove Node action code regardless of whether or */
/* not the CRG is Active. Action code dependent data of */
/* QcstRmvRcvyDmnNode will also be passed. */
/* */
/* Things to consider: */
/* - You may want to cleanup the removed node by deleting objects no */
/* longer needed there. */
/* - The job running the application is canceled and the IP takeover */
/* address is ended after the exit program is called if this is the */
/* primary node and the CRG is active. */
/* - If subsystems or server jobs were started as a result of the */
/* QcstCrgAcStart action code, end them here or consolidate all logic */
/* to end the application in the cancelHandler() since it will be */
/* invoked for all Cluster Resource Services APIs which must end the */
/* application on the current primary. */
/* */
/*****/
static int rmvNode(int role,
                  int priorRole,
                  Qcst_EXTP0100_t *crgData,
                  EpData *epData) {

/*-----*/

/* */
/* Determine if I am the node being removed. */
/* */
/*-----*/

    if (0 == memcmp(&crgData->This_Nodes_ID,
                   &crgData->Changing_Node_ID,
                   sizeof(Qcst_Node_Id_t)))
    {

/*-----*/
        /* */
        /* End the application if it is running on this node. */
        /* */
        /*-----*/

        endApplication(QcstCrgAcRemoveNode, role, priorRole, crgData,
epData);

    }

```



```

    return QcstSuccessful;
} /* end rmvNode */

/*****
/*
/* Action code = QcstCrgAcChange
/*
/* The QcstChangeClusterResourceGroup API was called. Some attribute
/* or information stored in the cluster resource group object is being
/* changed. Note that not all changes to the CRG object cause the exit
/* program to be called. As of V5R1M0, only these changes will cause the
/* exit program to be called-
/* - the current recovery domain is being changed
/* - the preferred recovery domain is being changed
/*
/* If any of the above changes are being made but additionally the exit
/* program is being changed to *NONE, the exit program is not called.
/*
/* Things to consider:
/* - None unless changing the recovery domain affects information or
/* processes for this cluster resource group. Note that the primary
/* node cannot be changed with the QcstChangeClusterResourceGroup API
/* if the CRG is active.
/*
*****/
static int chgCrg(int role,
                 int priorRole,
                 Qcst_EXTP0100_t *crgData,
                 EpData *epData) {

    return QcstSuccessful;
} /* end chgCrg() */

/*****
/*
/* Action code = QcstCrgAcDeleteCommand
/*
/* The Delete Cluster Resource Group (DLTCRG) CL command has been called
/* to delete a cluster resource group object, the QcstDeleteCluster API
/* has been called, or the QcstRemoveClusterNodeEntry API has been called.
/* In each case, cluster resource services is not active on the cluster
/* node where the command or API was called. Thus, this function is not
/* distributed cluster wide but occurs only on the node where the CL
/* command or API was called.
/*
/* If the QcstDeleteCluster API was used, action code dependent data of
/* QcstDltCluster is passed.
/*
/* See the considerations in the deleteCrg() function
/*
*****/
static int deleteCrgWithCmd(int role,
                           int doesNotApply,
                           Qcst_EXTP0100_t *crgData,
                           EpData *epData) {

    return QcstSuccessful;
} /* end deleteCrgWithCmd() */

/*****
/*
/* Action code = QcstCrgEndNode
/*
/* The QcstEndClusterNode API was called or a CRG job was canceled.
*****/

```

```

/* */
/* The QcstCrgEndNode action code is passed to the exit program only on the*/
/* node being ended or where the CRG job was canceled. On the node where */
/* a Cluster Resource Services job is canceled, action code dependent data*/
/* of QcstMemberFailure will be passed. */
/* When Cluster Resource Services ends on this node or the CRG job ends, it*/
/* will cause all other nodes in the cluster to go through failover */
/* processing. The action code passed to all other nodes will be */
/* QcstCrgAcFailover. Those nodes will see action code dependent data of */
/* QcstMemberFailure if a CRG job is canceled or QcstNodeFailure if the */
/* node is ended. */
/* */
/* Things to consider: */
/* - The job running the application is canceled and the IP takeover */
/* address is ended after the exit program is called if this is the */
/* primary node and the CRG is active. */
/* - If subsystems or server jobs were started as a result of the */
/* QcstCrgAcStart action code, end them here. */
/* */
/*****/
static int endNode(int role,
                  int priorRole,
                  Qcst_EXTP0100_t *crgData,
                  EpData *epData) {

/*-----*/
/* */
/* End the application if it is running on this node. */
/* */
/*-----*/

endApplication(QcstCrgEndNode, role, priorRole, crgData, epData);

return QcstSuccessful;
} /* end endNode() */

/*****/
/* */
/* Action code = QcstCrgAcChgNodeStatus */
/* */
/* The QcstChangeClusterNodeEntry API was called. The status of a node */
/* is being changed to failed. This API is used to inform cluster resource*/
/* services that the node did not partition but really failed. */
/* */
/* Things to consider: */
/* - The exit program was called previously with an action code of */
/* QcstCrgAcEnd if the CRG was active or an action code of */
/* QcstCrgAcFailover if the CRG was inactive because cluster resource */
/* services thought the cluster had become partitioned. The user is */
/* now telling cluster resource services that the node really failed */
/* instead of partitioned. The exit program has something to do only */
/* if it performed some action previously that needs to be changed now */
/* that node failure can be confirmed. */
/* */
/*****/
static int chgNodeStatus(int role,
                        int priorRole,
                        Qcst_EXTP0100_t *crgData,
                        EpData *epData) {

return QcstSuccessful;
} /* end chgNodeStatus() */

/*****/

```

```

/* */
/* Action code = QcstCrgAcCancelFailover */
/* */
/* Cluster resource services on the primary node has failed or ended */
/* for this cluster resource group. A message was sent to the failover */
/* message queue specified for the CRG, and the result of that message */
/* was to cancel the failover. This will change the status of the CRG to */
/* inactive and leave the primary node as primary. */
/* */
/* Things to consider: */
/* - The primary node is no longer participating in cluster activities. */
/* The problem which caused the primary node to fail should be fixed */
/* so that the CRG may be started again. */
/* */
/*****/
static int cancelFailover(int role,
                          int priorRole,
                          Qcst_EXTP0100_t *crgData,
                          EpData *epData) {

    return QcstSuccessful;
} /* end cancelFailover() */

/*****/
/* */
/* Action code = exit program does not know it yet */
/* */
/* A new action code has been passed to this exit program. This can occur */
/* after a new i5/OS release has been installed and some new cluster API */
/* was called or some new cluster event occurred. The logic in this exit */
/* program has not yet been updated to understand the new action code. */
/* */
/* Two different strategies could be used for the new action code. The */
/* correct strategy is dependent upon the kinds of things this particular */
/* exit program does for the application. */
/* */
/* One strategy is to not do anything and return a successful return code. */
/* This allows the new cluster API or event to run to completion. It */
/* allows the function to be performed even though this exit program */
/* did not understand the new action code. The risk, though, is that the */
/* exit program should have done something and it did not. At a minimum, */
/* you may want to log some kind of error message about what happened so */
/* that programming can investigate and get the exit program updated. */
/* */
/* The opposite strategy is to return an error return code such as */
/* QcstFailWithRestart. Of course doing this means that the new cluster */
/* API or event cannot be used until the exit program is updated for the */
/* new action code. Again, logging some kind of error message for */
/* programming to investigate would be worthwhile. */
/* */
/* Only the designer of the exit program can really decide which is the */
/* better course of action. */
/* */
/*****/
static int newActionCode(int role,
                        int doesNotApply,
                        Qcst_EXTP0100_t *crgData,
                        EpData *epData) {

/*-----*/
/* */
/* Add logic to log an error somewhere - operator message queue, job */
/* log, application specific error log, etc. so that the exit program */
/* gets updated to properly handle the new action code. */
/* */

```

```

/* Note that if this is left coded as it is, this is the "don't do */
/* anything" strategy described in the prologue above. */
/* */
/*-----*/

return QcstSuccessful;
} /* end newActionCode() */

/*****/
/* */
/* Action code = QcstCrgAcUndo */
/* */
/* Note: The exit program is never called with an undo action code for */
/* any of these prior action codes: */
/* QcstCrgAcChgNodeStatus */
/* QcstCrgAcDelete */
/* QcstCrgAcDeleteCommand */
/* QcstCrgAcEndNode */
/* QcstCrgAcRemoveNode (If the node being removed is active in the */
/* cluster and the API is Remove Cluster Node. */
/* The Remove Node From Recovery Domain will call */
/* with Undo and the Remove Cluster Node API will */
/* call with Undo if the node being removed is */
/* inactive. */
/* QcstCrgAcRestart */
/* QcstCrgAcUndo */
/* */
/* APIs that call an exit program do things in 3 steps. */
/* 1. Logic which must be done prior to calling the exit program. */
/* 2. Call the exit program. */
/* 3. Logic which must be done after calling the exit program. */
/* */
/* Any errors that occur during steps 2 or 3 result in the exit program */
/* being called again with the undo action code. This gives the exit */
/* program an opportunity to back out any work performed when it was first */
/* called by the API. The API will also be backing out any work it */
/* performed trying to return the state of the cluster and cluster objects */
/* to what it was before the API was called. */
/* */
/* It is suggested that the following return codes be returned for the */
/* specified action code as that return code will result in the most */
/* appropriate action being taken. */
/* */
/* QcstCrgAcInitialize: QcstSuccessful; The CRG is not created. */
/* QcstCrgAcStart: QcstSuccessful; The CRG is not started. */
/* QcstCrgAcEnd: QcstFailWithOutRestart; The CRG is set to Indoubt*/
/* The cause of the failure needs to*/
/* investigated. */
/* QcstCrgAcReJoin: QcstFailWithOutRestart; The CRG is set to Indoubt*/
/* The cause of the failure needs to*/
/* investigated. */
/* QcstCrgAcFailover: QcstFailWithOutRestart; The CRG is set to Indoubt*/
/* The cause of the failure needs to*/
/* investigated. */
/* QcstCrgAcSwitchover: QcstFailWithOutRestart; The CRG is set to Indoubt*/
/* The cause of the failure needs to*/
/* investigated. */
/* QcstCrgAcAddNode: QcstSuccessful; The node is not added. */
/* QcstCrgAcRemoveNode: QcstFailWithOutRestart; The CRG is set to Indoubt*/
/* The cause of the failure needs to*/
/* investigated. */
/* QcstCrgAcChange: QcstSuccessful; The recovery domain is not */
/* changed. */
/* */
/*****/

```

```

static int undoPriorAction(int role,
                          int priorRole,
                          Qcst_EXTP0100_t *crgData,
                          EpData *epData) {

/*-----*/
/*
/* The prior action code defines what the exit program was doing when
/* it failed, was canceled, or returned a non successful return code.
/*
/*
/*-----*/
    if (crgData->Prior_Action_Code &lt;= MaxAc )
        return (*undoFcn[crgData-&lt;Prior_Action_Code])
                (role, priorRole, crgData,
epData);
    else

/*-----*/
/*
/* IBM has defined a new action code in a new operating system release
/* and this exit program has not yet been updated to handle it. Take a
/* default action for now.
/*
/*
/*-----*/
    return newActionCode(role, priorRole, crgData, epData);
} /* end undoPriorAction()

/*****
/*
/* Action code = QcstCrgAcUndo
/*
/* Prior action code = QcstCrgAcInitialize
/*
/* Things to consider:
/* The CRG will not be created. Objects that might have been created
/* on nodes in the recovery domain should be deleted since a subsequent
/* create could fail if those objects already exist.
/*
/*
/*****
static int undoCreateCrg(int role,
                        int doesNotApply,
                        Qcst_EXTP0100_t *crgData,
                        EpData *epData) {

    return QcstSuccessful;
} /* end undoCreateCrg()

/*****
/*
/* Action code = QcstCrgAcUndo
/*
/* Prior action code = QcstCrgAcStart
/*
/* Things to consider:
/* Cluster Resource Services failed when it was finishing the Start CRG
/* API after it had already called the exit program with the Start
/* Action code.
/*
/* On the primary node, the exit program job which is running the
/* application will be canceled. The exit program will then be called
/* with the Undo action code.
/*
/*

```

```

/* All other nodes in the recovery domain will be called with the Undo */
/* action code. */
/* */
/*****/
static int undoStartCrg(int role,
                       int doesNotApply,
                       Qcst_EXTP0100_t *crgData,
                       EpData *epData) {

    return QcstSuccessful;
} /* end undoStartCrg() */

/*****/
/* */
/* Action code = QcstCrgAcUndo */
/* */
/* Prior action code = QcstCrgAcEnd */
/* */
/* Things to consider: */
/* The CRG will not be ended. If the exit program did anything to bring */
/* down the application it can either restart the application or it can */
/* decide to not restart the application. If the application is not */
/* restarted, the return code should be set to QcstFailWithOutRestart so */
/* the status of the CRG is set to Indoubt. */
/* */
/*****/
static int undoEndCrg(int role,
                     int doesNotApply,
                     Qcst_EXTP0100_t *crgData,
                     EpData *epData) {

    return QcstFailWithOutRestart;
} /* end undoEndCrg() */

/*****/
/* */
/* Action code = QcstCrgAcUndo */
/* */
/* Prior action code = QcstCrgAcReJoin */
/* */
/* Things to consider: */
/* An error occurred which won't allow the member to join this CRG */
/* group. Anything done for the Join action code needs to be looked at */
/* to see if something must be undone if this member is not an active */
/* member of the CRG group. */
/* */
/*****/
static int undoMemberIsJoining(int role,
                               int doesNotApply,
                               Qcst_EXTP0100_t *crgData,
                               EpData *epData) {

    return QcstFailWithOutRestart;
} /* end undoMemberIsJoining() */

/*****/
/* */
/* Action code = QcstCrgAcUndo */
/* */
/* Prior action code = QcstCrgAcFailover */
/* */
/* Things to consider: */
/* This does not mean that the node failure or failing member is being */
/* undone. That failure is irreversible. What it does mean is that the */

```

```

/* exit program returned an error from the Failover action code or */
/* Cluster Resource Services ran into a problem after it called the exit */
/* program. If the CRG was active when Failover was attempted, it is */
/* not at this point. End the resilient resource and expect a human to */
/* look into the failure. After the failure is corrected, the CRG will */
/* must be started with the Start CRG API. */
/* */
/* */
/*****/
static int undoMemberIsLeaving(int role,
                               int doesNotApply,
                               Qcst_EXTPO100_t *crgData,
                               EpData *epData) {

    return QcstFailWithOutRestart;
} /* end undoMemberIsLeaving() */

/*****/
/* */
/* Action code = QcstCrgAcUndo */
/* */
/* Prior action code = QcstCrgAcSwitchover */
/* */
/* Things to consider: */
/* Some error occurred after the point of access was moved from the */
/* original primary and before it could be brought up on the new primary.*/
/* The IP address was ended on the original primary before moving the */
/* point of access but is started on the original primary again. Cluster*/
/* Resource Services will now attempt to move the point of access back */
/* to the original primary. The application exit program and IP takeover*/
/* address will be started on the original primary. */
/* */
/* */
/*****/
static int undoSwitchPrimary(int role,
                             int doesNotApply,
                             Qcst_EXTPO100_t *crgData,
                             EpData *epData) {

    return QcstFailWithOutRestart;
} /* end undoSwitchPrimary() */

/*****/
/* */
/* Action code = QcstCrgAcUndo */
/* */
/* Prior action code = QcstCrgAcAddNode */
/* */
/* Things to consider: */
/* If objects were created on the new node, they should be removed so */
/* that a subsequent Add Node to aRecovery Domain does not fail if it */
/* attempts to create objects again. */
/* */
/* */
/*****/
static int undoAddNode(int role,
                       int doesNotApply,
                       Qcst_EXTPO100_t *crgData,
                       EpData *epData) {

    return QcstSuccessful;
} /* end undoAddNode() */

/*****/

```

```

/* */
/* Action code = QcstCrgAcUndo */
/* */
/* Prior action code = QcstCrgAcRemoveNode */
/* */
/* Things to consider: */
/* The node is still in the recovery domain. If objects were removed */
/* from the node, they should be added back. */
/* */
/*****/
static int undoRmvNode(int role,
                      int doesNotApply,
                      Qcst_EXTP0100_t *crgData,
                      EpData *epData) {

    return QcstFailWithoutRestart;
} /* end undoRmvNode() */

/*****/
/* */
/* Action code = QcstCrgAcUndo */
/* */
/* Prior action code = QcstCrgAcChange */
/* */
/* Things to consider: */
/* Changes to the CRG will be backed out so that the CRG and its */
/* recovery domain look just like it did prior to the attempted change. */
/* Any changes the exit program made should also be backed out. */
/* */
/*****/
static int undoChgCrg(int role,
                     int doesNotApply,
                     Qcst_EXTP0100_t *crgData,
                     EpData *epData) {

    return QcstSuccessful;
} /* end undoChgCrg() */

/*****/
/* */
/* Action code = QcstCrgAcUndo */
/* */
/* Prior action code = QcstCrgAcCancelFailover */
/* */
/* Things to consider: */
/* This does not mean that the node failure or failing member is being */
/* undone. That failure is irreversible. What it does mean is that */
/* Cluster Resource Services ran into a problem after it called the exit */
/* program. The CRG will be InDoubt regardless of what is returned from */
/* this exit program call. Someone will need to manually look into the */
/* the failure. After the failure is corrected, the CRG will must be */
/* started with the Start CRG API. */
/* */
/* */
/*****/
static int undoCancelFailover(int role,
                              int doesNotApply,
                              Qcst_EXTP0100_t *crgData,
                              EpData *epData) {

    return QcstSuccessful;
} /* end undoCancelFailover() */

/*****/

```



```

/*
/* A simple routine to take a null terminated object name and a null
/* terminated library name and build a 20 character non-null terminated
/* qualified name.
/*
/*
/*****
static void bldDataAreaName(char *objName, char* libName, char *qualName) {

    memset(qualName, 0x40, 20);
    memcpy(qualName, objName, strlen(objName));
    qualName += 10;
    memcpy(qualName, libName, strlen(libName));
    return;
} /* end bldDataAreaName

/*****
/*
/* The data area is checked to see if all the CRGs that this application
/* is dependent upon are ready. If they are not ready, a wait for a
/* certain amount of time is performed and the data area is checked again.
/* This check, wait loop continues until all dependent CRGs become ready or
/* until the maximum wait time has been reached.
/*
/* The length of the wait can be changed to some other value if a
/* particular situation would be better with shorter or longer wait times.
/*
/*
/*****
static int checkDependCrgDataArea(unsigned int maxWaitTime) {

    Qus_EC_t errCode = { sizeof(Qus_EC_t), 0 };
    char dataAreaName[20];
    struct {
        Qwc_Rdtaa_Data_Returned_t stuff;
        char ready;
    } data;

/*-----*/
/*
/* This is an accumulation of the time waited for the dependent CRGs to
/* become ready.
/*
/*-----*/
    unsigned int timeWaited = 0;

/*-----*/
/*
/* Build definition of the amount of time to wait.
/*
/*-----*/
    _MI_Time    timeToWait;
    int hours   = 0;
    int minutes = 0;
    int seconds = WaitSecondsIncrement;
    int hundreths = 0;
    short int options = _WAIT_NORMAL;
    mitime( &timeToWait, hours, minutes, seconds, hundreths );

/*-----*/
/*
/* Build the qualified name of the data area.
/*
/*

```

```

/*-----*/
    bldDataAreaName(DependCrgDataArea, ApplLib, dataAreaName);

/*-----*/
/*
/* Get the data from the data area that indicates whether or not the
/* CRGs are all ready. This data area is updated by the High
/* Availability Business Partners when it is ok for the application to
/* proceed.
/*
/*
/*-----*/
    QWCRDTAA(&data,
            sizeof(data),
            dataAreaName,
            offsetof(Qcst_HAAPPO_t,Data_Status)+1, /* API wants a 1 origin */
            sizeof(data.ready),
            &errCode);

/*-----*/
/*
/* If the dependent CRGs are not ready, wait for a bit and check again.
/*
/*
/*-----*/
    while (data.ready != Data_Available) {

/*-----*/
/*
/* If the dependent CRGs are not ready after the maximum wait time,
/* return an error. Consider logging some message to describe why the
/* application did not start so that the problem can be looked into.
/*
/*
/*-----*/
        if (timeWaited >= maxWaitTime)
            return QcstFailWithOutRestart;

/*-----*/
/*
/* Wait to allow the data CRGs to become ready.
/*
/*
/*-----*/
        waittime(&timeToWait, options);
        timeWaited += WaitSecondsIncrement;

/*-----*/
/*
/* Get information from the data area again to see if the data CRGs are
/* ready.
/*
/*
/*-----*/
        QWCRDTAA(&data,
                sizeof(data),
                dataAreaName,
                offsetof(Qcst_HAAPPO_t,Data_Status)+1, /* API wants a 1 origin */
                sizeof(data.ready),
                &errCode);
    }

```

```

    return QcstSuccessful;
} /* end checkDependCrgDataArea */

/*****
/*
/* The application CRG data area is updated to indicate that the
/* application is running or to indicate it is not running. This data area
/* information is used by the High Availability Business Partners to
/* coordinate the switchover activities between CRGs that have dependencies
/* on each other.
/*
/*
/*****
static void setApp1CrgDataArea(char status) {

    char cmd[54];
    char cmdEnd[3] = {0x00, '}', 0x00};

/*-----*/
/*
/* Set up the CL command string with the data area library name, the data
/* area name, and the character to put into the data area. Then run the
/* CL command.
/*
/*
/*-----*/
    memcpy(cmd, "CHGDTAARA DTAARA(", strlen("CHGDTAARA DTAARA")+1);
    strcat(cmd, ApplLib);
    strcat(cmd, "/");
    strcat(cmd, App1CrgDataArea);
    strcat(cmd, " (425 1)) VALUE("); /* @A1C */
    cmdEnd[0] = status;
    strcat(cmd, cmdEnd);

    system(cmd);

    return;
} /* end setApp1CrgDataArea */

/*****
/*
/* This function is called any time the exit program receives an exception
/* not specifically monitored for by some other exception handler. Add
/* appropriate logic to perform cleanup functions that may be required.
/*
/* A failure return code is then set and control returns to the operating
/* system. The job this exit program is running in will then end.
/*
/*
/* When this function gets called, myData->role may still contain the
/* UnknownRole value if an exception occurred before this node's role
/* value was set. To be completely correct, the role should be tested
/* for UnknownRole before making any decisions based upon the value of
/* role.
/*
/*
/*****
static void unexpectedExceptionHandler(_INTRPT_Hndlr_Parms_T
*exData) {

/*-----*/
/*
/* Get a pointer to the structure containing data that is passed to the
/* exception handler.
/*
/*

```

```

/*-----*/
HandlerDataT *myData = (HandlerDataT *)exData->Com_Area;

/*-----*/
/*
/* Perform as much cleanup function as necessary. Some global state */
/* information may must be kept so the exception handler knows what */
/* steps were completed before the failure occurred and thus knows what */
/* cleanup steps must be performed. This state information could be */
/* kept in the HandlerDataT structure or it could be kept in some other */
/* location that this function can address. */
/* */
/*-----*/

/*-----*/
/*
/* If this is the primary node and the application was started, end it. */
/* The application is ended because the exit program will be called again*/
/* with the Restart action code and want the restartCrg() function to */
/* always work the same way. In addition, ending the application may */
/* clear up the condition that caused the exception. */
/* If possible, warn users and have them stop using the application so */
/* things are done in an orderly manner. */
/* */
/*-----*/

endApplication(myData->actionCode,
               myData->role,
               myData->priorRole,
               myData->crgData,
               myData->epData);

/*-----*/
/*
/* Set the exit program return code. */
/* */
/*-----*/

*myData->retCode = QcstFailWithRestart;

/*-----*/
/*
/* Let the exception percolate up the call stack. */
/* */
/*-----*/

return;
} /* end unexpectedExceptionHandler */

/*****
/*
/* This function is called any time the job this exit program is running in*/
/* is canceled. The job could be canceled due to any of the following */
/* (the list is not intended to be all inclusive)- */
/* - an API cancels an active application CRG. The End CRG, Initiate */
/* Switchover, End Cluster Node, Remove Cluster Node or Delete Cluster */
/* API cancels the job which was submitted when the exit program was */
/* called with a Start action code. */
/* - operator cancels the job from some operating system display such as */
/* Work with Active Jobs */
/* - the subsystem this job is running in is ended */
*****/

```

```

/* - all subsystems are ended */
/* - the system is powered down */
/* - an operating system machine check occurred */
/*
/* When this function gets called, myData->role may still contain the
/* UnknownRole value if cancelling occurred before this node's role
/* value was set. To be completely correct, the role should be tested
/* for UnknownRole before making any decisions based upon the value of
/* role.
/*
/*****
static void cancelHandler(_CNL_Hndlr_Parms_T *cnldata) {

/*-----*/
/*
/* Get a pointer to the structure containing data that was passed to the
/* cancel handler.
/*
/*-----*/
HandlerDataT *myData = (HandlerDataT *)cnldata->Com_Area;

/*-----*/
/*
/* Perform as much cleanup function as necessary. Some global state
/* information may must be kept so the cancel handler knows what
/* steps were completed before the job was canceled and thus knows if
/* the function had really completed successfully or was only partially
/* complete and thus needs some cleanup to be done. This state
/* information could be kept in the HandlerDataT structure or it could
/* be kept in some other location that this function can address.
/*
/*-----*/

/*-----*/

/*
/* This job is being canceled. If I was running the application as a
/* result of the Start or Restart action codes, end the application now.
/* This job is being canceled because a Switch Over or some other
/* Cluster Resource Services API was used which affects the primary node
/* or someone did a cancel job with a CL command, from a system display,
/* etc.
/*
/*-----*/
endApplication(myData->actionCode,
               myData->role,
               myData->priorRole,
               myData->crgData,
               myData->epData);

/*-----*/
/*
/* Set the exit program return code.
/*
/*-----*/
myData->retCode = QcstSuccessful;

/*-----*/

```

```

/*
/* Return to the operating system for final ending of the job.
/*
/*
/*-----*/
return;
} /* end cancelHandler */

/*****
/*
/* A common routine used to end the application by various action code
/* functions, the exception handler, and the cancel handler.
/*
/*
/*****
static void endApplication(unsigned int actionCode,
                          int role,
                          int priorRole,
                          Qcst_EXTP0100_t *crgData,
                          EpData *epData) {

    if ( role == QcstPrimaryNodeRole
        &&
            crgData->Original_Cluster_Res_Grp_Stat == QcstCrgActive)
    {
/*-----*/
/*
/* Add logic to end the application here. You may need to add logic
/* to determine if the application is still running because this
/* function could be called once for an action code and again from
/* the cancel handler (End CRG is an example).
/*
/*
/*-----*/

/*-----*/
/*
/* After the application has ended, update the data area to indicate
/* the application is no longer running.
/*
/*
/*-----*/
setApp1CrgDataArea(App1_Ended);
    }

    return;
} /* end endApplication */

/*****
/*
/* Print out the data passed to this program.
/*
/*
/*****
static void printParms(int actionCode,
                      int role,
                      int priorRole,
                      Qcst_EXTP0100_t *crgData,
                      EpData *epData) {

    unsigned int i;
    char *str;

```

```

/* Print the action code. */
printf("%s", "Action_Code = ");
printActionCode(actionCode);

/* Print the action code dependent data. */
printf("%s", " Action_Code_Dependent_Data = ");
switch (crgData->Action_Code_Dependent_Data) {
    case QcstNoDependentData: str = "QcstNoDependentData";
                             break;
    case QcstMerge:          str = "QcstMerge";
                             break;
    case QcstJoin:           str = "QcstJoin";
                             break;
    case QcstPartitionFailure: str = "QcstPartitionFailure";
                             break;
    case QcstNodeFailure:    str = "QcstNodeFailure";
                             break;
    case QcstMemberFailure: str = "QcstMemberFailure";
                             break;
    case QcstEndNode:        str = "QcstEndNode";
                             break;
    case QcstRemoveNode:     str = "QcstRemoveNode";
                             break;
    case QcstApplFailure:    str = "QcstApplFailure";
                             break;
    case QcstResourceEnd:    str = "QcstResourceEnd";
                             break;
    case QcstDltCluster:     str = "QcstDltCluster";
                             break;
    case QcstRmvRcvyDmnNode: str = "QcstRmvRcvyDmnNode";
                             break;
    case QcstDltCrg:         str = "QcstDltCrg";
                             break;
    default: str = "unknown action code dependent data";
}
printf("%s \n", str);

/* Print the prior action code. */
printf("%s", " Prior_Action_Code = ");
if (crgData->Prior_Action_Code)
    printActionCode(crgData->Prior_Action_Code);
printf("\n");

/* Print the cluster name. */
printStr(" Cluster_Name = ",
         crgData->Cluster_Name, sizeof(Qcst_Cluster_Name_t));

/* Print the CRG name. */
printStr(" Cluster_Resource_Group_Name = ",
         crgData->Cluster_Resource_Group_Name,
         sizeof(Qcst_Crg_Name_t));

/* Print the CRG type. */
printf("%s \n", " Cluster_Resource_Group_Type =
QcstCrgApplResiliency");

/* Print the CRG status. */
printf("%s", " Cluster_Resource_Group_Status = ");
printCrgStatus(crgData->Cluster_Resource_Group_Status);

/* Print the CRG original status. */
printf("%s", " Original_Cluster_Res_Grp_Stat = ");
printCrgStatus(crgData->Original_Cluster_Res_Grp_Stat);

/* Print the Distribute Information queue name. */
printStr(" DI_Queue_Name = ",

```

```

        crgData->DI_Queue_Name,
sizeof(crgData->DI_Queue_Name));
    printStr(" DI_Queue_Library_Name = ",
        crgData->DI_Queue_Library_Name,
        sizeof(crgData->DI_Queue_Library_Name));

    /* Print the CRG attributes. */
    printf("%s", " Cluster_Resource_Group_Attr = ");
    if (crgData->Cluster_Resource_Group_Attr &
QcstTcpConfigByUsr)
        printf("%s", "User Configures IP Takeover Address");
    printf("\n");

    /* Print the ID of this node. */
    printStr(" This_Nodes_ID = ",
        crgData->This_Nodes_ID, sizeof(Qcst_Node_Id_t));

    /* Print the role of this node. */
    printf("%s %d \n", " this node's role = ", role);

    /* Print the prior role of this node. */
    printf("%s %d \n", " this node's prior role = ", priorRole);

    /* Print which recovery domain this role comes from. */
    printf("%s", " Node_Role_Type = ");
    if (crgData->Node_Role_Type == QcstCurrentRcvyDmn)
        printf("%s \n", "QcstCurrentRcvyDmn");
    else
        printf("%s \n", "QcstPreferredRcvyDmn");

    /* Print the ID of the changing node (if any). */
    printStr(" Changing_Node_ID = ",
        crgData->Changing_Node_ID, sizeof(Qcst_Node_Id_t));

    /* Print the role of the changing node (if any). */
    printf("%s", " Changing_Node_Role = ");
    if (crgData->Changing_Node_Role == -3)
        printf("%s \n", "*LIST");
    else if (crgData->Changing_Node_Role == -2)
        printf("%s \n", "does not apply");
    else
        printf("%d \n", crgData->Changing_Node_Role);

    /* Print the takeover IP address. */
    printStr(" Takeover_IP_Address = ",
        crgData->Takeover_IP_Address,
sizeof(Qcst_TakeOver_IP_Address_t));

    /* Print the job name. */
    printStr(" Job_Name = ", crgData->Job_Name, 10);

    /* Print the CRG changes. */
    printf("%s \n", " Cluster_Resource_Group_Changes = ");
    if (crgData->Cluster_Resource_Group_Changes &
QcstRcvyDomainChange)
        printf(" %s \n", "Recovery domain changed");
    if (crgData->Cluster_Resource_Group_Changes &
QcstTakeOverIpAddrChange)
        printf(" %s \n", "Takeover IP address changed");

    /* Print the failover wait time. */
    printf("%s", "Failover_Wait_Time = ");
    if (crgData->Failover_Wait_Time == QcstFailoverWaitForever)
        printf("%d %s \n", crgData->Failover_Wait_Time, "Wait
forever");
    else if (crgData->Failover_Wait_Time == QcstFailoverNoWait)
        printf("%d %s \n", crgData->Failover_Wait_Time, "No wait");

```



```

else
    printf("%d %s \n", crgData->Failover_Wait_Time, "minutes");

/* Print the failover default action. */
printf("%s", "Failover_Default_Action = ");
if (crgData->Failover_Default_Action == QcstFailoverProceed)
    printf("%d %s \n", crgData->Failover_Default_Action,
"Proceed");
else
    printf("%d %s \n", crgData->Failover_Default_Action,
"Cancel");

/* Print the failover message queue name. */
printStr(" Failover_Msg_Queue = ",
        crgData->Failover_Msg_Queue,
sizeof(crgData->Failover_Msg_Queue));
printStr(" Failover_Msg_Queue_Lib = ",
        crgData->Failover_Msg_Queue_Lib,
sizeof(crgData->Failover_Msg_Queue_Lib));

/* Print the cluster version. */
printf("%s %d \n",
        " Cluster_Version = ", crgData->Cluster_Version);

/* Print the cluster version mod level */
printf("%s %d \n",
        " Cluster_Version_Mod_Level = ",
        crgData->Cluster_Version_Mod_Level);

/* Print the requesting user profile. */
printStr(" Req_User_Profile = ",
        crgData->Req_User_Profile,
sizeof(crgData->Req_User_Profile));

/* Print the length of the data in the structure. */
printf("%s %d \n",
        " Length_Info_Returned = ",
crgData->Length_Info_Returned);

/* Print the offset to the recovery domain array. */
printf("%s %d \n",
        " Offset_Rcvy_Domain_Array = ",
crgData->Offset_Rcvy_Domain_Array);

/* Print the number of nodes in the recovery domain array. */
printf("%s %d \n",
        " Number_Nodes_Rcvy_Domain = ",
crgData->Number_Nodes_Rcvy_Domain);

/* Print the current/new recovery domain. */
printRcvyDomain(" The recovery domain:",
        crgData->Number_Nodes_Rcvy_Domain,
(Qcst_Rcvy_Domain_Array1_t *)
        ((char *)crgData +
crgData->Offset_Rcvy_Domain_Array));

/* Print the offset to the prior recovery domain array. */
printf("%s %d \n",
        " Offset_Prior_Rcvy_Domain_Array = ",
crgData->Offset_Prior_Rcvy_Domain_Array);

/* Print the number of nodes in the prior recovery domain array. */
printf("%s %d \n",
        " Number_Nodes_Prior_Rcvy_Domain = ",
crgData->Number_Nodes_Prior_Rcvy_Domain);

/* Print the prior recovery domain if one was passed. */

```

```

    if (crgData->Offset_Prior_Rcvy_Domain_Array) {
        printRcvyDomain("    The prior recovery domain:",
            crgData->Number_Nodes_Prior_Rcvy_Domain,
            (Qcst_Rcvy_Domain_Array1_t *)
                ((char *)crgData +
                    crgData->Offset_Prior_Rcvy_Domain_Array));
    }

    return;
} /* end printParms */

/*****
/*
/* Print a string for the action code.
/*
/*
*****/
static void printActionCode(unsigned int ac) {

    char *code;
    switch (ac) {
        case QcstCrgAcInitialize: code = "QcstCrgAcInitialize";
            break;
        case QcstCrgAcStart: code = "QcstCrgAcStart";
            break;
        case QcstCrgAcRestart: code = "QcstCrgAcRestart";
            break;
        case QcstCrgAcEnd: code = "QcstCrgAcEnd";
            break;
        case QcstCrgAcDelete: code = "QcstCrgAcDelete";
            break;
        case QcstCrgAcReJoin: code = "QcstCrgAcReJoin";
            break;
        case QcstCrgAcFailover: code = "QcstCrgAcFailover";
            break;
        case QcstCrgAcSwitchover: code = "QcstCrgAcSwitchover";
            break;
        case QcstCrgAcAddNode: code = "QcstCrgAcAddNode";
            break;
        case QcstCrgAcRemoveNode: code = "QcstCrgAcRemoveNode";
            break;
        case QcstCrgAcChange: code = "QcstCrgAcChange";
            break;
        case QcstCrgAcDeleteCommand: code = "QcstCrgAcDeleteCommand";
            break;
        case QcstCrgAcUndo: code = "QcstCrgAcUndo";
            break;
        case QcstCrgAcEndNode: code = "QcstCrgAcEndNode";
            break;
        case QcstCrgAcAddDevEnt: code = "QcstCrgAcAddDevEnt";
            break;
        case QcstCrgAcRmvDevEnt: code = "QcstCrgAcRmvDevEnt";
            break;
        case QcstCrgAcChgDevEnt: code = "QcstCrgAcChgDevEnt";
            break;
        case QcstCrgAcChgNodeStatus: code = "QcstCrgAcChgNodeStatus";
            break;
        case QcstCrgAcCancelFailover: code = "QcstCrgAcCancelFailover";
            break;
        case QcstCrgAcVerificationPhase: code =
"QcstCrgAcVerificationPhase";
            break;
        default: code = "unknown action code";
            break;
    }
    printf("%s", code);
}

```

```

    return;
} /* end printActionCode */

/*****
/*
/* Print the CRG status.
/*
/*
/*****
static void printCrgStatus(int status) {

    char * str;
    switch (status) {
        case QcstCrgActive:          str = "QcstCrgActive";
                                    break;
        case QcstCrgInactive:        str= "QcstCrgInactive";
                                    break;
        case QcstCrgIndoubt:         str = "QcstCrgIndoubt";
                                    break;
        case QcstCrgRestored:        str = "QcstCrgRestored";
                                    break;
        case QcstCrgAddnodePending:  str =
"QcstCrgAddnodePending";
                                    break;
        case QcstCrgDeletePending:   str = "QcstCrgDeletePending";
                                    break;
        case QcstCrgChangePending:   str = "QcstCrgChangePending";
                                    break;
        case QcstCrgEndCrgPending:   str = "QcstCrgEndCrgPending";
                                    break;
        case QcstCrgInitializePending: str =
"QcstCrgInitializePending";
                                    break;
        case QcstCrgRemovenodePending: str =
"QcstCrgRemovenodePending";
                                    break;
        case QcstCrgStartCrgPending: str =
"QcstCrgStartCrgPending";
                                    break;
        case QcstCrgSwitchOverPending: str =
"QcstCrgSwitchOverPending";
                                    break;
        case QcstCrgDeleteCmdPending: str =
"QcstCrgDeleteCmdPending";
                                    break;
        case QcstCrgAddDevEntPending: str =
"QcstCrgAddDevEntPending";
                                    break;
        case QcstCrgRmvDevEntPending: str =
"QcstCrgRmvDevEntPending";
                                    break;
        case QcstCrgChgDevEntPending: str =
"QcstCrgChgDevEntPending";
                                    break;
        case QcstCrgChgNodeStatusPending: str =
"QcstCrgChgNodeStatusPending";
                                    break;
        default: str = "unknown CRG status";
    }
    printf("%s \n", str);

    return;
} /* end printCrgStatus */

/*****
/*

```

```

/* Print the recovery domain. */
/*
/*****
static void printRcvyDomain(char *str,
                           unsigned int count,
                           Qcst_Rcvy_Domain_Array1_t *rd) {

    unsigned int i;
    printf("\n %s \n", str);
    for (i=1; i<=count; i++) {
        printStr("    Node_ID = ", rd->Node_ID,
sizeof(Qcst_Node_Id_t));
        printf("%s %d \n", "    Node_Role = ", rd->Node_Role);
        printf("%s", "    Membership_Status = ");
        switch (rd->Membership_Status) {
            case 0: str = "Active";
                break;
            case 1: str = "Inactive";
                break;
            case 2: str = "Partition";
                break;
            default: str = "unknown node status";
        }
        printf("%s \n", str);
        rd++;
    }
    return;
} /* end printRcvyDomain */

/*****
/*
/* Concatenate a null terminated string and a non null terminated string
/* and print it.
/*
/*
/*****
static void printStr(char *s1, char *s2, unsigned int len) {

    char buffer[132];
    memset(buffer, 0x00, sizeof(buffer));
    memcpy(buffer, s1, strlen(s1));
    strcat(buffer, s2, len);
    printf("%s \n", buffer);
    return;
} /* end printStr */

```

规划数据弹性

数据弹性是数据可供用户或应用程序使用的能力。可通过将 i5/OS 集群技术与切换式磁盘、跨站点镜像或逻辑复制技术配合使用来实现数据弹性。

对于 i5/OS 支持的数据弹性实现，有多种技术可供选择。将这些技术与 i5/OS 集群资源服务合并到一起后，就可以构建完整的高可用性解决方案了。这些技术可按以下方式分类：

i5/OS 独立磁盘池技术

这些技术都以 i5/OS 实现的独立磁盘池为基础。为实现使用独立磁盘池技术的高可用性，需要有弹性的所有数据应存储在独立磁盘池中。在许多情况下，这要求将数据迁移至独立磁盘池。此信息假定数据迁移已完成。

以下 i5/OS 支持的技术以独立磁盘池为基础：

- 切换式磁盘
- 地理镜像
- 城域镜像

- 全局镜像

逻辑复制技术

逻辑复制是一种基于日志的技术，使用逻辑复制时数据将实时复制至另一系统。逻辑复制技术将 i5/OS 集群资源服务和日志记录与 IBM 业务合作伙伴应用程序配合使用。这些解决方案要求高可用性业务合作伙伴应用程序配置并管理环境。本信息未提供这些 IBM 业务合作伙伴解决方案的特定需求。如果要实现高可用性的逻辑复制解决方案，请查阅有关应用程序的信息或与服务代表联系。

相关信息



IBM eServer iSeries Independent ASPs: A Guide to Moving Applications to IASPs

确定要变为弹性数据的数据

了解应将哪些类型的数据变为弹性数据。

确定要变为弹性数据的数据类似于为系统准备备份和恢复策略时确定要备份并保存的数据种类。您需要确定环境中的哪些数据是让业务持续正常运行的关键。

例如，如果运行基于 Web 的业务，那么下列数据是关键所在：

- 当天订单
- 库存
- 客户记录

一般来说，更改不太频繁的数据或不需要每天使用的信息可能不必成为弹性信息。

规划切换式磁盘

数据的单个副本保留在可切换硬件上，如扩展单元（塔式机柜）或逻辑分区环境中的 IOP。

主节点上发生中断时，对可切换硬件上的数据的访问会切换至指定的备份节点。此外，还可在跨站点镜像（XSM）环境中使用独立磁盘池。这允许独立磁盘池的镜像副本保留在地理上远离原始站点的系统上（可选），以提高可用性或出于保护目的。

如果计划使用驻留在可切换独立磁盘池或跨站点镜像上的可切换资源，那么需要仔细规划。

还应评估当前系统磁盘配置以确定是否需要额外的磁盘机。与任何系统磁盘配置类似，可供应用程序使用的磁盘机数会对其性能产生重大影响。对有限数目的磁盘机增加额外工作负载可能导致磁盘等待时间延长，并且最终导致对应用程序的响应时间延长。对于配置了独立磁盘池的系统中的临时存储器而言，这一点尤其重要。临时存储器中的所有内容都将写入 SYSBAS 磁盘池。如果应用程序使用的临时存储空间不多，那么 SYSBAS 磁盘池中数目不多的磁盘臂尚能应付。您还必须记住，操作系统和基本功能都使用 SYSBAS 磁盘池。

您需要先为专用服务工具（DST）设置适当的权限，才能使用 IBM Systems Director Navigator for i5/OS 来执行所有磁盘管理任务，如创建独立磁盘池。

相关任务

启用和访问磁盘机

1 切换式磁盘的硬件需求：

- 1 要使用切换式磁盘，必须使用特定的硬件。
- 1 要使用切换式磁盘，必须使用下列其中一种硬件：

- 一个或多个驻留在高速链路（HSL）环路上的扩展部件（机架/部件）。
- 共享总线中的一个或多个 IOP 或者分配给 I/O 池的 IOP。在 LPAR 环境中，可以在系统分区之间切换包含独立切换式磁盘的 IOP，而不必使用扩展部件。IOP 必须在由多个分区共享的总线上或者被分配到 I/O 池。IOP 上的所有 IOA 都将进行切换。

除了这些硬件需求以外，切换式磁盘还要求进行下列物理规划：

- 必须使用高速链路（HSL）线缆将扩展部件连接至集群中的系统。在 HSL 环路中，该扩展部件在物理上必须与备用系统或者该备用系统拥有的扩展部件相邻。虽然每个系统（集群节点）可以连接至多个 HSL 环路，但每个 HSL 环路最多只能包含两个系统。虽然每个环路段最多可以包含 3 个扩展部件，但每个 HSL 环路最多只能包含 4 个扩展部件。在包含两个系统的 HSL 环路中，存在两个由这两个系统分隔的段。一个环路段中的所有扩展部件必须包含在同一个设备集群资源组（CRG）中。
- 为了使扩展部件可切换，它在物理上必须离环路段中的拥有系统最远。注意：在尝试使一个扩展部件可切换时，如果有另一个扩展部件离拥有系统更远并且不可切换，那么将出错。
- 可切换扩展部件必须通过 SPCN 线缆连接到最初用作设备集群资源组（设备 CRG）主节点的系统部件。主节点可以是系统部件中的主逻辑分区或辅助逻辑分区。如果正在使用逻辑分区，那么所需扩展部件中的系统总线必须由集群所使用的分区拥有并专用。

切换式磁盘的软件需求：

如果计划使用切换式磁盘来获取 i5/OS 高可用性，应确保满足最低软件需求。

- 要使用此技术中新的和增强的功能及功能部件，建议在参与基于此技术的高可用性解决方案的每个系统或逻辑分区上安装最新发行版和最新版本的操作系统。

注：对于同一 HSL 环上的系统，请参阅高可用性 Web 站点以确保您具有 i5/OS 的兼容版本。

- 要执行实现独立磁盘池时所需的一些磁盘管理任务，需要下列其中一个图形界面。

- IBM Systems Director Navigator for i5/OS
- System i 导航器

- 需要安装 i5/OS 选项 41，即 HA 可切换资源。选项 41 使您能够在系统之间切换独立磁盘池。要在系统间切换独立磁盘池，这些系统必须是某个集群的成员，并且独立磁盘池必须与该集群中的某个设备集群资源组相关联。要使用 IBM System i High Availability Solutions Manager (iHASM) 许可程序中提供的高可用性管理界面，也需要选项 41。

相关信息

高可用性和集群

切换式磁盘的通信需求：

切换式磁盘在集群中的系统之间至少需要一个 TCP/IP 通信接口。

为了实现冗余，建议您在系统之间至少设置两个独立的接口。

规划跨站点镜像

跨站点镜像提供了一些 i5/OS 灾难恢复和高可用性技术：地理镜像、城域镜像和全局镜像。

跨站点镜像技术通过维护分散站点（相互距离通常较远）来实现灾难恢复。其中每个技术都有特定通信、硬件和软件需求。但是，在实现其中任一技术之前，都应规划站点。一个站点通常被视为生产站点或源站点。此站点包含镜像或复制至远程站点的生产数据。远程站点（有时称为备份或目标站点）包含生产数据的镜像副本。如果生产站点上发生站点范围的灾难，那么备份站点会使用镜像数据帮助您继续开展业务。配置跨站点镜像技术之前，应考虑有关站点规划的下列信息。

确定哪些站点将成为生产站点和备份站点

访问每个站点上适当位置的当前硬件和软件资源，以确定是否缺少跨站点镜像解决方案所需的组件。

确定生产站点与备份站点之间的距离

根据通信带宽和其他因素，站点之间的距离会影响所选镜像技术的性能和等待时间。某些跨站点镜像技术更适用远距离站点，而另一些跨站点镜像技术在用于远距离站点时性能可能会下降。

确保您有适当的权限使用 DST

您需要先为专用服务工具（DST）设置适当的权限，才能使用 IBM Systems Director Navigator for i5/OS 来执行任何磁盘管理任务。

相关任务

启用和访问磁盘机

规划地理镜像：

地理镜像是跨站点镜像的子功能。此技术在 i5/OS 环境中提供灾难恢复和高可用性。

地理镜像的硬件需求：

如果您计划通过地理镜像实现 i5/OS 高可用性，那么请确保符合最低硬件需求。

- 必须符合独立磁盘池的所有硬件需求。
- 至少需要两个 System i 系统，这两个系统在地理上可以相互分离。
- 在每个站点上，至少需要两组容量相同的磁盘。
- 应该使用进行地理镜像的独立磁盘池为作业配置独立的存储池。在负载极高的情况下，从主存储池中执行地理镜像会导致系统挂起。
- 地理镜像在磁盘池可用时执行。执行地理镜像时，不应更改时间（QTIME）系统值。
- 独立磁盘池的通信需求会影响吞吐量，因此尤其关键。

相关概念

第 50 页的『地理镜像的通信需求』

在实现使用地理镜像的 i5/OS 高可用性解决方案时，应该规划通信线路，以使地理镜像流量不会对系统性能产生负面影响。

地理镜像的软件需求：

如果您计划在 i5/OS 高可用性解决方案中使用地理镜像，那么需要下列软件。

- 要使用地理镜像的高级功能，必须安装 IBM System i High Availability Solutions Manager (iHASM) 许可程序 (5761-HAS)。
- 要使用此技术中新的和增强的功能及功能部件，建议在参与基于此技术的高可用性解决方案的每个系统或逻辑分区上安装最新发行版和最新版本的操作系统。

注：对于同一 HSL 环上的系统，请参阅高可用性 Web 站点以确保您具有 i5/OS 的兼容版本。

- 要执行实现独立磁盘池时所需的一些磁盘管理任务，需要下列其中一个图形界面。

- IBM Systems Director Navigator for i5/OS
- System i 导航器

- 需要安装 i5/OS 选项 41，即 HA 可切换资源。选项 41 使您能够在系统之间切换独立磁盘池。要在系统间切换独立磁盘池，这些系统必须是某个集群的成员，并且独立磁盘池必须与该集群中的某个设备集群资源组相关联。要使用 IBM System i High Availability Solutions Manager (iHASM) 许可程序中提供的高可用性管理界面，也需要选项 41。

相关信息

高可用性和集群

地理镜像的通信需求:

在实现使用地理镜像的 i5/OS 高可用性解决方案时，应该规划通信线路，以使地理镜像流量不会对系统性能产生负面影响。

建议您采取下列措施:

- 地理镜像会生成大量的通信流量。如果地理镜像与另一个应用程序（例如集群）共享同一个 IP 连接，那么地理镜像可能会暂挂，这将导致需要进行同步。同样，集群功能的响应速度可能不可接受，这将产生分区节点。地理镜像应该有自己的专用通信线路。如果地理镜像没有自己的专用通信线路，那么就会干扰其他使用同一通信线路的应用程序并影响用户网络的性能和吞吐量。这还包括对集群脉动信号监视产生负面影响，从而导致集群处于分区状态。因此，建议您让地理镜像和集群都使用专用的通信线路。地理镜像最多支持 4 条通信线路。

地理镜像通过多条线路分发更改，以便最大程度地提高性能。数据依次（从 1 到 4）通过每一条已配置的通信线路发送，不断反复。4 条通信线路能够实现最佳的性能，但使用两条线路时，可以实现相对不错的性能。

如果将两个节点之间的多条线路用于地理镜像，那么最好将那些线路分到不同的子网中，以便在这两个系统上平衡对那些线路的使用。

- 在配置中，如果多个应用程序或服务要求使用同一条通信线路，那么通过 i5/OS 的 TCP/IP 功能实现服务质量（QoS），可以缓解一些问题。i5/OS 服务质量（QoS）解决方案允许策略为网络中的 TCP/IP 应用程序申请网络优先级和带宽。

如果将两个节点之间的多条线路用于地理镜像，那么最好将那些线路分到不同的子网中，以便在这两个系统上平衡对那些线路的使用。

- 确保每个数据端口连接的吞吐量匹配。这意味着，在一对系统之间，所有连接的速度和连接类型应该相同。如果吞吐量不同，那么性能受速度最低的连接约束。

- 考虑为 TCP/IP 连接配置虚拟专用网，其优点如下所示:

- 通过对数据进行加密，保护数据传输的安全
- 通过提高发送冗余度，提高数据传输的可靠性

相关概念

第 49 页的『地理镜像的硬件需求』

如果您计划通过地理镜像实现 i5/OS 高可用性，那么请确保符合最低硬件需求。

相关参考

服务质量（QoS）

地理镜像的日志规划:

实现基于 i5/OS 地理镜像的高可用性时，应该规划日志管理。

日志管理功能防止事务由于系统异常关闭而丢失。如果对对象进行日志记录，那么系统将记录您对该对象所作的更改。无论实现哪种高可用性解决方案，日志记录都是防止数据在系统异常运行中断期间丢失的最佳措施。

相关信息

日志管理

地理镜像的备份规划:

| 在实现基于地理镜像的高可用性之前，您应该了解并规划此环境中的备份策略。

| 在配置任何高可用性解决方案之前，请对当前备份策略进行评估，并在必要时进行相应的更改。地理镜像不允许对独立磁盘池的镜像副本进行并发访问，这会对远程备份的执行产生影响。如果要从地理镜像副本备份到磁带，那么必须在生产系统上停止镜像并在启用跟踪的情况下拆离镜像副本。跟踪功能用于对生产系统上的更改进行跟踪，以便在镜像副本重新联机后进行同步。然后，必须将独立磁盘池的已拆离副本联机，执行备份过程，使独立磁盘池脱机并将其重新连接至原始生产主机。此过程只要求在生产副本与镜像副本之间进行部分数据再同步。

| 在执行备份以及进行同步期间，系统处于未受保护运行状态。并且，建议您在启用跟踪的情况下暂挂镜像，这将提高同步过程的速度。在发生持久传输中断（例如源系统与目标系统之间的所有通信路径长时间丢失）之后，也需要进行同步。您还可以使用冗余通信路径来帮助消除某些与通信故障相关联的风险。

| 建议您至少在第三个系统或逻辑分区中使用地理镜像功能，在这种情况下，可以将独立磁盘池的生产副本切换到同一站点上另一个能够维护地理镜像的系统。

| 相关概念

| 第 104 页的『场景：在地理镜像环境中执行备份』

| 本场景概述在使用地理镜像的 i5/OS 高可用性解决方案中执行远程备份时必须完成的任务。

| 第 73 页的『场景：切换式磁盘与地理镜像配合』

| 本场景描述一种 i5/OS 高可用性解决方案，此方案在三节点集群中将切换式磁盘与地理镜像配合使用。此解决方案提供了灾难恢复和高可用性功能。

| 地理镜像的性能规划:

| 实现地理镜像解决方案时，您需要了解并规划环境，以便最大程度地降低对性能的潜在影响。

| 有多种因素会影响地理镜像的性能。为了最大程度地提高地理镜像环境的性能，通常应该针对下列因素进行规划:

| CPU 注意事项

| 地理镜像会提高 CPU 负载，因此必须要有足够的额外 CPU 容量。您可能需要添加处理器或提高 CPU 容量。通常，用于运行地理镜像的分区不止需要一个部分处理器。在最低的 CPU 配置中，运行地理镜像时，您可能会注意到 CPU 开销为 5 - 20%。如果备份系统的处理器数比生产系统少，并且有许多写操作，那么 CPU 开销可能会相当大并影响性能。

| 机器池大小注意事项

| 为了提高地理镜像的性能，尤其是提高同步期间的性能，请至少将机器池大小增大以下公式给出的幅度:

| • 附加机器池存储量为: $300 \text{ MB} + .3\text{MB} \times \text{独立磁盘池中的磁盘 ARM 数}$ 。下列示例说明分别具有 90 个磁盘 ARM 和 180 个磁盘 ARM 的独立磁盘池所需的附加机器池存储量:

| - $300 + (.3 \times 90 \text{ ARM}) = 327 \text{ MB}$ 附加机器池存储器

| - $300 + (.3 \times 180 \text{ ARM}) = 354 \text{ MB}$ 附加机器池存储器

| 在集群资源组 (CRG) 中的所有节点上都需要附加的机器池存储器，以使目标节点在进行切换或故障转移时有足够的存储器。通常，独立磁盘池中的磁盘机越多，性能就应该越高，这是因为可以并行地执行许多操作。

| 为了防止性能调整程序功能减小机器池大小，应该执行下列其中一项操作:

| 1. 通过“使用共享存储池” (WRKSHRPOOL) 命令或“更改共享存储池” (CHGSHRPOOL) 命令，将机器池最小大小设置为计算得到的容量 (当前大小加上公式计算而得的地理镜像附加大小)。

- | 注：建议您将此选项与“使用共享存储池”（WRKSHRPOOL）选项配合使用。
- | 2. 将“自动调整内存池和活动级别”（QPFRADJ）系统值设置为零，这将禁止性能调整程序更改机器池大小。

| 磁盘机注意事项

| 磁盘机和 IOA 的性能会影响整体地理镜像性能。当镜像系统上的磁盘子系统速度较慢时，情况尤其如此。当地理镜像处于同步方式时，对生产副本执行的所有写操作受镜像副本的磁盘写操作制约。因此，低速目标磁盘子系统会影响源端的性能。为了最大程度地降低这种对性能的影响，可以通过异步方式运行地理镜像。通过以异步方式运行，可以缩短等待目标端磁盘子系统的时间。当更改后的内存页进入目标端的内存后，会将确认消息发回给源端。

| 系统磁盘池注意事项

| 与任何系统磁盘配置类似，可供应用程序使用的磁盘机数会对其性能产生重大影响。对有限数目的磁盘机增加额外工作负载可能导致磁盘等待时间延长，并且最终导致对应用程序的响应时间延长。对于配置了独立磁盘池的系统中的临时存储器而言，这一点尤其重要。临时存储器中的所有内容都将写入 SYSBAS 磁盘池。如果应用程序使用的临时存储空间不多，那么 SYSBAS 磁盘池中数目不多的磁盘臂尚能应付。您还必须记住，操作系统和基本功能都使用 SYSBAS 磁盘池。

| 网络配置注意事项

| 网络布线和配置会对地理镜像性能产生潜在影响。除了确保在不同子网中为每一组数据端口 IP 地址设置网络寻址方式以外，还应该以相同的方式设置网络布线和配置。

规划城域镜像：

i5/OS 高可用性支持城域镜像，城域镜像提供了高可用性和灾难恢复功能。为有效配置和管理使用此技术的高可用性解决方案，需要进行适当的规划。

相关信息

-  将“复制服务功能”与 DS6000 配合使用的准则和建议
-  将“复制服务功能”与 DS8000 配合使用的准则和建议

城域镜像的硬件需求：

要配置和管理使用城域镜像技术的 i5/OS 高可用性解决方案，应该确保符合最低硬件需求。

建议的最低硬件需求如下所示：

- 至少两个 System i 系统，它们在地理上通过至少一个与这两个系统都连接的 IBM System Storage™ DS6000™ 或 DS8000™ 外部存储器部件分隔。DS6000 和 DS8000 外部存储器部件在所有支持通过光纤通道连接外部存储器的 System i 型号上都受支持。
- 需要下列其中一种受支持的光纤通道适配器：
 - 2766 2 千兆位光纤通道磁盘控制器 PCI
 - 2787 2 千兆位光纤通道磁盘控制器 PCI-X
 - | – 5760 4 千兆位光纤通道磁盘控制器 PCI-X
- 要支持 DS6000 或 DS8000 上的外部装入源部件，需要新的 IOP：
 - 用于 SAN 装入源的功能部件 2847 PCI-X IOP

- 在进行任何配置前，应该调整系统存储器的磁盘空间大小。应该让源卷使用一组磁盘，让目标卷使用相同的一组磁盘机，并让每个一致性副本使用另一组磁盘机。

相关信息

 [iSeries™ and IBM TotalStorage: A Guide to Implementing External Disk on i5](#)

 [IBM System Storage DS6000 信息中心](#)

 [IBM System Storage DS8000 信息中心](#)

| 城域镜像的软件需求:

| 在配置使用城域镜像的 i5/OS 高可用性解决方案之前，请确保符合最低软件需求。

| 城域镜像具有下列最低软件需求:

- | • 高可用性解决方案中的每个 System i 模型必须正在运行 i5/OS V6R1 以便与 IBM System i High Availability Solutions Manager (iHASM) 许可程序 (5761-HAS) 配合使用。

| **注:** 对于前发行版，仍可使用 Lab Services 提供的 IBM Copy Services for System i 来处理 IBM System Storage 解决方案。如果正在多个平台上使用全局镜像，或者希望在多个 System i 分区上实现全局镜像，那么还可使用 IBM Copy Services for System i。

- | • 在每个参与使用城域镜像的高可用性解决方案的系统上安装 IBM System i High Availability Solutions Manager (iHASM) 许可程序 (5761-HAS)。

| • 需要安装 i5/OS 选项 41，即 HA 可切换资源。选项 41 使您能够在系统之间切换独立磁盘池。要在系统间切换独立磁盘池，这些系统必须是某个集群的成员，并且独立磁盘池必须与该集群中的某个设备集群资源组相关联。要使用 IBM System i High Availability Solutions Manager (iHASM) 许可程序中提供的高可用性管理界面，也需要选项 41。

| • 要控制存储器，iHASM 许可程序还需要存储器命令行界面 (DSCLI)。DSCLI 是所有 IBM System Storage 解决方案所必需的软件。要管理任何 IBM System Storage 解决方案，例如 Flashcopy、城域镜像或全局镜像，必须在每个参与使用了这些存储器解决方案的高可用性解决方案的系统或分区上安装 DSCLI。DSCLI 具有下列附加的软件需求:

- | – Java™ V1.4
- | – 在每个系统或分区上安装选项 35 (CCA 密码服务提供者)
- | • 确保已安装最新的 PTF。

| 相关信息

|  [iSeries™ and IBM TotalStorage: A Guide to Implementing External Disk on i5](#)

|  [IBM System Storage DS6000 信息中心](#)

|  [IBM System Storage DS8000 信息中心](#)

| 城域镜像的通信需求:

| 在配置使用城域镜像的 i5/OS 高可用性解决方案之前，请确保符合最低通信需求。

| 要使用城域镜像技术，您必须正在使用或计划使用存储区域网络 (SAN)。

1 SAN 是集中管理的专用安全信息基础结构，它允许在系统与存储系统之间进行任意互连。要使用 IBM System
1 Storage，例如 DS8000 或 DS6000 外部存储器部件，SAN 连接是必需的。

1 以下是使用城域镜像的 i5/OS 高可用性解决方案的最低通信需求：

- 1 • 需要下列其中一种受支持的光纤通道适配器：
 - 1 – 2766 2 千兆位光纤通道磁盘控制器 PCI
 - 1 – 2787 2 千兆位光纤通道磁盘控制器 PCI-X
 - 1 – 5760 4 千兆位光纤通道磁盘控制器 PCI-X
- 1 • System i 产品支持各种 SAN 交换机和导向器。请参阅存储区域网络 (SAN) 的 Web 站点，以获取受支持
1 的交换机和导向器的完整列表。
- 1 • 此外，强烈建议您利用多路径 I/O，以提高整体弹性和性能。多路径 I/O 允许对存储器中相同的逻辑磁盘机
1 配置多个光纤通道设备。正确地进行配置后，这允许单一设备、I/O 机柜或者 HSL 环路在发生故障时不会丢
1 失与磁盘机的连接。多路径 I/O 还将工作负载分散在所有可用连接（路径）上，从而提高性能。多路径磁盘
1 机的每个连接都独立地工作。多个连接使磁盘存储器在单一路径发生故障时保持可用状态，从而提高弹
1 性。

1 相关参考



1 存储区域网络 (SAN) Web 站点

城域镜像的日志规划:

日志记录对于缩短所有高可用性解决方案的恢复时间而言非常重要。对于基于 IBM System Storage 的技术，
例如城域镜像，由于数据镜像在 System i 存储器外部发生，因此通过进行日志记录强制对外部存储器部件执行
写操作十分关键。

日志管理功能防止事务由于系统异常关闭而丢失。如果对对象进行日志记录，那么系统将记录您对该对象所作
的更改。无论实现哪种高可用性解决方案，日志记录都是防止数据在系统异常运行中断期间丢失的最佳措施。

相关信息

日志管理

城域镜像的备份规划:

对于城域镜像而言，可以使用 FlashCopy® 对 IBM System Storage 外部存储器部件中存储的数据创建副本。

FlashCopy 操作使您能够创建时间点副本。FlashCopy 操作一旦处理完成，源卷和目标卷都可供应用程序使用。
FlashCopy 可以与其他 IBM System Storage 技术（例如城域镜像和全局镜像）配合使用，以便在远程站点上创
建一致的时间点数据副本，接着，可以使用标准备份过程对该数据副本进行备份。在实现 FlashCopy 之前，应
该完成下列操作：

- 确定 FlashCopy 关系的源卷和目标卷。为了提高性能，应该选择不同等级的 FlashCopy 目标卷。
- 了解 FlashCopy 数据一致性注意事项。在某些环境中，数据存储在系统内存高速缓存中，并且在以后某个时
间被写入磁盘。为了避免这些类型的重新启动操作，请确保已将所有与 FlashCopy 源卷相关的数据写入磁盘，
然后再执行 FlashCopy 操作。
- 可以使用现有的城域镜像源卷作为 FlashCopy 目标卷。这允许您使用 FlashCopy 对的目标卷创建时间点副本，
然后将该数据以镜像方式复制至远程位置的源城域镜像卷。

城域镜像的性能规划:

在配置城域镜像之前，您应该了解下列性能注意事项。

在使用城域镜像前，请考虑下列需求和准则：

- 城域镜像关系中的源卷和目标卷的存储器类型必须相同。
- 源逻辑卷和目标逻辑卷的大小相同或者目标逻辑卷更大。
- 对于城域镜像环境，通过不将所有更新定向到单一目标存储器部件上的一小组公共卷来分担工作负载。对目标站点存储器部件的性能影响会对源站点的性能产生负面影响。
- 与任何系统磁盘配置类似，可供应用程序使用的磁盘机数会对其性能产生重大影响。对有限数目的磁盘机增加额外工作负载可能导致磁盘等待时间延长，并且最终导致对应用程序的响应时间延长。对于配置了独立磁盘池的系统中的临时存储器而言，这一点尤其重要。临时存储器中的所有内容都将写入 SYSBAS 磁盘池。如果应用程序使用的临时存储空间不多，那么 SYSBAS 磁盘池中数目不多的磁盘臂尚能应付。您还必须记住，操作系统和基本功能都使用 SYSBAS 磁盘池。

相关信息



将“复制服务功能”与 DS6000 配合使用的准则和建议



将“复制服务功能”与 DS8000 配合使用的准则和建议

| 规划全局镜像：

| i5/OS 高可用性支持全局镜像，全局镜像在使用外部存储器解决方案的环境中提供高可用性和灾难恢复功能。为有效配置和管理使用此技术的高可用性，需要进行适当的规划。

| IBM System Storage 全局镜像技术要求所有用户共享一个全局镜像连接。在给定的 System Storage 服务器上，i5/OS 高可用性全局镜像只允许一个 System i™ 分区在全局镜像会话中处于活动状态。其他平台上的任何 System i 分区或服务器都不能同时使用全局镜像。向全局镜像会话添加多个用户会导致不可预测的结果。

| 如果正在多个平台上使用全局镜像，或者希望在多个 System i 分区上实现全局镜像，那么可使用 IBM Copy Services for System i。本产品由 Lab Services 提供。

| 相关信息



| 将“复制服务功能”与 DS6000 配合使用的准则和建议



| 将“复制服务功能”与 DS8000 配合使用的准则和建议

| 全局镜像的硬件需求：

| 要配置和管理使用全局镜像技术的 i5/OS 高可用性解决方案，应该确保符合最低硬件需求。

| 对于全局镜像，应该符合下列最低硬件需求：

- | • 至少两个 System i 系统，它们在地理上通过至少一个与这两个系统都连接的 IBM System Storage DS6000 或 DS8000 外部存储器部件分隔。DS6000 和 DS8000 外部存储器部件在所有支持通过光纤通道连接外部存储器的 System i 型号上都受支持。
- | • 需要下列其中一种受支持的光纤通道适配器：
 - | – 2766 2 千兆位光纤通道磁盘控制器 PCI
 - | – 2787 2 千兆位光纤通道磁盘控制器 PCI-X
 - | – 5760 4 千兆位光纤通道磁盘控制器 PCI-X
- | • 要支持 DS6000 或 DS8000 上的外部装入源部件，需要新的 IOP：
 - | – 用于 SAN 装入源的功能部件 2847 PCI-X IOP

- 在进行任何配置前，应该调整系统存储器的磁盘空间大小。应该让源卷使用一组磁盘，让目标卷使用相同的一组磁盘机，并让每个一致性副本使用另一组磁盘机。

相关信息



[iSeries™ and IBM TotalStorage: A Guide to Implementing External Disk on i5](#)



[IBM System Storage DS6000 信息中心](#)



[IBM System Storage DS8000 信息中心](#)

全局镜像的软件需求:

在配置使用全局镜像的 i5/OS 高可用性解决方案之前，请确保符合最低软件需求。

全局镜像具有下列最低软件需求:

- 高可用性解决方案中的每个 System i 模型必须正在运行 i5/OS V6R1 以便与 IBM System i High Availability Solutions Manager (iHASM) 许可程序 (5761-HAS) 配合使用。

注: 对于前发行版，仍可使用 Lab Services 提供的 IBM Copy Services for System i 来处理 IBM System Storage 解决方案。如果正在多个平台上使用全局镜像，或者希望在多个 System i 分区上实现全局镜像，那么还可使用 IBM Copy Services for System i。

- 在使用全局镜像的高可用性解决方案中包含的每个系统上安装 IBM System i High Availability Solutions Manager (iHASM) 许可产品 (5761-HAS)。
- 要控制存储器，iHASM 许可程序还需要存储器命令行界面 (DSCLI)。DSCLI 是所有 IBM System Storage 解决方案所必需的软件。要管理任何 IBM System Storage 解决方案，例如 Flashcopy、城域镜像或全局镜像，必须在每个参与使用了这些存储器解决方案的高可用性解决方案的系统或分区上安装 DSCLI。DSCLI 具有下列附加的软件需求:
 - Java V1.4
 - 在每个系统或分区上安装选项 35 (CCA 密码服务提供者)
- 确保已安装最新的 PTF。

相关信息



[iSeries™ and IBM TotalStorage: A Guide to Implementing External Disk on i5](#)



[IBM System Storage DS6000 信息中心](#)



[IBM System Storage DS8000 信息中心](#)

全局镜像的通信需求:

在配置使用全局镜像的 i5/OS 高可用性解决方案之前，应该确保符合最低通信需求。

要使用全局镜像技术，您必须正在使用或计划使用存储区域网络 (SAN)。

SAN 是集中管理的专用安全信息基础结构，它允许在系统与存储系统之间进行任意互连。要使用 IBM System Storage，例如 DS8000 或 DS6000 外部存储器部件，SAN 连接是必需的。

以下是使用全局镜像的 i5/OS 高可用性解决方案的最低通信需求:

- 需要下列其中一种受支持的光纤通道适配器:

- | - 2766 2 千兆位光纤通道磁盘控制器 PCI
- | - 2787 2 千兆位光纤通道磁盘控制器 PCI-X
- | - 5760 4 千兆位光纤通道磁盘控制器 PCI-X
- | • System i 产品支持各种 SAN 交换机和导向器。请参阅存储区域网络 (SAN) 的 Web 站点, 以获取受支持的交换机和导向器的完整列表。
- | • 此外, 强烈建议您利用多路径 I/O, 以提高整体弹性和性能。多路径 I/O 允许对存储器中相同的逻辑磁盘机配置多个光纤通道设备。正确地进行配置后, 这允许单一设备、I/O 机柜或者 HSL 环路在发生故障时不会丢失与磁盘机的连接。多路径 I/O 还将工作负载分散在所有可用连接 (路径) 上, 从而提高性能。多路径磁盘机的每个连接都独立地工作。多个连接使磁盘存储器在单一路径发生故障时保持可用状态, 从而提高弹性。

| 相关参考



| 存储区域网络 (SAN) Web 站点

| 全局镜像的日志规划:

| 日志记录对于缩短所有高可用性解决方案的恢复时间而言非常重要。对于基于 IBM System Storage 的技术, 例如全局镜像, 日志记录功能强制对外部存储器部件执行写操作。由于数据镜像在 System i 存储器外部发生, 所以必须这样做。

| 日志管理功能防止事务由于系统异常关闭而丢失。如果对对象进行日志记录, 那么系统将记录您对该对象所作的更改。无论实现哪种高可用性解决方案, 日志记录都是防止数据在系统异常运行中断期间丢失的最佳措施。

| 相关信息

| 日志管理

| 全局镜像的备份规划:

| 在高可用性解决方案中使用全局镜像技术时, 可以使用 FlashCopy 来创建数据的时间点副本。

| FlashCopy 操作使您能够创建时间点副本。FlashCopy 操作一旦处理完成, 源卷和目标卷都可供应用程序使用。FlashCopy 可以与其他 IBM System Storage 技术 (例如城域镜像和全局镜像) 配合使用, 以便在远程站点上创建一致的时间点数据副本, 接着, 可以使用标准备份过程对该数据副本进行备份。在实现 FlashCopy 之前, 应该完成下列操作:

- | • 确定 FlashCopy 关系的源卷和目标卷。为了提高性能, 应该选择不同等级的 FlashCopy 目标卷。
- | • 了解 FlashCopy 数据一致性注意事项。在某些环境中, 数据存储在系统内存高速缓存中, 并且在以后某个时间被写入磁盘。为了避免这些类型的重新启动操作, 请确保已将所有与 FlashCopy 源卷相关的数据写入磁盘, 然后再执行 FlashCopy 操作。

| 全局镜像的性能规划:

| 在配置全局镜像之前, 您应该了解下列性能注意事项。

| 在使用全局镜像前, 请考虑下列性能准则:

- | • 城域镜像关系中的源卷和目标卷的存储器类型必须相同。
- | • 城域镜像关系中的源卷和目标卷的存储器类型必须相同。
- | • 与任何系统磁盘配置类似, 可供应用程序使用的磁盘机会对其性能产生重大影响。对有限数目的磁盘机增加额外工作负载可能导致磁盘等待时间延长, 并且最终导致对应用程序的响应时间延长。对于配置了独立磁盘池的系统中的临时存储器而言, 这一点尤其重要。临时存储器中的所有内容都将写入 SYSBAS 磁盘

池。如果应用程序使用的临时存储空间不多，那么 SYSBAS 磁盘池中数目不多的磁盘臂尚能应付。您还必须记住，操作系统和基本功能都使用 SYSBAS 磁盘池。

相关信息



将“复制服务功能”与 DS6000 配合使用的准则和建议



将“复制服务功能”与 DS8000 配合使用的准则和建议

规划逻辑复制

逻辑复制用于维护数据的多个副本。数据会从集群中的主节点复制至恢复域中的指定备份节点。主节点上发生中断时，因为指定的备份节点会作为主访问点接管，所以数据仍然可用。

逻辑复制会实时复制某些对象。这是将对象从集群中的一个节点复制至该集群中的一个或多个其他节点的过程。逻辑复制会使这些对象在系统上保持完全相同。如果更改集群中一个节点上的对象，那么此更改会复制至该集群中的其他节点。

必须决定要用于逻辑复制的软件技术。下列解决方案可用于实现集群中的逻辑复制：

- **IBM 业务合作伙伴产品**

来自已认可集群 IBM 业务合作伙伴的数据复制软件允许您在多个节点间复制对象。

- **以定制方式编写的复制应用程序**

IBM 日志管理允许您记录系统上的对象的活动。可编写应用程序来利用日志管理以实现逻辑复制。

相关信息

日志管理

确定要用于逻辑复制的系统：

确定要用于逻辑复制的系统时，有一些关键注意事项。

关键注意事项包括：

- 性能处理能力
- 磁盘容量
- 关键数据
- 灾难避免

如果系统发生故障转移，您需要知道在主系统和备份系统运行的数据和应用程序。您需要将关键数据放在处理工作负载能力最强的系统上以防发生故障转移。您不希望出现磁盘空间不足的情况。如果主系统空间不足并且发生故障转移，那么备份系统也很有可能因为磁盘空间不足而发生故障转移。为确保出现自然灾害（洪水、龙卷风或飓风）时数据中心不会完全破坏，应将复制的系统放在远程位置。

集群中间件 IBM 业务合作伙伴和可用的集群产品：

除了 IBM 管理解决方案以外，还可从高可用性业务合作伙伴处购买使用逻辑复制技术的集群中间件产品。

IBM 集群中间件 IBM 业务合作伙伴提供了具有专用复制和集群管理功能的软件解决方案。大多数业务合作伙伴解决方案以逻辑复制为基础。逻辑复制会实时复制对象和记录级别更改。这是将对象从集群中的一个节点复制至该集群中的一个或多个其他节点的过程。复制会使这些对象在系统上保持完全相同。如果更改集群中一个节点上的对象，那么此更改会复制至该集群中的其他节点。

逻辑复制的日志规划:

如果使用逻辑复制，那么应该进行日志记录，以强制将数据的生产副本内容写至该数据的备份副本。

日志管理功能防止事务由于系统异常关闭而丢失。如果对对象进行日志记录，那么系统将记录您对该对象所作的更改。无论实现哪种高可用性解决方案，日志记录都是防止数据在系统异常运行中断期间丢失的最佳措施。

在逻辑复制环境中，日志记录是解决方案的基础，并且是实现基于此技术的解决方案的要求。对于逻辑复制而言，根据所复制对象的大小不同，可能无法以实时方式复制至备份系统。例如，程序对已进行日志记录的文件中的记录进行更新。在执行此操作期间，它还更新未进行日志记录的对象，例如用户空间。如果将用户空间完全复制至备份系统，那么备份副本将完全一致。但在实际上，如果主系统发生故障并且尚未完全复制用户空间对象，那么需要执行手工恢复过程，以使用户空间状态与上一个完全复制了数据的有效操作匹配。

相关信息

日志管理

逻辑复制的备份规划:

如果您使用由第三方应用程序提供的逻辑复制技术，那么应该规划此环境中的备份操作。

逻辑复制将对象（例如生产副本中的文件或程序）的更改复制至备份副本。此复制以近乎实时的方式进行。通常，如果已对该对象（例如文件）进行日志记录，那么将在记录级进行复制。此技术的关键优点是，可以对备份副本进行实时访问以进行备份。可以对数据的备份副本执行远程备份，而不会破坏该数据的生产副本。

逻辑复制的性能规划:

如果您正在高可用性解决方案中使用逻辑复制技术，那么应该了解此技术对此解决方案性能的潜在影响。

对于逻辑复制而言，对性能的潜在影响归因于复制过程的等待时间。此时间是指从源系统上进行更改直到那些更改在备份系统上变为可用的延迟时间量。通过进行同步远程日志记录，可以大幅缩短此时间。无论使用哪种传输机制，您都必须充分预测传输量并正确规划通信线路和速度，以帮助确保环境在复制量达到顶峰时能够从容应对。在数据量很大的环境中，即使正确地规划传输设施，延迟也仍然是一个问题。

规划环境弹性

环境弹性可确保对象和属性在高可用性环境中定义的资源间保持弹性。您需要标出哪些资源需要一致的环境才能正常工作，并创建集群管理域来确保这些资源属性在高可用性解决方案中保持一致。

规划集群管理域

集群管理域要求进行规划以管理在集群管理域中各个节点间同步的资源。为了确保应用程序在高可用环境中的任何系统上以一致方式运行，需要标出影响应用程序行为的所有资源、运行该应用程序的集群节点以及应用程序数据可能的所在位置。

集群管理员可创建集群管理域，并添加在节点间同步的受监视资源。i5/OS 集群提供了可由集群管理域同步并由受监视资源项（MRE）表示的系统资源列表。

设计集群管理域时，应考虑下列问题:

集群管理域将包括哪些节点？

应考虑集群中将由集群管理域管理的节点。这些集群节点表示可运行应用程序、存储应用程序数据以及需要一致运营环境的系统。节点不能同时存在于多个集群管理域中。例如，如果一个集群中有四个

节点（节点 A、节点 B、节点 C 和节点 D），那么节点 A 和 B 可能在一个集群管理域中，而节点 C 和 D 可能在另一个管理域中。但是，不能将节点 B 和 C 放在第三个集群管理域中而同时将它们保留在原始集群管理域中。

集群管理域的命名约定是什么？

根据集群环境的复杂程度和大小，您可能需要为对等 CRG 和集群管理域建立标准命令约定。因为对等 CRG 是在您创建集群管理域时创建的，所以您需要将其他对等 CRG 与表示集群管理域的对等 CRG 区分开来。例如，表示集群管理域的对等 CRG 可命名为 *ADMDMN1*、*ADMDMN2* 等等，而其他同步 CRG 可命名为 *PEERI*。还可使用 列示集群资源组信息（`QcstListClusterResourceGroupIn`）API 来确定是否将对等 CRG 用作集群管理域。表示集群管理域的对等 CRG 可通过应用程序标识（`QIBM.AdminDomain`）来标识。

规划受监视资源项（MRE）

受监视资源是可在集群管理域中定义的 i5/OS 对象。这些资源需要在高可用性环境中的系统间保持一致，否则中断期间应用程序可能无法按预期方式执行。应规划需要监视环境中的哪些受支持资源。

需要确定哪些系统资源需要同步。可选择其中每个资源的属性以定制要同步的对象。在多个节点上运行的应用程序可能需要特定环境变量才能正常运行。此外，跨越多个节点的数据还有可能要求访问特定用户概要文件。在确定需要由集群管理域管理的资源之前，应了解应用程序和数据的操作需求。

规划集群

在实现高可用性解决方案之前，必须确保您已满足集群的所有先决条件。

集群的硬件需求

为实现高可用性解决方案，需要规划并配置集群。集群会将系统和资源组合到高可用性环境中。

以下是集群的最低硬件需求：

- 您至少需要两个 System i 模型或逻辑分区。一个集群最多包含 128 个系统。任何能够运行 i5/OS V4R4M0 或更高版本的 System i 模型允许使用集群。
- 建议使用外部不间断电源或等效设备，以避免突然掉电从而导致集群分区。
- 集群使用因特网协议（IP）多点广播功能。多点广播不能很好地映射至所有类型的物理介质。
- 如果计划使用需要独立磁盘池的数据弹性技术，那么还需要规划特定于所选数据弹性技术的硬件。还可使用不同的磁盘保护方法来避免受保护磁盘发生故障时出现故障转移。

相关概念

第 46 页的『规划数据弹性』

数据弹性是数据可供用户或应用程序使用的能力。可通过将 i5/OS 集群技术与切换式磁盘、跨站点镜像或逻辑复制技术配合使用来实现数据弹性。

相关参考

第 65 页的『集群的规划核对表』

请完成集群配置核对表，以确保在开始配置集群前正确地准备环境。

相关信息

不间断电源

IP 多点广播

磁盘保护

集群的软件需求

为使用集群，必须有正常的软件和许可证。

1. 已安装 i5/OS V6R1。
2. 已安装 TCP/IP Connectivity Utilities 功能。
3. 如果计划使用数据弹性技术（如切换式磁盘或跨站点镜像），那么还有一些额外需求。
4. 如果计划使用下列界面，那么需要选项 41（高可用性可切换资源）：

- “System i 导航器集群管理”界面

注：有关使用“System i 导航器集群管理”界面的信息，请参阅 i5/OS V5R4 信息中心中的集群。

- IBM System i High Availability Solutions Manager (iHASM) 许可程序。此许可程序提供了下列界面，它们都需要选项 41：
 - “高可用性解决方案管理器”图形界面
 - “集群资源服务”图形界面
 - IBM System i High Availability Solutions Manager (iHASM) 命令
 - IBM System i High Availability Solutions Manager (iHASM) API

注：

5. 还可使用 IBM 业务合作伙伴产品或通过使用集群 API 编写自己的高可用性管理应用程序。

相关概念

第 47 页的『规划切换式磁盘』

数据的单个副本保留在可切换硬件上，如扩展单元（塔式机柜）或逻辑分区环境中的 IOP。

第 48 页的『规划跨站点镜像』

跨站点镜像提供了一些 i5/OS 灾难恢复和高可用性技术：地理镜像、城域镜像和全局镜像。

第 46 页的『规划数据弹性』

数据弹性是数据可供用户或应用程序使用的能力。可通过将 i5/OS 集群技术与切换式磁盘、跨站点镜像或逻辑复制技术配合使用来实现数据弹性。

相关参考

第 65 页的『集群的规划核对表』

请完成集群配置核对表，以确保在开始配置集群前正确地准备环境。

相关信息

集群 API

集群的通信需求

可在集群环境中使用任何支持因特网协议（IP）的类型的通信介质。

集群资源服务使用 TCP/IP 和 UDP/IP 协议在节点间通信。局域网（LAN）、广域网（WAN）、OptiConnect 系统区域网络（SAN）或这些连接设备的任意组合都是受支持的。您应根据下列因素进行选择：

- 交易量
- 响应时间需求
- 节点之间的距离
- 成本注意事项

可在确定用于连接资源的主要位置和备份位置的连接介质时使用相同注意事项。规划集群时，建议在远程位置中指定一个或多个备份节点以从站点崩溃恢复。

为避免可能因为容量不足而导致的性能问题，您需要评估用于处理节点间发送的信息量的通信介质。可选择您的首选物理介质，如令牌环、以太网、异步传输方式（ATM）、SPD OptiConnect、高速链接（HSL）OptiConnect 或虚拟 OptiConnect（逻辑分区之间的高速内部连接）。

HSL OptiConnect 是由 OptiConnect for i5/OS 软件（i5/OS Option 23 - i5/OS OptiConnect）提供的技术。它可以用来构建高可用性解决方案。HSL OptiConnect 是一个系统区域网络，可通过使用高速链接（HSL）环技术在集群节点间提供高速点到点连接。HSL OptiConnect 只需要标准 HSL 电缆而无需其他任何硬件。

对于可切换硬件（又称为弹性设备 CRG），环境中需要切换式磁盘。在逻辑分区环境中，它是由多个逻辑分区共享的总线上的一组磁盘机，或者是连接至已分配给 I/O 池的输入/输出处理器的一组磁盘机。对于多系统环境，它是同时在恢复域中包含系统的 HSL 环上正确配置的一个或多个可切换扩展单元。还可在 LPAR 环境中使用可切换扩展单元。

注：如果要使用的 LAN 适配器仅使用 TCP/IP 而不使用系统网络体系结构（SNA）或 IPX，可通过处理行描述（WRKLIND）命令对特定行描述指定“仅对 TCP(*YES) 启用”来提高 OS/400® V4R5M0 系统上的适配器性能。“仅对 TCP(*YES) 启用”是在 OS/400 V5R1M0 及更高发行版中自动设置的。

相关概念

第 47 页的『规划切换式磁盘』

数据的单个副本保留在可切换硬件上，如扩展单元（塔式机柜）或逻辑分区环境中的 IOP。

相关参考

第 65 页的『集群的规划核对表』

请完成集群配置核对表，以确保在开始配置集群前正确地准备环境。

为集群提供网络：

在正常操作期间，至少需要基本集群通信流量。但是，强烈建议为集群中的每个节点配置冗余通信路径。

冗余通信路径意味着在集群中的两个节点间配置了两条线路。如果第一条通信路径出现故障，那么第二条通信路径可以接管以便让节点间的通信继续运行，从而尽量避免将集群的一个或多个节点放到一个集群分区中的情况。配置这些路径时要考虑的一点是，如果两条通信线路进入系统上的同一适配器，那么此适配器发生故障时两条线路仍然存在风险。但需要注意的是，集群分区并非总是可以避免的。如果系统遇到掉电或者发生硬件故障，那么会对集群进行分区。通过配置两条线路，可将一条线路用于集群流量，而将另一条线路用于正常流量，并在专门用于集群的线路发生故障时用作备份线路。通过在集群中的所有节点间配置冗余通信路径，可最大程度地避免与网络有关的典型集群分区。

技巧：集群通信：

在设置通信路径时，请考虑下列技巧。

- 确保通信线路有足够的带宽，以便处理非集群活动和集群脉动信号传递功能并持续监视不断增加的活动。
- 为了最大程度地提高可靠性，请不要配置单一的通信路径来链接一个或多个节点。
- 不要使负责确保节点通信保持畅通的线路负担过重。
- 尽可能消除单一故障点，例如将两条通信线路连接到单一适配器、同一个输入/输出处理器（IOP）或同一个扩展部件。
- 如果要通过通信线路传输极为大量的数据，那么可以考虑让不同的网络完成数据复制和脉动信号监视任务。
- 集群通信基础结构在集群中的节点之间发送集群管理信息时，使用的首选协议是用户数据报协议（UDP）多点广播协议。如果物理介质支持多点广播功能，那么集群通信将使用 UDP 多点广播功能将管理消息从给

定节点发送至所有支持同一子网地址的本地集群节点。发送至远程网络中节点的消息始终通过 UDP 点到点功能发送。集群通信并不依赖于多点广播消息的路由功能。

- 支持集群管理消息传递功能的多点广播流量在本质上具有波动性。根据给定 LAN 中支持公共子网地址的节点数以及集群管理员选择的集群管理结构的复杂性，与集群相关的多点广播信息包很容易超出每秒 40 个信息包这一速率。这种性质的波动会对较旧的联网设备产生负面影响。一个示例是，在 LAN 中用作简单网络管理协议（SNMP）代理的设备上，由于需要对每个 UDP 多点广播信息包进行评估，因此会发生拥塞问题。某些早期的联网设备没有足够的带宽来支持此类流量。您需要确保您或网络管理员已检查网络处理 UDP 多点广播流量的能力，以确保集群功能不会对网络性能产生负面影响。

集群的性能规划:

由于通信环境可能存在相当大的差别，因此，您可以调整影响集群通信的变量，以使其更好地与环境匹配。

通常，大部分公共环境应该能够接受缺省值。如果这些缺省值并不适合于您的特定环境，那么您可以调整集群通信，以使其更好地与环境匹配。您可以进行基本调整，也可以进行高级调整。

基本调整

基本调整允许将调整参数设置为一组预定义的值，这些值根据高、低和正常超时值和消息传递时间间隔值确定。如果选择正常级别，那么集群通信性能参数和配置参数将使用缺省值。选择低级将导致集群增大脉动信号传递时间间隔以及各种消息超时值。脉动信号越少以及超时值越大，集群对通信故障就越不灵敏。选择高级将导致集群减小脉动信号传递时间间隔以及各种消息超时值。脉动信号频率越高以及超时值越小，集群对通信故障就越灵敏。

高级调整

对于高级调整，可以使用预定义的值范围来调整各个参数。这允许您更细致地进行调整，以便适应通信环境中的任何特殊情况。如果需要高级调整，那么建议您向 IBM 支持人员或同等人员寻求帮助。这是因为，不正确地设置各个参数很容易导致性能下降。

可调整的集群通信参数:

“更改集群资源服务”（QcstChgClusterResourceServices）API 允许对某些集群技术服务以及集群通信性能和配置参数进行调整，以使这些参数更适合于许多采用了集群功能的独特应用程序和联网环境。

“更改集群”（CHGCLU）命令用于进行基本调整，而 QcstChgClusterResourceServices API 用于进行基本调整和高级调整。

QcstChgClusterResourceServices API 和“更改集群配置”（CHGCLUCFG）命令可用于调整集群性能和配置。此 API 和命令提供了基本的调整支持，即，集群根据一组预定义的值进行调整，而这些值根据高、低和正常超时值和消息传递时间间隔值确定。如果要进行高级调整（通常在 IBM 支持人员的帮助下进行），那么可以通过对预定义的值范围使用 API 来调整各个参数。不恰当地更改各个参数很容易导致集群性能下降。

何时以及如何调整集群参数

CHGCLU 命令和 QcstChgClusterResourceServices API 使您能够轻松快捷地设置集群性能参数和配置参数，而不要求您了解细节。这种基本调整主要影响脉动信号灵敏度和集群消息超时值。用于基本调整支持的有效值如下所示:

1（高超时值/低频率脉动信号）

对集群通信进行调整，以降低脉动信号频率并增大各种消息超时值。脉动信号越少以及超时值越大，集群对通信故障的响应速度就越慢（灵敏度较低）。

2 (缺省值)

集群通信性能参数和配置参数使用正常的缺省值。此设置可用于将所有参数还原为原始缺省值。

3 (低超时值/高频率脉动信号)

对集群通信进行调整，以减小脉动信号时间间隔并减小各种消息超时值。脉动信号频率越高以及超时值越小，集群对通信故障的响应速度就越快（灵敏度较高）。

下表显示了导致节点分区的脉动信号故障的示例响应时间：

注：时间以“分钟:秒”格式指定。

	1 (低灵敏度)			2 (缺省)			3 (高灵敏度)		
	检测脉动信号问题	分析	总计	检测脉动信号问题	分析	总计	检测脉动信号问题	分析	总计
单一子网	00:24	01:02	01:26	00:12	00:30	00:42	00:04	00:14	00:18
多个子网	00:24	08:30	08:54	00:12	04:14	04:26	00:04	02:02	02:06

根据典型网络负载和所使用具体物理介质的不同，集群管理员可以选择调整脉动信号灵敏度和消息超时级别。例如，对于可靠性较高的高速传输，例如集群中所有系统都在公共 OptiConnect 总线上的 OptiConnect 传输，我们可能希望建立更灵敏的环境，以确保快速进行检测，从而更快地进行故障转移。因此，选择选项 3。如果我们使用的是高负载的 10 Mbs 以太网总线，并且缺省设置偶尔会因为网络负载高峰而导致分区，那么可以选择选项 1 以降低集群对高峰负载的灵敏度。

“更改集群资源服务”API 还允许在网络环境需求体现了独特情况时调整各个特定参数。例如，请再次考虑集群中所有节点都在公共 OptiConnect 总线上的情况。通过将消息段大小参数设置为最大值（32,500 字节）而不是采用缺省值（1,464 字节）以便更好地与 OptiConnect 最大传输单元（MTU）大小匹配，可以大幅提高集群消息的性能。这将降低大型消息的分段和重新装配开销。当然，优点取决于集群应用程序以及那些应用程序引起的对集群消息传递功能的使用。API 文档对其他参数进行了定义，那些参数可以用于调整集群消息传递性能或者更改集群对分区的灵敏度。

相关参考

QcstChgClusterResourceServices API

相关信息

更改集群（CHGCLU）命令

更改集群资源服务设置：

影响消息超时和重试的缺省值针对最典型的安装版本设置。但是，您可以更改这些值，以使其更紧密地与通信环境匹配。

可以通过下列其中一种方法来调整这些值：

- 设置与环境相匹配的常规性能级别。
- 设置特定消息调整参数的值，以便更具体地进行调整。

在第一种方法中，根据三种通信级别中的一种调整消息流量。缺省级别为“正常”，“脉动信号监视”一节对此级别作了详细描述。

第二种方法通常只应该在专家的建议下采用。

“更改集群资源服务”（QcstChgClusterResourceServices）API 描述了这两种方法的详细信息。

相关参考

QcstChgClusterResourceServices API

相关信息

脉动信号监视

规划多发行版集群:

如果要创建的集群包括的节点在多个集群版本上，那么创建集群时需要执行特定步骤。

缺省情况下，当前集群版本设置为添加至集群的第一个节点的潜在集群版本。如果此节点处于集群中的最低版本级别，那么此方法适用。但是，如果此节点处于较高版本级别，那么随后不能添加版本级别较低的节点。另一方法是在您创建集群时，使用目标集群版本值将当前集群版本设置为比添加至集群的第一个节点的可能集群版本低一级。

注：如果要使用 IBM System i High Availability Solutions Manager (iHASM) 许可程序，那么 V6R1 在集群内的所有系统上是必需的。

例如，考虑创建包含两个节点的集群的情况。此集群的节点如下所示：

节点标识	发行版	潜在集群版本
节点 A	V5R4	5
节点 B	V6R1	6

如果要从节点 B 创建集群，那么必须注意指示此集群是混合发行版集群。目标集群版本必须设置为指示集群节点将在比请求节点的可能节点版本低一级的版本上通信。

集群的性能规划

对集群进行更改时，可能会影响管理集群所需的开销。

集群功能所需的资源仅仅是执行脉动信号监视、管理集群资源组和集群节点以及处理集群资源组与集群节点之间发生的消息传递所必需的资源。在集群环境正常运行后，只有对集群进行更改，才会增加开销。

在正常操作环境中，集群活动对集群系统的影响微乎其微。

集群的规划核对表

请完成集群配置核对表，以确保在开始配置集群前正确地准备环境。

表 1. 集群的 TCP/IP 配置核对表

TCP/IP 需求	
—	使用“启动 TCP/IP” (STRTCP) 命令，在您计划包括在集群中的每个节点上启动 TCP/IP。
—	配置 TCP 回送地址 (127.0.0.1) 并验证其状态是否为“活动”。通过在集群中每个节点上使用使用 TCP/IP 网络状态 (WRKTCPSTS) 命令，验证 TCP/IP 回送地址。
—	通过在节点上使用使用 TCP/IP 网络状态 (WRKTCPSTS) 命令，验证该节点上用于集群功能的 IP 地址是否处于活动状态。

表 1. 集群的 TCP/IP 配置核对表 (续)

TCP/IP 需求	
—	<p>通过完成下列步骤，使用 STRTCPSVR *INETD 命令或通过 IBM Systems Director Navigator for i5/OS 验证 INETD 是否在集群中的所有节点上都处于活动状态：</p> <ol style="list-style-type: none"> 1. 在 Web 浏览器中输入 <code>http://mysystem:2001</code>，其中 <code>mysystem</code> 是系统的主机名。 2. 使用您的用户概要文件和密码登录系统。 3. 从 IBM Systems Director Navigator for i5/OS 中选择网络。 4. 在“欢迎”页上，选择 TCP/IP 服务器。 5. 在“TCP/IP 服务器”页上，选择 TCP/IP 服务器。将显示可用的 TCP/IP 服务器的列表。 6. 从列表中，选择 INETD。 7. 从选择操作菜单中选择启动。该服务器的状态将更改为已启动。 <p>也可以使用“启动 TCP/IP 服务器” (STRTCPSVR) 命令并指定 *INETD 参数来启动 INETD 服务器。您可以通过确定主题节点上的活动作业列表是否包含“用户 QTCP” (QTOGINTD) 作业来验证是否已启动该服务器。</p>
—	<p>验证 INETD 的用户概要文件（在 <code>/QIBM/ProdData/OS400/INETD/inetd.config</code> 中指定）是否未具有除最低权限以外的更多权限。如果此用户概要文件具有除最低权限以外的更多权限，那么可能无法启动集群节点。缺省情况下，已指定 QUSER 作为 INETD 的用户概要文件。</p>
—	<p>验证是否该集群中的每个 IP 地址都可以路由到该集群中的其他每个 IP 地址以及将 UDP 数据报发送到那些 IP 地址。请使用指定了本地 IP 地址的 PING 命令以及指定了 UDP 消息的 TRACEROUTE 命令来执行验证。</p>
—	<p>验证端口 5550 和 5551 是否未被其他应用程序使用。这些端口保留给 IBM 集群功能使用。可以使用使用 TCP/IP 网络状态 (WRKTCPS) 命令查看端口使用情况。INETD 启动后，集群功能将打开端口 5550 并使其处于侦听状态。</p>

表 2. 集群的管理域核对表

集群资源服务的集群界面注意事项	
—	<p>安装 IBM System i High Availability Solutions Manager (iHASM) 许可程序 (5761-HAS)。在高可用性解决方案将要包含的所有集群节点上都必须存在有效的许可证密钥。</p>
—	<p>安装选项 41 (i5/OS - HA 可切换资源)。在设备域将要包含的所有集群节点上都必须存在有效的许可证密钥。</p>
—	<p>通过使用“启动主机服务器” (STRHOSTSVR) 命令，验证是否所有主机服务器都已启动: STRHOSTSVR SERVER (*ALL)</p>

如果您计划在集群中使用可切换设备，那么必须满足下列需求：

表 3. 集群的弹性设备配置核对表

弹性设备需求	
—	<p>安装 IBM System i High Availability Solutions Manager (iHASM) 许可程序 (5761-HAS)。在高可用性解决方案将要包含的所有集群节点上都必须存在有效的许可证密钥。</p>
—	<p>验证是否已安装选项 41 (HA 可切换资源)，并验证在该设备域将要包含的所有集群节点上是否都存在有效的许可证密钥。</p>
—	<p>要访问磁盘管理功能，请访问 DST 并使用用户概要文件来配置服务工具服务器 (STS)。有关详细信息，请参阅启用和访问磁盘机。</p>
—	<p>如果正在系统上的逻辑分区之间切换弹性设备，并且正在使用除 HMC 以外的工具来管理逻辑分区，那么请对分区启用虚拟 OptiConnect。要执行此操作，请登录专用服务工具 (DST)。有关详细信息，请参阅虚拟 OptiConnect。</p> <p>如果您正在使用硬件管理控制台来管理分区，那么请在 OptiConnect 选项卡上更改分区概要文件属性，以便对可切换配置中的每个分区启用虚拟 OptiConnect。必须激活分区概要文件才能使更改生效。</p>

表 3. 集群的弹性设备配置核对表 (续)

弹性设备需求	
—	<p>如果 HSL OptiConnect 环路中的扩展部件在两个系统之间切换，并且其中一个系统包含逻辑分区，那么请对那些分区启用 HSL OptiConnect。如果正在使用除 HMC 以外的工具来管理逻辑分区，那么请登录专用服务工具（DST）并完成此操作。</p> <p>如果您正在使用硬件管理控制台来管理分区，那么请在 OptiConnect 选项卡上更改分区概要文件属性，以便对可切换配置中的每个分区启用 HSL OptiConnect。必须激活分区概要文件才能使更改生效。</p>
—	<p>如果正在逻辑分区之间切换弹性设备，并且正在使用除 HMC 以外的工具来管理逻辑分区，那么必须将总线配置成由各个分区共享或者配置 I/O 池。必须将总线配置成由一个分区“拥有并共享”，并将所有其他将要参与设备切换的分区配置为“使用共享总线”。</p> <p>如果您正在使用硬件管理控制台来管理逻辑分区，那么必须配置包含 I/O 处理器、I/O 适配器以及所有相连资源的 I/O 池，以便允许在分区之间切换独立磁盘池。每个分区都必须能够访问该 I/O 池。有关更多详细信息，请参阅使硬件可切换。有关可切换设备的硬件规划需求的详细信息，请参阅切换式磁盘的硬件需求。</p>
—	<p>在两个不同系统之间切换 HSL 环路中的扩展部件时，请将扩展部件配置为可切换。有关详细信息，请参阅使硬件可切换。</p>
—	<p>将扩展部件添加至现有 HSL 环路时，请重新启动该环路中的所有服务器。</p>
—	<p>通信路径的最大传输单元（MTU）必须大于可调整的集群通信参数“消息段大小”。要验证集群 IP 地址的 MTU，请在主题节点上运行使用 TCP/IP 网络状态（WRKTCPSST）命令。还必须在整个通信路径的每一段上验证 MTU。创建集群后，减小“消息段大小”参数比增大通信路径的 MTU 容易。有关消息段大小的更多信息，请参阅可调整的集群通信参数。可以使用“检索集群资源服务信息”（QcstRetrieveCRSInfo）API 来查看调整参数的当前设置，并可以使用“更改集群资源服务”（QcstChgClusterResourceServices）API 来更改设置。</p>
—	<p>对于地理镜像，请确保对两个节点指定不同的站点名。</p>

表 4. 集群的安全性配置核对表

安全性需求	
—	<p>如果正在尝试启动远程节点，那么请在目标节点上适当地设置“允许添加至集群”（ALWADDCLU）网络属性。根据环境的不同，应该将此属性设置为 *ANY 或 *RQSAUT。如果将此属性设置为 *RQSAUT，那么必须安装 i5/OS 选项 34（数字证书管理器）和选项 35（CCA 密码服务提供者）。有关设置 ALWADDCLU 网络属性的详细信息，请参阅使节点能够被添加至集群。</p>
—	<p>启用 /QIBM/ProdData/OS400/INETD/inetd.config 中指定的 INETD 的用户概要文件。它不能具有 *SECADM 或 *ALLOBJ 特权。缺省情况下，已指定 QUSER 作为 INETD 的用户概要文件。</p>
—	<p>验证调用集群资源服务 API 的用户概要文件是否在所有集群节点上都存在并具有 *IOSYSCFG 权限。</p>
—	<p>验证用于运行集群资源组（CRG）出口程序的用户概要文件是否在所有恢复域节点上都存在。</p>

表 5. 集群的作业配置核对表

作业注意事项	
—	<p>集群资源服务 API 可以通过提交作业来处理请求。这些作业将在创建集群资源组时指定的用于运行出口程序的用户概要文件下运行，或者在请求执行该 API 的用户概要文件下运行（仅用于将弹性设备 CRG 中的设备联机）。请确保将服务于该用户概要文件的相关作业队列的子系统配置为：对于它可以从事该作业队列中运行的作业数，指定 *NOMAX。</p>
—	<p>作业将被提交至作业队列，此作业队列由 CRG 的已定义用户概要文件中的作业描述指定。缺省作业描述将使作业被发送至 QBATCH 作业队列。由于此作业队列用于多个用户作业，因此，出口程序作业可能无法及时地运行。请考虑使用具有唯一用户队列的唯一作业描述。</p>

表 5. 集群的作业配置核对表 (续)

作业注意事项	
—	当出口程序作业运行时，它们使用作业描述中的路由数据来选择要使用的主存储池和运行时属性。缺省值将使那些作业与其他运行优先级为 50 的作业一起在池中运行。这两种情况都会导致出口程序作业的性能不理想。启动出口程序作业的子系统（即使用唯一作业队列的子系统）应该将出口程序作业分配到未被同一子系统或其他子系统启动的其他作业使用的池。此外，应该对出口程序作业指定运行优先级 15，以使它们在几乎所有其他用户作业之前运行。
—	请将 QMLTTHDACN 系统值设置为 1 或 2。

可以通过多个软件界面来配置和管理集群。在这些界面中，“集群资源服务”界面是其中之一。如果您选择使用“集群资源服务”，那么必须满足下列需求。

表 6. 集群的集群资源服务配置核对表

“集群资源服务”图形界面注意事项	
—	安装 IBM System i High Availability Solutions Manager (iHASM) 许可程序 (5761-HAS)。在高可用性解决方案将要包含的所有集群节点上都必须存在有效的许可证密钥。
—	安装选项 41 (i5/OS - HA 可切换资源)。在设备域将要包含的所有集群节点上都必须存在有效的许可证密钥。
—	通过使用“启动主机服务器” (STRHOSTSVR) 命令，验证是否所有主机服务器都已启动: STRHOSTSVR SERVER (*ALL)

规划 FlashCopy

可使用 FlashCopy 来减少在使用 IBM Systems Storage 外部存储器单元的 i5/OS 高可用性环境中使用备份窗口的机率。在使用 FlashCopy 之前，应确保已满足最低需求。

FlashCopy 的硬件需求

要在 i5/OS 高可用性解决方案中使用 FlashCopy，请确保符合最低硬件需求。

FlashCopy 的最低硬件需求如下所示：

- 至少两个 System i 系统或逻辑分区，它们在地理上通过至少一个与这两个系统都连接的 IBM System Storage DS6000 或 DS8000 外部存储器部件分隔。DS6000 和 DS8000 外部存储器部件在所有支持通过光纤通道连接外部存储器的 System i 型号上都受支持。
- 需要下列其中一种受支持的光纤通道适配器：
 - 2766 2 千兆位光纤通道磁盘控制器 PCI
 - 2787 2 千兆位光纤通道磁盘控制器 PCI-X
 - 5760 4 千兆位光纤通道磁盘控制器 PCI-X
- 在进行任何配置前，应该调整系统存储器的磁盘空间大小。应该让源卷使用一组磁盘，让目标卷使用相同的一组磁盘机，并让每个一致性副本使用另一组磁盘机。

FlashCopy 的软件需求

要在 i5/OS 高可用性解决方案中使用 FlashCopy，应该符合最低软件需求。

FlashCopy 具有下列最低软件需求：

- 高可用性解决方案中的每个 System i 模型必须正在运行 i5/OS V6R1 以便与 IBM System i High Availability Solutions Manager (iHASM) 许可程序 (5761-HAS) 配合使用。

注：对于前发行版，仍可使用 Lab Services 提供的 IBM Copy Services for System i 来处理 IBM System Storage 解决方案。如果正在多个平台上使用全局镜像，或者希望在多个 System i 分区上实现全局镜像，那么还可使用 IBM Copy Services for System i。

- 在每个系统上安装 IBM System i High Availability Solutions Manager (iHASM)。
- 确保已安装最新的 PTF。

FlashCopy 的通信需求

将您的简短描述放在此处用作第一段和摘要。

对于 FlashCopy，应该符合下列最低通信需求：

- 至少两个 System i 系统，它们在地理上通过至少一个与这两个系统都连接的 IBM System Storage DS6000 或 DS8000 外部存储器部件分隔。DS6000 和 DS8000 外部存储器部件在所有支持通过光纤通道连接外部存储器的 System i 型号上都受支持。
- 需要下列其中一种受支持的光纤通道适配器：
 - 2766 2 千兆位光纤通道磁盘控制器 PCI
 - 2787 2 千兆位光纤通道磁盘控制器 PCI-X
 - 5760 4 千兆位光纤通道磁盘控制器 PCI-X
- 要支持 DS6000 或 DS8000 上的外部装入源部件，需要新的 IOP：
 - 用于 SAN 装入源的功能部件 2847 PCI-X IOP
- 在进行任何配置前，应该调整系统存储器的磁盘空间大小。应该让源卷使用一组磁盘，让目标卷使用相同的一组磁盘机，并让每个一致性副本使用另一组磁盘机。

用于高可用性的安全性规划

在配置高可用性解决方案之前，应重新评估环境中的当前安全策略并进行适当的更改以使高可用性顺利实施。

分发集群范围的信息

了解使用和管理集群范围的信息对安全性的影响。

可以使用“分发信息”（QcstDistributeInformation）API 将消息从集群资源组恢复域中的一个节点发送至该恢复域中的其他节点。在出口程序处理中，此功能非常有用。但是，您应该注意，此功能不会对该信息进行加密。因此，除非您正在使用安全的网络，否则不应该通过此机制发送机密信息。

通过使用集群散列表 API，可以在集群节点之间共享和复制非持久数据。此数据存储在非持久存储器中。这意味着，此数据只保留到集群散列表不再包含该集群节点为止。只能从集群散列表域中定义的集群节点使用这些 API。该集群节点在集群中必须处于活动状态。

同样，使用集群消息传递功能（包括低级集群消息传递功能）分发的其他信息也未受保护。对出口程序的数据进行更改时，不会对包含该数据的消息进行加密。

将集群与防火墙配合使用时的注意事项

如果您正在使用了防火墙的网络中使用集群功能，那么应该了解一些局限性和要求。

如果将集群功能与防火墙配合使用，那么需要使每个节点都能够将出站消息发送至其他集群节点以及从那些集群节点接收入站消息。对于每个节点上的每个集群地址，在防火墙上必须存在开口，以允许这些地址与其他每个集群节点上的每个集群地址进行通信。通过网络传输的 IP 信息包可以是各种不同类型的流量。集群功能使用

PING（其类型为 ICMP），并且还使用 UDP 和 TCP。在配置防火墙时，可以根据类型对流量进行过滤。要使集群功能有效，防火墙必须允许 ICMP、UDP 和 TCP 类型的流量。可以在任何端口上发送出站流量，并且可以在端口 5550 和 5551 上接收入站流量。

在所有节点上维护用户概要文件

可以通过两种机制在集群中的所有节点上维护用户概要文件。

- 在高可用性环境中，认为各个系统上名称相同的用户概要文件相同。名称是集群中的唯一标识。但是，用户概要文件还包含用户标识号（UID）和组标识号（GID）。为了减少切换期间进行的内部处理量（在切换期间，会使独立磁盘池在一个系统上处于不可用状态，然后使其在另一个系统上处于可用状态），应该在设备 CRG 的恢复域中对 UID 和 GID 值进行同步。可以通过两种方法在高可用性环境中对用户概要文件进行同步。

一种机制是创建集群管理域，以便对集群中各个节点上的共享资源进行监视。集群管理域除了可以监视用户概要文件以外，还可以监视多种类型的资源，从而使您能够轻松方便地管理各个节点之间共享的资源。如果集群管理域处于活动状态，那么当用户概要文件被更新时，所作的更改将自动传播到其他节点。如果集群管理域未处于活动状态，那么这些更改将在集群管理域被激活后传播。由于此方法在高可用性环境中自动维护用户概要文件，因此建议您采用此方法。

在第二种机制中，管理员还可以使用 System i 导航器中的“管理中心”对多个系统和系统组执行功能。此支持包含操作员需要对集群中多个系统执行的一些公共用户管理任务。借助“管理中心”，可以对系统组执行用户概要文件功能。管理员在创建用户概要文件时，可以指定要在目标系统上运行的传播后命令。

要点:

- 如果您计划在集群中共享使用了密码同步功能的用户概要文件，那么必须将“保留服务器安全性”（QRETSVRSEC）系统值设置为 1。
- 如果在添加用户概要文件的受监视资源项（MRE）后将 QRETSVRSEC 更改为 0，然后更改密码（如果正在对密码进行监视），那么该 MRE 的全局状态将设置为“不一致”。该 MRE 将被标记为“不可用”。进行此更改后，对该用户概要文件进行的任何更改都不会进行同步。要解决此问题，请将 QRETSVRSEC 更改为 1，除去该 MRE，然后再次添加该 MRE。

相关任务

第 94 页的『创建集群管理域』

- 在高可用性解决方案中，集群管理域提供了使资源在该集群中系统和分区之间保持同步的机制。

配置高可用性

应首先确保您已经完成适当的规划，并已了解用于高可用性和灾难恢复的资源 and 目标，才能在 i5/OS 环境中配置高可用性解决方案。将配置方案用于高可用性以及与高可用性技术相关联的任务，以创建您自己的高可用性解决方案。

场景：配置高可用性

- 配置场景提供了各种 i5/OS 高可用性环境的示例以及逐步的配置任务，旨在帮助您根据需求和弹性要求实现高可用性解决方案。

- 这些场景包含高可用性的业务目标描述，并提供了插图来说明该高可用性解决方案中的资源。每个解决方案示例都包含用于设置和测试高可用性的逐步指示信息。但是，此信息未阐述所有配置案例以及验证高可用性时可能需要执行的其他测试工作。

场景：逻辑分区之间的切换式磁盘

本场景描述一种 i5/OS 高可用性解决方案，此方案使用在单一系统上的两个逻辑分区之间进行切换的磁盘池。

概述

逻辑分区能够使单个 i5/OS 系统以两个或更多独立系统的方式运行。此解决方案适用于已经在环境中配置了逻辑分区的企业。

本场景未说明逻辑分区的配置。

目标

此解决方案具有下列优点：

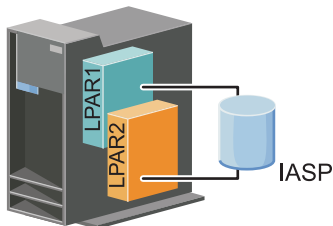
- 这是使用可用系统资源的低成本解决方案。
- 它在计划内中断期间为企业资源提供可用性
- 它在某些计划外中断（如单个逻辑分区故障）期间为企业资源提供可用性
- 因为此解决方案使用单个数据副本，所以将所需磁盘机数降到了最低。
- 此解决方案包含不需要同步的最新数据。

此解决方案具有下列限制：

- 不能对站点范围的中断执行任何灾难恢复。
- 要求您配置逻辑分区。
- 可能要求分区间有冗余硬件。
- 独立磁盘池中只有一个逻辑数据副本。尽管可使用 RAID 保护来保护数据，但这可能造成单点故障。
- 不能从两个逻辑分区并行访问磁盘池。

详细信息

下图描绘了本场景的环境：



配置步骤

请完成下列任务，以配置与本场景相关联的高可用性技术：

1. 完成集群的核对表
2. 创建集群
3. 添加节点
4. 启动节点
5. 将节点添加至设备域
6. 创建集群管理域
7. 启动集群管理域

- | 8. 创建独立磁盘池
- | 9. 添加受监视资源项
- | 10. 使硬件可切换
- | 11. 创建设备 CRG
- | 12. 启动设备 CRG
- | 13. 使磁盘池变为可用状态
- | 14. 执行切换，以测试高可用性解决方案

| 场景：在系统间使用切换式磁盘

| 本场景介绍一个 i5/OS 高可用性解决方案，此方案在两个系统之间使用切换式磁盘，并且在计划内和计划外运行中断期间提供数据高可用性、应用程序高可用性或设备高可用性。

| 概述

| 通过使用切换式磁盘技术，此解决方案提供了简单的高可用性解决方案。借助此解决方案，存储在切换式磁盘中的单个数据副本始终保持最新版本，这样就不必在系统间同步数据，而且避免了传输时丢失数据的风险。

| 目标

| 此解决方案具有下列优点：

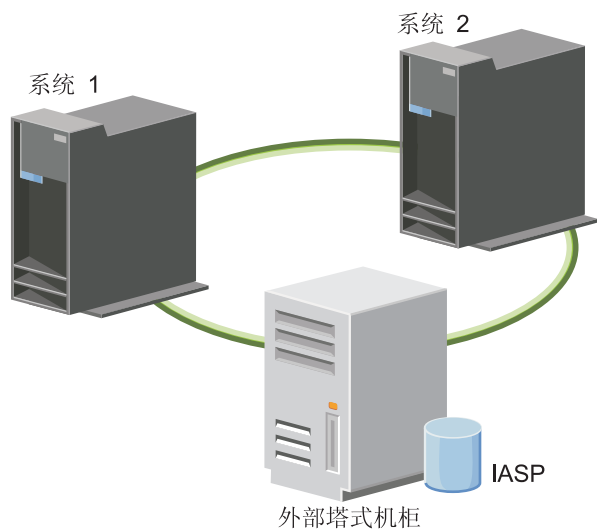
- | • 在计划内中断期间为企业资源提供可用性
- | • 在某些计划外中断期间为企业资源提供可用性
- | • 允许使用单个数据副本，这样可以将所需磁盘机数降至最低
- | • 使性能开销降至最低
- | • 允许数据保持最新版本并且不需要同步

| 此解决方案具有下列限制：

- | • 不能对站点范围的中断执行任何灾难恢复
- | • 独立磁盘池中只有一个逻辑数据副本。尽管可使用 RAID 保护来保护数据，但这可能造成单点故障。
- | • 不能从两个系统并行访问磁盘池

详细信息

下图描述了本场景的环境:



配置步骤

1. 完成规划核对表
2. 创建集群
3. 添加节点
4. 启动节点
5. 将节点添加至设备域
6. 创建集群管理域
7. 启动集群管理域
8. 创建独立磁盘池
9. 添加受监视资源项
10. 使硬件可切换
11. 创建设备 CRG
12. 启动设备 CRG
13. 使磁盘池变为可用状态
14. 执行切换, 以测试高可用性解决方案

场景: 切换式磁盘与地理镜像配合

本场景描述一种 i5/OS 高可用性解决方案, 此方案在三节点集群中将切换式磁盘与地理镜像配合使用。此解决方案提供了灾难恢复和高可用性功能。

概述

在生产站点 (非中心区) 上, 切换式磁盘用于在两个节点间移动独立磁盘池。该解决方案还会使用地理镜像在第二个站点 (中心区) 上生成独立磁盘的副本。因此, 此解决方案同时提供灾难恢复和高可用性。此解决方案的优点在本质上与基本切换式磁盘解决方案相同, 但此解决方案的特别之处在于通过将应用程序数据复制至另

一位置来为该数据提供灾难恢复。生产站点（非中心区）包含独立磁盘池，独立磁盘池可在逻辑分区之间切换以提供高可用性，并且在计划内中断（如应用修订）时能够很快地进行切换。此解决方案还为灾难恢复提供跨站点镜像和地理镜像。

地理镜像是跨站点镜像的子功能，进行地理镜像时数据会镜像至远程位置上的独立磁盘池的副本。生产站点（非中心区）上的独立磁盘池中的数据会镜像至备份站点（中心区）上的独立磁盘池。与基于外部存储器的解决方案（如 IBM System Storage 全局镜像和城域镜像）相比，此解决方案既简单又节省成本。但是，地理镜像不会提供外部存储器解决方案提供的所有性能选项。

目标

此解决方案具有下列优点：

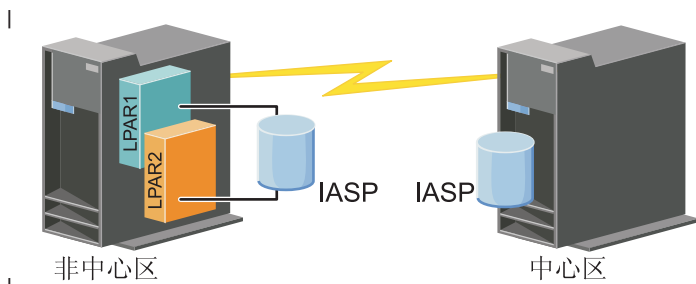
- 在计划内中断期间为企业资源提供可用性
- 在某些计划外中断期间为企业资源提供可用性
- 在发生站点范围灾难期间为企业资源提供可用性
- 允许每个站点包含单个数据副本，这样可以将所需磁盘机数降至最小
- 允许数据保持最新版本并且不需要同步

此解决方案具有下列限制：

- 无法对磁盘池进行并发访问。但是，可拆离镜像副本以便脱机处理数据的第二个副本。
- 因为增加了支持地理镜像所需的中央处理器（CPU）负载，所以可能会影响性能
- 考虑使用冗余通信路径和足够的带宽

详细信息

下图对此解决方案作了说明：



配置步骤

1. 完成集群的规划核对表
2. 创建集群
3. 添加节点
4. 启动节点
5. 将节点添加至设备域
6. 创建设备 CRG
7. 定义站点名
8. 创建集群管理域
9. 启动集群管理域

- | 10. 创建独立磁盘池
- | 11. 添加受监视资源项
- | 12. 使硬件可切换
- | 13. 配置地理镜像
- | 14. 使磁盘池变为可用状态
- | 15. 执行切换，以测试配置。

| 相关任务

- | 第 102 页的『配置地理镜像』
- | 地理镜像是跨站点镜像的子功能。要使用地理镜像来配置高可用性解决方案，需要在生产系统与备份系统之间配置镜像会话。

| 场景：跨站点镜像与地理镜像配合

- | 本场景描述在双节点集群中使用地理镜像的 i5/OS 高可用性解决方案。此解决方案提供了灾难恢复和高可用性功能。

| 概述

- | 地理镜像是跨站点镜像的子功能，进行地理镜像时数据会镜像至远程位置上的独立磁盘池的副本。此解决方案提供了灾难恢复以应付生产系统（系统 1）上出现站点范围中断的情况。在此情况下会进行故障转移以切换到备份站点（系统 2），可在该备份站点上继续对数据的镜像副本执行操作。与基于外部存储器的解决方案（如 IBM System Storage 全局镜像和城域镜像）相比，此解决方案既简单又节省成本。但是，地理镜像不会提供外部存储器解决方案提供的所有性能选项。

| 目标

- | 此解决方案具有下列优点：

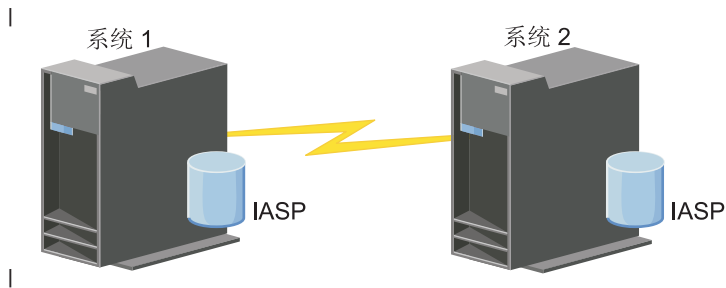
- | • 在计划内中断期间为企业资源提供可用性
- | • 在某些计划外中断期间为企业资源提供可用性
- | • 在发生灾难期间为企业资源提供可用性
- | • 允许数据保持最新版本并且不需要同步

- | 此解决方案具有下列限制：

- | • 无法对磁盘池进行并发访问。但是，可拆离镜像副本以便脱机处理数据的第二个副本。
- | • 因为增加了支持地理镜像所需的中央处理器（CPU）负载，所以可能会影响性能
- | • 考虑使用冗余通信路径和足够的带宽

详细信息

下图对此解决方案作了说明：



配置步骤

1. 完成集群的规划核对表
2. 创建集群
3. 添加节点
4. 启动节点
5. 将节点添加至设备域
6. 创建集群管理域
7. 启动集群管理域
8. 创建独立磁盘池
9. 添加受监视资源项
10. 使硬件可切换
11. 创建设备 CRG
12. 启动设备 CRG
13. 使磁盘池变为可用状态
14. 配置地理镜像
15. 执行切换，以测试配置。

场景：跨站点镜像与城域镜像配合

本场景描述一个 i5/OS 高可用性解决方案，此方案基于外部存储器并为相距不远的存储系统提供灾难恢复和高可用性功能。城域镜像是一个 IBM System Storage 解决方案，此方案以同步方式将数据从生产站点的存储器部件复制至备份站点的存储器部件。这样，数据在备份站点保持一致。

概述

“跨站点镜像与城域镜像配合”解决方案通过在都市区域内使用外部存储器部件来提供高可用性和灾难恢复功能。将在外部存储器设备之间复制独立磁盘池，以便在发生计划内和计划外运行中断时提供高可用性。当城域镜像接收到主机对生产卷的更新时，它对备份卷完成相应的更新。城域镜像支持的最大距离为 300 公里（186 英里）。城域镜像的响应时间延迟与卷之间的距离成正比。

本场景阐述内置 i5/OS 高可用性技术的配置，而未提供与 IBM System Storage DS6000 或 DS8000 相关的安装或配置指示信息。本信息假定，在进行 i5/OS 高可用性配置前，已存在完备的 IBM System Storage 解决方案。有关安装和配置 DS6000 的信息，请参阅 IBM System Storage DS6000 信息中心。有关安装和配置 DS8000 的信息，请参阅 IBM System Storage DS8000 信息中心。

目标

此解决方案具有下列优点:

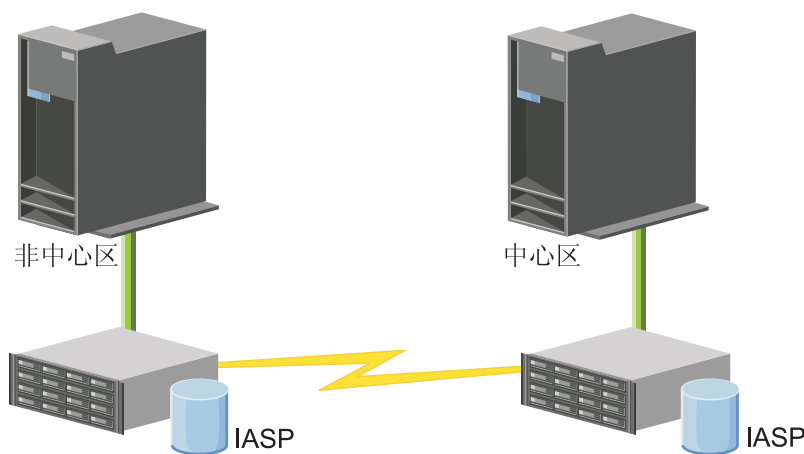
- 复制完全由外部存储器部件管理，因此不必使用 System i 的 CPU。即使系统遇到系统级故障，存储器部件中执行的复制操作也会继续进行。
- 业务资源在计划内和计划外运行中断期间保持可用状态，这包括维护运行中断、与软件/PTF 相关的运行中断以及灾难恢复。
- I/O 保持一致，不需要进行同步
- 如果同时进行日志记录，那么恢复速度较快。日志记录功能能够在发生计划外运行中断或故障转移时快速恢复数据。日志记录功能将数据更改强制写入进行了镜像的磁盘。如果未使用日志记录功能，那么可能会丢失内存中的数据。日志记录功能能够恢复这些数据级事务，并且能够缩短恢复时间。
- 能够在城域镜像的源端或目标端使用 FlashCopy 功能。

此解决方案具有下列限制:

- 需要外部存储器硬件
- 考虑使用冗余通信路径和足够的带宽
- 无法对磁盘池进行并发访问

详细信息

下图对此解决方案作了说明:



配置步骤

1. 完成集群的规划核对表
2. 创建集群
3. 添加节点
4. 启动节点
5. 将节点添加至设备域
6. 创建集群管理域
7. 启动集群管理域
8. 创建独立磁盘池
9. 添加受监视资源项

- | 10. 使硬件可切换
- | 11. 创建设备 CRG
- | 12. 启动设备 CRG
- | 13. 使磁盘池变为可用状态
- | 14. 配置城域镜像会话
- | 15. 执行切换，以测试配置

| 场景：跨站点镜像与全局镜像配合

| 本场景描述一个 i5/OS 高可用性解决方案，此方案基于外部存储器并为相距甚远的存储系统提供灾难恢复和高可用性功能。全局镜像是一个 IBM Systems Storage 解决方案，此方案以异步方式将数据从生产站点的存储器部件复制至备份站点的存储器部件。这样，数据在备份站点保持一致。

| 概述

| “跨站点镜像与全局镜像配合”解决方案通过使用远距离外部存储器部件来提供灾难恢复解决方案。将在外部存储器设备之间复制独立磁盘池，以便在发生计划内和计划外运行中断时提供高可用性。

| 本场景阐述内置 i5/OS 高可用性技术的配置，而未提供与 IBM System Storage DS6000 或 DS8000 相关的安装或配置指示信息。本信息假定，在进行 i5/OS 高可用性配置前，已存在完备的 IBM System Storage 解决方案。有关安装和配置 DS6000 的信息，请参阅 IBM System Storage DS6000 信息中心。有关安装和配置 DS8000 的信息，请参阅 IBM System Storage DS8000 信息中心。

| 目标

| 此解决方案具有下列优点：

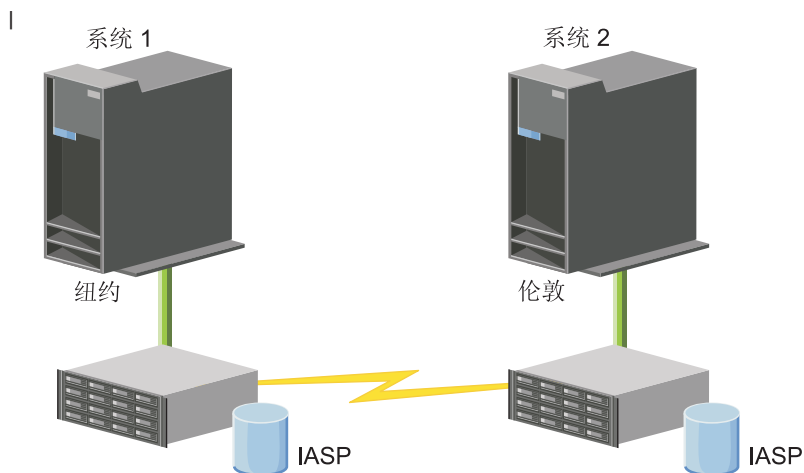
- | • 复制完全由外部存储器部件管理，因此不必使用 System i 的 CPU。即使系统遇到系统级故障，存储器部件中执行的复制操作也会继续进行。
- | • 业务资源在计划内和计划外运行中断期间保持可用状态，这包括维护运行中断、与软件/PTF 相关的运行中断以及灾难恢复。
- | • 如果同时进行日志记录，那么恢复速度较快。日志记录功能能够在发生计划外运行中断或故障转移时快速恢复数据。日志记录功能将数据更改强制写入进行了镜像的磁盘。如果未使用日志记录功能，那么可能会丢失内存中的数据。日志记录功能能够恢复这些数据级事务，并且能够缩短恢复时间。
- | • 能够在全局镜像的源端或目标端使用 FlashCopy 功能。

| 此解决方案具有下列限制：

- | • 需要外部存储器硬件
- | • 考虑使用冗余通信路径和足够的带宽
- | • 无法对磁盘池进行并发访问
- | • 在给定的 System Storage 服务器上，只有一个 System i 分区可配置全局镜像。其他平台上的任何 System i 分区或服务器都不能同时使用全局镜像。向全局镜像会话添加多个用户会导致不可预测的结果。

详细信息

下图对此解决方案作了说明:



配置步骤

1. 完成集群的规划核对表
2. 创建集群
3. 添加节点
4. 启动节点
5. 将节点添加至设备域
6. 创建集群管理域
7. 启动集群管理域
8. 创建独立磁盘池
9. 添加受监视资源项
10. 使硬件可切换
11. 创建设备 CRG
12. 启动设备 CRG
13. 使磁盘池变为可用状态
14. 配置全局镜像会话
15. 执行切换, 以测试配置

对高可用性设置 TCP/IP

因为集群资源服务仅使用 IP 来与其他集群节点（高可用性环境中的系统或逻辑分区）通信，所以所有集群节点必须是可通过 IP 访问的，这意味着您必须先配置 IP 接口才能连接至集群中的节点。

IP 地址必须由网络管理员在每个集群节点上的 TCP/IP 路由表中手工设置，也可通过网络中的路由器上运行的路由协议生成。此 TCP/IP 路由表是集群用于查找每个节点的图；因此，每个节点必须有自己的唯一 IP 地址。

每个节点最多可分配 2 个 IP 地址。其他网络通信应用程序不能以任何方式更改这些地址。确保您在分配每个地址时已考虑到哪个地址使用哪种通信线路。如果已有首选使用的特定通信介质类型，应确保通过使用首选介

质来配置第一个 IP 地址。第一个 IP 地址通过可靠消息功能和脉动信号监视优先处理。节点上的所有 IP 地址必须能够访问集群中的所有其他 IP 地址。如果可执行 ping 操作并双向使用 UDP 消息跟踪路由，那么一个地址可访问另一个地址。

注：您需要确保回送地址（127.0.0.1）处于活动状态，可进行集群。此地址用于将所有消息送回本地节点，缺省情况下通常处于活动状态。但是，如果它已被错误关闭，那么在此地址重新启动前集群消息传递无法工作。

设置 TCP/IP 配置属性

要启用集群资源服务，网络的 TCP/IP 配置中需要特定属性设置。

必须先设置下列属性，才能向集群添加任何节点：

- 如果计划将 System i 产品用作与其他网络通信的路由器，并且该服务器上没有运行任何其他路由协议，那么通过使用 CHGTCPA（更改 TCP/IP 属性）命令将 IP 数据报设置为 *YES。
- 将 INETD 服务器设置为 START。有关启动 INETD 服务器的信息，请参阅『启动 INETD 服务器』。
- 使用 CHGTCPA90（更改 TCP/IP 属性）命令将用户数据报协议（UDP）CHECKSUM 设置为 *YES。
- 如果要使用网桥来连接令牌环网络，那么将 MCAST 转发设置为 *YES。
- 如果使用 OptiConnect for i5/OS 在集群节点间通信，那么通过指定 STRSBS(QSOC/QSOC) 来启动 QSOC 子系统。

启动 INETD 服务器

要添加或启动节点以及进行合并分区处理，必须启动因特网守护程序（INETD）服务器。

建议您始终在集群中运行 INETD 服务器。

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 中选择**网络**。
4. 在“欢迎”页上，选择 **TCP/IP 服务器**。
5. 在“TCP/IP 服务器”页上，选择 **TCP/IP 服务器**。将显示可用的 TCP/IP 服务器的列表。
6. 从列表中，选择 **INETD**。
7. 从**选择操作**菜单中选择**启动**。该服务器的状态将更改为**已启动**。

也可以使用“启动 TCP/IP 服务器”（STRTCPSVR）命令并指定 *INETD 参数来启动 INETD 服务器。INETD 服务器启动后，主题节点上的活动作业列表将包含“用户 QTCP”（QTOGINTD）作业。

相关参考

STRTCPSVR（启动 TCP/IP 服务器）命令

配置集群

任何 i5/OS 高可用性实现都要求配置集群，以便控制和管理弹性资源。与切换式磁盘、跨站点镜像或逻辑复制之类的其他数据弹性技术配合使用时，集群技术提供了高可用性解决方案所必需的关键基础结构。

集群资源服务提供了一组集成服务，这些服务维护集群技术、执行脉动信号传递监视并允许创建和管理集群配置和集群资源组。集群资源服务还提供了可靠的消息传递功能，这些功能跟踪集群中每个节点并确保所有节点包含一致的集群资源状态信息。IBM System i High Availability Solutions Manager (iHASM) 许可程序 (5761-

HAS) 包含的“集群资源服务”图形用户界面允许您配置和管理高可用性解决方案中的集群。并且, 该许可程序还提供了一组用于处理集群配置的控制语言 (CL) 命令。

此外, 为了帮助应用程序提供者或客户改善应用程序可用性, 还提供了应用程序接口 (API) 和设施。

除了这些 IBM 技术以外, 高可用性业务合作伙伴也提供了将集群与逻辑复制技术配合使用的应用程序。

创建集群

要创建集群, 必须至少将一个节点包括在该集群中, 并且您必须能够访问至少一个将要包含在该集群中的节点。

如果仅指定了一个节点, 那么它必须是您当前正在访问的系统。要获取与创建集群相关的需求的完整列表, 请参阅第 65 页的『集群的规划核对表』。

- | 如果将要在集群中使用可切换设备或者通过跨站点镜像技术来配置高可用性解决方案, 那么还有其他需求。请参阅场景: 配置高可用性解决方案, 以获取多个使用这些技术的高可用性解决方案的配置示例。每个场景都提供了逐步的配置任务以及此解决方案所应对的系统运行中断的概述。您可以使用这些示例来配置高可用性解决方案, 也可以对它们进行定制以使其符合您的需要。

要创建集群, 请执行下列步骤:

- | 1. 在 Web 浏览器中输入 `http://mysystem:2001`, 其中 `mysystem` 是系统的主机名。
- | 2. 使用您的用户概要文件和密码登录系统。
- | 3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
- | 4. 在“欢迎”页中, 选择**新建集群**。
- | 5. 按照“新建集群”向导中的指示信息执行操作, 以创建集群。

创建新集群后, “欢迎”页将有所变化, 页顶部显示了该集群的名称。“欢迎”页列示了多个用于处理集群的任务。

创建集群后, 您需要添加其他节点并创建 CRG。

相关信息

创建集群 (CRTCLU) 命令

创建集群 (QcstCreateCluster) API

使节点能够被添加至集群:

必须先设置“允许添加至集群” (ALWADDCLU) 网络属性值, 然后才能将节点添加至集群。

在任何要设置为集群节点的服务器上使用“更改网络属性” (CHGNETA) 命令。CHGNETA 命令用于更改系统的网络属性。ALWADDCLU 网络属性指定一个节点是否允许另一个系统将其添加为集群节点。

注: 您必须具有 *IOSYSCFG 权限才能更改网络属性 ALWADDCLU。

可能的值如下所示:

***SAME**

不更改值。系统附带提供的值为 *NONE。

***NONE**

没有任何其他系统可以将此系统添加为集群节点。

***ANY** 任何其他系统都可以将此系统添加为集群节点。

*RQSAUT

只有在集群添加请求通过认证后，任何其他系统才可以将此系统添加为集群节点。

将检查 ALWADDCLU 网络属性，以确定是否允许所添加的节点成为集群的组成部分以及是否使用 X.509 数字证书来验证集群请求。数字证书是一种可以通过电子方式进行验证的个人标识。如果需要验证，那么发出请求的节点以及正在添加的节点必须在系统上安装下列产品：

- i5/OS 选项 34 (数字证书管理器)
- i5/OS 选项 35 (CCA 密码服务提供者)

如果对 ALWADDCLU 选择了 *RQSAUT，那么必须正确地设置 i5/OS 集群安全性服务器应用程序的认证中心信任列表。此服务器应用程序的标识是 QIBM_QCST_CLUSTER_SECURITY。您至少应该为允许加入集群的节点添加认证中心。

添加节点:

“集群资源服务”图形界面允许您在最初创建集群时创建简单的双节点集群。在 i5/OS 高可用性解决方案中，可以将其他节点添加到集群中。

如果正在创建新集群以使其成为高可用性解决方案的组成部分，那么必须通过该集群中的活动节点添加其他节点。

要将节点添加至现有集群，请执行下列步骤：

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在**集群资源服务**页中，选择**使用集群节点任务**以显示该集群中节点的列表。
5. 在**节点选项卡**上，单击**选择操作**菜单，然后选择**添加节点**操作。将显示“添加节点”页。
6. 在“添加节点”页上，指定新节点的信息。单击**确定**以添加该节点。新节点将显示在节点列表中。一个集群可以包含多达 128 个节点。

启动节点:

“启动集群节点”操作在 i5/OS 高可用性环境中的节点上启动集群功能和集群资源服务。

节点可以启动自身，并且，如果它能找到当前活动集群中的活动节点，那么能够重新加入该集群。

要在节点上启动集群功能，请执行下列步骤：

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在**节点选项卡**上，选择要启动的节点。
5. 单击**选择操作**菜单，然后选择**启动**。在所指定节点上成功地启动集群资源服务后，该节点的状态将设置为“已启动”。

将节点添加至设备域:

设备域是集群中的一小部分节点，它们共享设备资源。

如果您正在实现的高可用性解决方案包含基于独立磁盘池的技术，例如切换式磁盘或跨站点镜像，那么必须将节点定义为设备域的成员。将该节点添加至设备域后，可以创建用于定义集群恢复域的设备集群资源组（CRG）。所有将要包含在设备 CRG 恢复域中的节点都必须在同一个设备域中。一个集群节点只能属于一个设备域。

要创建和管理设备域，必须安装 i5/OS 选项 41（HA 可切换资源）。设备域中的所有集群节点上都必须存在有效的许可证密钥。

要将节点添加至设备域，请执行下列步骤：

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在“集群资源服务”页上，选择**使用集群节点任务**以显示该集群中节点的列表。
5. 在**节点**选项卡上，选择要添加至设备域的节点。
6. 从**选择操作**菜单中选择**属性**。
7. 在**集群**选项卡上的**设备域**字段中，指定设备域的名称，节点将被添加至该设备域。

创建集群资源组（CRG）：

集群资源组（CRG）用于管理高可用性资源，例如应用程序、数据和设备。每种类型的 CRG 都管理高可用性环境中特定类型的资源。

“集群资源服务”图形界面允许您通过创建不同的 CRG 来管理高可用性资源。每种类型的 CRG 都可以单独使用或者与其他 CRG 配合使用。例如，您可能有需要高可用性的独立业务应用程序。使该应用程序支持高可用性后，可以创建 CRG 来帮助管理该应用程序的高可用性。

如果您只希望应用程序在系统运行中断期间可用，而它的数据不必如此，那么可以创建应用程序 CRG。但是，如果要让数据和应用程序都可用，那么可以将它们都存储在设置 CRG 中定义的独立磁盘池中。发生系统运行中断时，整个独立磁盘池将切换到备份节点，从而使该应用程序及其数据都可用。

创建应用程序 CRG：

如果要使高可用性解决方案中的应用程序具备高可用性，那么可以创建应用程序集群资源组（CRG）来管理该应用程序的故障转移。

在创建应用程序 CRG 时，可以指定允许活动接管 IP 地址。在启动允许活动接管 IP 地址的应用程序 CRG 时，将允许该 CRG 启动。

要创建应用程序 CRG，请执行下列步骤：

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在“集群资源服务”页中，选择**使用集群资源组**以显示该集群中集群资源组的列表。
5. 在“集群资源组”页中，单击**选择操作**菜单。
6. 选择**新建应用程序 CRG**，然后单击**执行**。将显示“新建应用程序 CRG”页：
7. 在**常规**页中，指定有关应用程序 CRG 的下列信息：
 - 在**名称**字段中，指定 CRG 的名称。此名称限长 10 个字符。

- 在接管 IP 地址字段中，指定要与该应用程序 CRG 相关联的 IP 地址。此值必须采用点分十进制格式，并且最大长度为 15 个字符。接管 IP 地址允许对应用程序进行访问，而与当前正在运行该应用程序的系统无关。配置接管 IP 地址字段确定是用户还是集群资源服务负责创建该 IP 地址。
- 在描述字段中，输入 CRG 的描述。此描述限长 50 个字符。
- 选择允许重新启动并指定尝试重新启动应用程序 CRG 的次数。在进行故障转移以切换到备份节点前，这些值确定在同一节点上尝试重新启动应用程序的次数。
- 在配置接管 IP 地址字段中，选择是要让集群资源服务还是用户负责配置并管理应用程序 CRG 的接管 IP 地址。可能的值包括：

集群资源服务

如果指定此值，那么在创建 CRG 之前，在恢复域中的任何节点上都不能存在该接管 IP 地址。系统将在所有恢复域节点上创建该地址。如果该 IP 地址已存在，那么将无法创建该应用程序 CRG。

用户 如果指定此值，那么在启动 CRG 之前，必须在恢复域中定义的所有主节点和备份节点上添加该接管 IP 地址。

- 选择允许活动接管 IP 地址以允许对应用程序 CRG 指定活动的接管 IP 地址。仅当“配置接管 IP 地址”字段设置为“集群资源服务”时，此字段才有效。
- 在分布式信息用户队列字段中，指定用于接收分布式信息的用户队列的名称。此名称限长 10 个字符。在库字段中，指定用于接收分布式信息的用户队列所在库的名称。库名不能是 *CURLIB、QTEMP 或 *LIBL。此名称限长 10 个字符。

注：如果将“分发信息用户队列”设置为空，那么还必须将“库”设置为空，将“故障转移等待时间”设置为 0，并将“故障转移缺省操作”设置为 0。

- 在故障转移消息队列字段中，指定此集群资源组发生故障转移时用于接收消息的消息队列的名称。如果设置了此字段，那么在出口程序完成后，恢复域中的所有节点上都必须存在所指定消息队列。此故障转移消息队列不能在独立磁盘池中。在库字段中，指定用于接收故障转移消息的消息队列所在库的名称。库名不能是 *CURLIB、QTEMP 或 *LIBL。
- 在故障转移等待时间字段中，指定等待对集群消息队列中故障转移消息应答的分钟数。可能的值包括：
不等待 在用户不进行干预的情况下继续进行故障转移。

无限期等待

故障转移进行无限期等待，直到接收到对故障转移查询消息的响应为止。

数值 指定等待对故障转移查询消息的响应的分钟数。如果在所指定分钟数内未接收到响应，那么“故障转移缺省操作”字段值将指定如何继续。

- 在故障转移缺省操作字段中，指定在故障转移等待时间限制内未接收到对集群消息队列中故障转移消息的响应时，集群功能应如何操作。可以将此字段设置为**继续进行故障转移**或**取消故障转移**。
- 在出口程序页中，可以指定 CRG 出口程序的信息。除设备 CRG 以外，所有类型的 CRG 都需要出口程序。出口程序将在 CRG 的集群相关事件发生后被调用并响应该事件。
 - 在恢复域页中，将节点添加至恢复域并指定它们在集群中的角色。

创建数据 CRG:

数据集群资源组 (CRG) 主要与多个高可用性业务合作伙伴提供的逻辑复制应用程序配合使用。如果您正在实现基于逻辑复制功能的高可用性解决方案，那么可以创建数据 CRG 以帮助在主节点与备份节点之间复制数据。

要创建数据 CRG，请执行下列步骤：

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在“集群资源服务”页中，选择**使用集群资源组**以显示该集群中集群资源组的列表。
5. 在“集群资源组”页中，单击**选择操作菜单**。
6. 选择**新建数据 CRG**，然后单击**执行**。将显示“新建数据 CRG”页。
7. 在**常规**页上，指定有关数据 CRG 的下列信息：
 - 在**名称**字段中，指定 CRG 的名称。此名称限长 10 个字符。
 - 在**描述**字段中，输入 CRG 的描述。此描述限长 50 个字符。
 - 在**分布式信息用户队列**字段中，指定用于接收分布式信息的用户队列的名称。此名称限长 10 个字符。在**库**字段中，指定用于接收分布式信息的用户队列所在库的名称。库名不能是 `*CURLIB`、`QTEMP` 或 `*LIBL`。此名称限长 10 个字符。

注：如果将“分发信息用户队列”设置为空，那么还必须将“库”设置为空，将“故障转移等待时间”设置为 0，并将“故障转移缺省操作”设置为 0。

 - 在**故障转移消息队列**字段中，指定此集群资源组发生故障转移时用于接收消息的消息队列的名称。如果设置了此字段，那么在出口程序完成后，恢复域中的所有节点上都必须存在所指定消息队列。此故障转移消息队列不能在独立磁盘池中。在**库**字段中，指定用于接收故障转移消息的消息队列所在库的名称。库名不能是 `*CURLIB`、`QTEMP` 或 `*LIBL`。
 - 在**故障转移等待时间**字段中，指定等待对集群消息队列中故障转移消息应答的分钟数。可能的值包括：
 - 不等待** 在用户不进行干预的情况下继续进行故障转移。
 - 无限期等待**
 - 故障转移进行无限期等待，直到接收到对故障转移查询消息的响应为止。
 - 数值** 指定等待对故障转移查询消息的响应的分钟数。如果在所指定分钟数内未接收到响应，那么“故障转移缺省操作”字段值将指定如何继续。
8. 在**出口程序**页中，可以指定 CRG 出口程序的信息。除设备 CRG 以外，所有类型的 CRG 都需要出口程序。出口程序将在 CRG 的集群相关事件发生后被调用并响应该事件。
9. 在**恢复域**页中，将节点添加至恢复域并指定它们在集群中的角色。

创建设备 CRG:

设备集群资源组 (CRG) 由可以作为一个实体进行切换的硬件资源池构成。要在高可用性解决方案中创建可切换的设备，使用这些设备的节点必须包含在设备 CRG 中。

在创建设备 CRG 之前，请将所有将要共享可切换资源的节点添加到设备域。

要创建设备 CRG，请执行下列步骤:

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在“集群资源服务”页中，选择**使用集群资源组**以显示该集群中集群资源组的列表。
5. 在“集群资源组”页中，单击**选择操作菜单**。
6. 选择**新建设备 CRG**，然后单击**执行**。将显示**新建设备 CRG** 向导。仅当恢复域中的所有节点都启动后，**新建设备 CRG** 任务才可用。

7. 然后，按照**新建设备 CRG** 向导中的指示信息执行操作，以创建新的设备 CRG。运行此向导时，可以创建新的设备 CRG。也可以创建新的独立磁盘池或指定要使用的现有磁盘池。

设备 CRG 保留在所有恢复域节点上都相同的硬件资源信息，并验证资源名是否完全相同。您还可以配置集群管理域，以保留集群管理域中完全相同的配置对象的已登记属性，这可能包括资源名。如果正在使用跨站点镜像，那么应该为每个站点上的独立磁盘池和其他类型可切换设备创建不同的设备 CRG。

创建对等 CRG:

您可以创建对等 CRG，以便定义负载均衡环境中的节点角色。

要在集群中创建对等 CRG，请执行下列步骤:

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在“集群资源服务”页中，选择**使用集群资源组**以显示该集群中集群资源组的列表。
5. 在“集群资源组”页中，单击**选择操作菜单**。
6. 选择**新建对等 CRG**，然后单击**执行**。将显示“新建数据 CRG”页。
7. 在**常规**页上，指定有关对等 CRG 的下列信息：
 - 在**名称**字段中，指定 CRG 的名称。此名称限长 10 个字符。
 - 在**描述**字段中，输入 CRG 的描述。此描述限长 50 个字符。
 - 在**应用程序标识**字段中，以 `[VendorName].[ApplicationName]` 格式指定对等集群资源组的应用程序标识。例如，`MyCompany.MyApplication`。此标识限长 50 个字符。
8. 在**出口程序**页中，可以指定 CRG 出口程序的信息。除设备 CRG 以外，所有类型的 CRG 都需要出口程序。出口程序将在 CRG 的集群相关事件发生后被调用并响应该事件。
9. 在**恢复域**页中，将节点添加至恢复域并指定它们在集群中的角色。

启动 CRG:

启动集群资源组 (CRG) 会在 i5/OS 高可用性环境中激活集群。

要启动 CRG，请执行下列任务:

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在“集群资源服务”页中，选择**使用集群资源组**以显示该集群中集群资源组的列表。
5. 在**集群资源组**选项卡中，选择要启动的 CRG 的名称。
6. 从**选择操作菜单**中选择**启动**。“状态”列将显示该 CRG 已启动。

相关信息

启动集群资源组 (STRCRG) 命令

创建集群资源组 (QcstCreateClusterResourceGroup) API

指定消息队列

您可以指定集群消息队列或故障转移消息队列。这些消息队列可以帮助您确定 i5/OS 高可用性环境发生故障的原因。

集群消息队列用于存储集群级消息，并且，它提供了一条消息，该消息控制所有集群资源组（CRG）到特定节点的故障转移。故障转移消息队列用于存储 CRG 级消息，它为每个正在进行故障转移的 CRG 提供一条消息。

指定集群消息队列

注：通过在运行“创建集群”向导时指定消息队列，还可以将集群配置为使用集群消息队列。

要指定集群消息队列，请执行下列步骤：

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在“集群资源服务”页中，单击**显示集群属性**。
5. 在“集群属性”页中，单击**集群消息队列**。
6. 指定以下信息，以创建集群消息队列：
 - 在**名称**字段中，指定一个消息队列的名称，该消息队列将接收用于处理集群级或节点级故障转移的消息。对于节点级故障转移而言，将发送一条消息以控制所有具有相同新主节点的集群资源组的故障转移。如果集群资源组独立地进行故障转移，那么将发送一条消息以控制该集群资源组的故障转移。将在新的主节点上发送该消息。如果设置了此字段，那么当该集群中的节点启动时，所有那些节点上都必须存在所指定消息队列。此消息队列不能在独立磁盘池中。
 - 在**库**字段中，指定用于接收故障转移消息的消息队列所在库的名称。库名不能是 *CURLIB、QTEMP、*LIBL、*USRLIBL、*ALL 或 *ALLUSR。
 - 在**故障转移等待时间**字段中，选择**不等待**或**永久等待**，或者指定等待对集群消息队列中故障转移消息应答的分钟数。
 - 在**故障转移缺省操作**字段中，指定集群资源服务在故障转移消息响应超出故障转移等待时间值时执行的操作。可以将此字段设置为**继续进行故障转移**或**取消故障转移**。

指定故障转移消息队列

要指定故障转移消息队列，请执行下列步骤：

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在“集群资源服务”页中，选择**使用集群资源组**以显示该集群中集群资源组的列表。
5. 在集群资源组列表中，选择要处理的集群资源组。
6. 在“集群资源组”页中，单击**选择操作菜单**，然后选择**属性**。
7. 在“常规”页中指定下列值，以指定故障转移消息队列：
 - 在**故障转移消息队列**字段中，指定此集群资源组发生故障转移时用于接收消息的消息队列的名称。如果设置了此字段，那么在出口程序完成后，恢复域中的所有节点上都必须存在所指定消息队列。此故障转移消息队列不能在独立磁盘池中。
 - 在**库**字段中，指定用于接收故障转移消息的消息队列所在库的名称。库名不能是 *CURLIB、QTEMP 或 *LIBL。
 - 在**故障转移等待时间**字段中，指定等待对故障转移消息队列中故障转移消息应答的分钟数。您还可以指定集群资源服务在故障转移消息响应超出所指定故障转移等待时间时执行的操作。

执行切换

可以通过执行切换来测试高可用性解决方案或者处理主节点的计划内运行中断，例如执行备份操作或者已安排的系统维护操作。

执行手工切换时，将致使当前主节点切换到备份节点。集群资源组的恢复域定义了这些角色。执行切换时，恢复域中当前定义的节点的角色将进行更改，如下所示：

- 对当前主节点指定“最后一个活动备份节点”角色。
- 对当前第一个备份节点指定“主节点”角色。
- 后续备份节点在备份节点顺序中上移一个位置。

只允许对处于活动状态的应用程序 CRG、数据 CRG 和设备 CRG 执行切换。

注：如果对设备 CRG 执行切换，那么应该对用户概要文件名称、UID 和 GID 进行同步，以提高性能。集群管理域简化了对用户概要文件进行的同步。

要对资源执行切换，请执行下列步骤：

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在“集群资源服务”页中，选择**使用集群资源组**以显示该集群中集群资源组的列表。
5. 选择要执行切换的 CRG。可以选择应用程序 CRG、数据 CRG 或设备 CRG 以执行切换。
6. 从**选择操作**菜单中选择**切换**。
7. 在确认面板上，选择**是**。

现在，所选集群资源组将切换到备份节点。“状态”列将更新并显示新节点名。

相关概念

集群管理域

相关任务

第 94 页的『配置集群管理域』

在高可用性环境中，应用程序和运营环境在参与该环境的各个节点间保持一致十分必要。集群管理域是 i5/OS 实现的弹性环境，它确保运营环境在各个节点上保持一致。

相关信息

更改集群资源组主节点（CHGCRGPRI）命令

启动切换（QcstInitiateSwitchOver）API

配置节点

节点是参与 i5/OS 高可用性解决方案的系统或逻辑分区。

有多个与节点配置相关的任务。在使用“创建集群”向导时，可以配置简单的双节点集群。然后，可以添加其他节点，节点总数最高可达 128 个。根据高可用性解决方案所采用技术的不同，可能需要完成其他节点配置任务。

启动节点：

“启动集群节点”操作在 i5/OS 高可用性环境中的节点上启动集群功能和集群资源服务。

节点可以启动自身，并且，如果它能找到当前活动集群中的活动节点，那么能够重新加入该集群。

要在节点上启动集群功能，请执行下列步骤：

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在**节点**选项卡上，选择要启动的节点。
5. 单击**选择操作**菜单，然后选择**启动**。在所指定节点上成功地启动集群资源服务后，该节点的状态将设置为“已启动”。

相关信息

启动集群节点（STRCLUNOD）命令

启动集群节点（QcstStartClusterNode）API

使节点能够被添加至集群：

必须先设置“允许添加至集群”（ALWADDCLU）网络属性值，然后才能将节点添加至集群。

在任何要设置为集群节点的服务器上使用“更改网络属性”（CHGNETA）命令。CHGNETA 命令用于更改系统的网络属性。ALWADDCLU 网络属性指定一个节点是否允许另一个系统将其添加为集群节点。

注：您必须具有 *IOSYSCFG 权限才能更改网络属性 ALWADDCLU。

可能的值如下所示：

***SAME**

不更改值。系统附带提供的值为 *NONE。

***NONE**

没有任何其他系统可以将此系统添加为集群节点。

***ANY** 任何其他系统都可以将此系统添加为集群节点。

***RQSAUT**

只有在集群添加请求通过认证后，任何其他系统才可以将此系统添加为集群节点。

将检查 ALWADDCLU 网络属性，以确定是否允许所添加的节点成为集群的组成部分以及是否使用 X.509 数字证书来验证集群请求。数字证书是一种可以通过电子方式进行验证的个人标识。如果需要进行验证，那么发出请求的节点以及正在添加的节点必须在系统上安装下列产品：

- i5/OS 选项 34（数字证书管理器）
- i5/OS 选项 35（CCA 密码服务提供者）

如果对 ALWADDCLU 选择了 *RQSAUT，那么必须正确地设置 i5/OS 集群安全性服务器应用程序的认证中心信任列表。此服务器应用程序的标识是 QIBM_QCST_CLUSTER_SECURITY。您至少应该为允许加入集群的节点添加认证中心。

添加节点：

“集群资源服务”图形界面允许您在最初创建集群时创建简单的双节点集群。在 i5/OS 高可用性解决方案中，可以将其他节点添加到集群中。

如果正在创建新集群以使其成为高可用性解决方案的组成部分，那么必须通过该集群中的活动节点添加其他节点。

要将节点添加至现有集群，请执行下列步骤：

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在**集群资源服务**页中，选择**使用集群节点任务**以显示该集群中节点的列表。
5. 在**节点**选项卡上，单击**选择操作**菜单，然后选择**添加节点**操作。将显示“添加节点”页。
6. 在“添加节点”页上，指定新节点的信息。单击**确定**以添加该节点。新节点将显示在节点列表中。一个集群可以包含多达 128 个节点。

相关信息

添加集群节点项 (ADDCLUNODE) 命令

添加集群节点项 (QcstAddClusterNodeEntry) API

将节点添加至设备域:

设备域是集群中的一小部分节点，它们共享设备资源。

- 1 如果您正在实现的高可用性解决方案包含基于独立磁盘池的技术，例如切换式磁盘或跨站点镜像，那么必须将节点定义为设备域的成员。将该节点添加至设备域后，可以创建用于定义集群恢复域的设备集群资源组 (CRG)。所有将要包含在设备 CRG 恢复域中的节点都必须在同一个设备域中。一个集群节点只能属于一个设备域。

要创建和管理设备域，必须安装 i5/OS 选项 41 (HA 可切换资源)。设备域中的所有集群节点上都必须存在有效的许可证密钥。

要将节点添加至设备域，请执行下列步骤:

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在“集群资源服务”页上，选择**使用集群节点任务**以显示该集群中节点的列表。
5. 在**节点**选项卡上，选择要添加至设备域的节点。
6. 从**选择操作**菜单中选择**属性**。
7. 在**集群**选项卡上的**设备域**字段中，指定设备域的名称，节点将被添加至该设备域。

相关信息

添加设备域项 (ADDDEVDMNE) 命令

添加设备域项 (QcstAddDeviceDomainEntry) API

配置 CRG

集群资源组 (CRG) 用于管理 i5/OS 高可用性环境中的资源。您可以使用多个任务通过 CRG 管理高可用性资源。

启动 CRG:

启动集群资源组 (CRG) 会在 i5/OS 高可用性环境中激活集群。

要启动 CRG，请执行下列任务:

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。

- 1 3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
- 1 4. 在“集群资源服务”页中，选择**使用集群资源组**以显示该集群中集群资源组的列表。
- 1 5. 在**集群资源组**选项卡中，选择要启动的 CRG 的名称。
- 1 6. 从**选择操作**菜单中选择**启动**。“状态”列将显示该 CRG 已启动。

相关信息

启动集群资源组 (STRCRG) 命令

创建集群资源组 (QcstCreateClusterResourceGroup) API

1 创建集群资源组 (CRG) :

1 集群资源组 (CRG) 用于管理高可用性资源，例如应用程序、数据和设备。每种类型的 CRG 都管理高可用性环境中特定类型的资源。

1 “集群资源服务”图形界面允许您通过创建不同的 CRG 来管理高可用性资源。每种类型的 CRG 都可以单独使用或者与其他 CRG 配合使用。例如，您可能有需要高可用性的独立业务应用程序。使该应用程序支持高可用性后，可以创建 CRG 来帮助管理该应用程序的高可用性。

1 如果您只希望应用程序在系统运行中断期间可用，而它的数据不必如此，那么可以创建应用程序 CRG。但是，如果要让数据和应用程序都可用，那么可以将它们都存储在设置 CRG 中定义的独立磁盘池中。发生系统运行中断时，整个独立磁盘池将切换到备份节点，从而使该应用程序及其数据都可用。

1 创建应用程序 CRG:

1 如果要使高可用性解决方案中的应用程序具备高可用性，那么可以创建应用程序集群资源组 (CRG) 来管理该应用程序的故障转移。

1 在创建应用程序 CRG 时，可以指定允许活动接管 IP 地址。在启动允许活动接管 IP 地址的应用程序 CRG 时，将允许该 CRG 启动。

1 要创建应用程序 CRG，请执行下列步骤:

- 1 1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
- 1 2. 使用您的用户概要文件和密码登录系统。
- 1 3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
- 1 4. 在“集群资源服务”页中，选择**使用集群资源组**以显示该集群中集群资源组的列表。
- 1 5. 在“集群资源组”页中，单击**选择操作**菜单。
- 1 6. 选择**新建应用程序 CRG**，然后单击**执行**。将显示“新建应用程序 CRG”页:
- 1 7. 在**常规**页中，指定有关应用程序 CRG 的下列信息:
 - 1 • 在**名称**字段中，指定 CRG 的名称。此名称限长 10 个字符。
 - 1 • 在**接管 IP 地址**字段中，指定要与该应用程序 CRG 相关联的 IP 地址。此值必须采用点分十进制格式，并且最大长度为 15 个字符。接管 IP 地址允许对应用程序进行访问，而与当前正在运行该应用程序的系统无关。**配置接管 IP 地址**字段确定是用户还是集群资源服务负责创建该 IP 地址。
 - 1 • 在**描述**字段中，输入 CRG 的描述。此描述限长 50 个字符。
 - 1 • 选择**允许重新启动**并指定尝试重新启动应用程序 CRG 的次数。在进行故障转移以切换到备份节点前，这些值确定在同一节点上尝试重新启动应用程序的次数。
 - 1 • 在**配置接管 IP 地址**字段中，选择是要让集群资源服务还是用户负责配置并管理应用程序 CRG 的接管 IP 地址。可能的值包括:

集群资源服务

如果指定此值，那么在创建 CRG 之前，在恢复域中的任何节点上都不能存在该接管 IP 地址。系统将在所有恢复域节点上创建该地址。如果该 IP 地址已存在，那么将无法创建该应用程序 CRG。

用户 如果指定此值，那么在启动 CRG 之前，必须在恢复域中定义的所有主节点和备份节点上添加该接管 IP 地址。

- 选择**允许活动接管 IP 地址**以允许对应用程序 CRG 指定活动的接管 IP 地址。仅当“配置接管 IP 地址”字段设置为“集群资源服务”时，此字段才有效。
- 在**分布式信息用户队列**字段中，指定用于接收分布式信息的用户队列的名称。此名称限长 10 个字符。在**库**字段中，指定用于接收分布式信息的用户队列所在库的名称。库名不能是 *CURLIB、QTEMP 或 *LIBL。此名称限长 10 个字符。

注：如果将“分发信息用户队列”设置为空，那么还必须将“库”设置为空，将“故障转移等待时间”设置为 0，并将“故障转移缺省操作”设置为 0。

- 在**故障转移消息队列**字段中，指定此集群资源组发生故障转移时用于接收消息的消息队列的名称。如果设置了此字段，那么在出口程序完成后，恢复域中的所有节点上都必须存在所指定消息队列。此故障转移消息队列不能在独立磁盘池中。在**库**字段中，指定用于接收故障转移消息的消息队列所在库的名称。库名不能是 *CURLIB、QTEMP 或 *LIBL。
- 在**故障转移等待时间**字段中，指定等待对集群消息队列中故障转移消息应答的分钟数。可能的值包括：**不等待** 在用户不进行干预的情况下继续进行故障转移。

无限期等待

故障转移进行无限期等待，直到接收到对故障转移查询消息的响应为止。

数值 指定等待对故障转移查询消息的响应的分钟数。如果在所指定分钟数内未接收到响应，那么“故障转移缺省操作”字段值将指定如何继续。

- 在**故障转移缺省操作**字段中，指定在故障转移等待时间限制内未接收到对集群消息队列中故障转移消息的响应时，集群功能应如何操作。可以将此字段设置为**继续进行故障转移**或**取消故障转移**。

8. 在**出口程序**页中，可以指定 CRG 出口程序的信息。除设备 CRG 以外，所有类型的 CRG 都需要出口程序。出口程序将在 CRG 的集群相关事件发生后被调用并响应该事件。
9. 在**恢复域**页中，将节点添加至恢复域并指定它们在集群中的角色。

相关信息

创建集群资源组 (CRTCRG) 命令

创建集群资源组 (QcstCreateClusterResourceGroup) API

创建数据 CRG:

数据集群资源组 (CRG) 主要与多个高可用性业务合作伙伴提供的逻辑复制应用程序配合使用。如果您正在实现基于逻辑复制功能的高可用性解决方案，那么可以创建数据 CRG 以帮助在主节点与备份节点之间复制数据。

要创建数据 CRG，请执行下列步骤：

1. 在 Web 浏览器中输入 <http://mysystem:2001>，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在“集群资源服务”页中，选择**使用集群资源组**以显示该集群中集群资源组的列表。
5. 在“集群资源组”页中，单击**选择操作**菜单。

6. 选择**新建数据 CRG**，然后单击**执行**。将显示“新建数据 CRG”页。
7. 在**常规**页上，指定有关数据 CRG 的下列信息：
 - 在**名称**字段中，指定 CRG 的名称。此名称限长 10 个字符。
 - 在**描述**字段中，输入 CRG 的描述。此描述限长 50 个字符。
 - 在**分布式信息用户队列**字段中，指定用于接收分布式信息的用户队列的名称。此名称限长 10 个字符。在**库**字段中，指定用于接收分布式信息的用户队列所在库的名称。库名不能是 *CURLIB、QTEMP 或 *LIBL。此名称限长 10 个字符。

注：如果将“分发信息用户队列”设置为空，那么还必须将“库”设置为空，将“故障转移等待时间”设置为 0，并将“故障转移缺省操作”设置为 0。

 - 在**故障转移消息队列**字段中，指定此集群资源组发生故障转移时用于接收消息的消息队列的名称。如果设置了此字段，那么在出口程序完成后，恢复域中的所有节点上都必须存在所指定消息队列。此故障转移消息队列不能在独立磁盘池中。在**库**字段中，指定用于接收故障转移消息的消息队列所在库的名称。库名不能是 *CURLIB、QTEMP 或 *LIBL。
 - 在**故障转移等待时间**字段中，指定等待对集群消息队列中故障转移消息应答的分钟数。可能的值包括：
 - 不等待** 在用户不进行干预的情况下继续进行故障转移。
 - 无限期等待**
 - 故障转移进行无限期等待，直到接收到对故障转移查询消息的响应为止。
 - 数值** 指定等待对故障转移查询消息的响应的分钟数。如果在所指定分钟数内未接收到响应，那么“故障转移缺省操作”字段值将指定如何继续。
8. 在**出口程序**页中，可以指定 CRG 出口程序的信息。除设备 CRG 以外，所有类型的 CRG 都需要出口程序。出口程序将在 CRG 的集群相关事件发生后被调用并响应该事件。
9. 在**恢复域**页中，将节点添加至恢复域并指定它们在集群中的角色。

相关信息

创建集群资源组 (CRTCRG) 命令

创建集群资源组 (QcstCreateClusterResourceGroup) API

创建设备 CRG:

- 设备集群资源组 (CRG) 由可以作为一个实体进行切换的硬件资源池构成。要在高可用性解决方案中创建可切换的设备，使用这些设备的节点必须包含在设备 CRG 中。
- 在创建设备 CRG 之前，请将所有将要共享可切换资源的节点添加到设备域。
- 要创建设备 CRG，请执行下列步骤：
1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
 2. 使用您的用户概要文件和密码登录系统。
 3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
 4. 在“集群资源服务”页中，选择**使用集群资源组**以显示该集群中集群资源组的列表。
 5. 在“集群资源组”页中，单击**选择操作**菜单。
 6. 选择**新建设备 CRG**，然后单击**执行**。将显示**新建设备 CRG** 向导。仅当恢复域中的所有节点都启动后，**新建设备 CRG** 任务才可用。
 7. 然后，按照**新建设备 CRG** 向导中的指示信息执行操作，以创建新的设备 CRG。运行此向导时，可以创建新的设备 CRG。也可以创建新的独立磁盘池或指定要使用的现有磁盘池。

设备 CRG 保留在所有恢复域节点上都相同的硬件资源信息，并验证资源名是否完全相同。您还可以配置集群管理域，以保留集群管理域中完全相同的配置对象的已登记属性，这可能包括资源名。如果正在使用跨站点镜像，那么应该为每个站点上的独立磁盘池和其他类型可切换设备创建不同的设备 CRG。

相关信息

创建集群资源组 (CRTCRG) 命令

创建集群资源组 (QcstCreateClusterResourceGroup) API

创建对等 CRG:

您可以创建对等 CRG，以便定义负载均衡环境中的节点角色。

要在集群中创建对等 CRG，请执行下列步骤:

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在“集群资源服务”页中，选择**使用集群资源组**以显示该集群中集群资源组的列表。
5. 在“集群资源组”页中，单击**选择操作菜单**。
6. 选择**新建对等 CRG**，然后单击**执行**。将显示“新建数据 CRG”页。
7. 在**常规**页上，指定有关对等 CRG 的下列信息：
 - 在**名称**字段中，指定 CRG 的名称。此名称限长 10 个字符。
 - 在**描述**字段中，输入 CRG 的描述。此描述限长 50 个字符。
 - 在**应用程序标识**字段中，以 `[VendorName].[ApplicationName]` 格式指定对等集群资源组的应用程序标识。例如，`MyCompany.MyApplication`。此标识限长 50 个字符。
8. 在**出口程序**页中，可以指定 CRG 出口程序的信息。除设备 CRG 以外，所有类型的 CRG 都需要出口程序。出口程序将在 CRG 的集群相关事件发生后被调用并响应该事件。
9. 在**恢复域**页中，将节点添加至恢复域并指定它们在集群中的角色。

相关信息

创建集群资源组 (CRTCRG) 命令

创建集群资源组 (QcstCreateClusterResourceGroup) API

配置集群管理域

在高可用性环境中，应用程序和运营环境在参与该环境的各个节点间保持一致十分必要。集群管理域是 i5/OS 实现的弹性环境，它确保运营环境在各个节点上保持一致。

创建集群管理域:

在高可用性解决方案中，集群管理域提供了使资源在该集群中系统和分区之间保持同步的机制。

要创建集群管理域，用户必须具有 *IOSYSCFG 权限并有权使用 QCLUSTER 用户概要文件。要管理集群管理域，用户必须有权使用代表该集群管理域的 CRG、QCLUSTER 用户概要文件和集群资源组命令。

要创建集群管理域，请执行下列步骤:

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。

4. 在“集群资源服务”页中，单击**使用管理域**以列示该集群中的集群管理域。如果尚未配置任何集群管理域，那么此列表为空。
5. 在**管理域**选项卡上，选择**新建管理域**。
6. 在“新建管理域”页中，指定有关集群管理域的下列信息：
 - 在**名称**字段中，输入集群管理域的名称。此名称限长 10 个字符。
 - **集群**字段显示集群的名称。您无法更改此字段的值。
 - 在**同步选项**字段中，指定节点加入集群管理域时的同步行为。仅当集群处于 V6 或更高版本时，才会启用此字段。可能的值如下所示：
 - “**上次更改**”选项（**缺省**）
如果对受监视资源所作的所有更改都应该应用于集群管理域，那么请选择此选项。对受监视资源所作的最新更改将应用于所有活动节点上的资源。
 - “**活动域**”选项
如果只允许来自活动节点的受监视资源更改，那么请选择此选项。当该节点加入集群管理域时，将废弃对不活动节点上的受监视资源所作的更改。“活动域”选项不适用于网络服务器存储空间（*NWSSTG）或网络服务器配置（*NWSCFG）。这些资源的同步始终基于最后进行的更改。
 - 从**管理域中的节点**列表中，选择要添加至集群管理域的节点，然后选择**添加**。

相关概念

第 70 页的『在所有节点上维护用户概要文件』
可以通过两种机制在集群中的所有节点上维护用户概要文件。

相关信息

创建集群管理域（CRTCAD）命令
创建集群管理域（QcstCrtClusterAdminDomain）API

将节点添加至集群管理域:

- 可以将附加的节点添加至高可用性解决方案中的集群管理域。
- 在将节点添加至集群管理域之前，请确保该节点也是该集群管理域所在集群的组成部分。如果情况并非如此，那么无法将该节点添加至该集群管理域。该集群管理域不必处于活动状态，但资源直到该集群管理域进入活动状态后才会保持一致。
- 将节点添加至管理域时，将把该域中的 MRE 复制至所添加的节点。如果受监视资源在新节点上不存在，那么集群管理域将创建该资源。如果受监视资源在所添加的节点上已存在，并且该集群管理域处于活动状态，那么该节点将与该域中其余部分进行同步。即，在正在加入集群管理域的节点上，每个受监视资源的属性值都将更改为与活动域中受监视资源的全局值匹配。

要将节点添加至集群管理域，请执行下列步骤:

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在“集群资源服务”页中，单击**使用管理域**以显示该集群中集群管理域的列表。
5. 在“管理域”页中，选择集群管理域。
6. 从**选择操作**菜单中选择**属性**。
7. 在**属性**页中，从**管理域中的节点**列表中选择要添加至集群管理域的节点。单击**添加**。

相关信息

添加集群管理域节点项 (ADDCADNODE) 命令

向恢复域添加节点 (QcstAddNodeToRcvyDomain) API

启动集群管理域:

集群管理域为 i5/OS 高可用性解决方案中的资源提供了环境弹性。

集群管理域启动后, 在集群管理域关闭时对任何受监视资源所作的任何更改将传播至集群管理域中的所有活动节点。

要启动集群管理域, 请执行下列步骤:

1. 在 Web 浏览器中输入 `http://mysystem:2001`, 其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在“集群资源服务”页中, 单击**使用管理域**以显示该集群中集群管理域的列表。
5. 在“管理域”页中, 选择集群管理域。
6. 从**选择操作**菜单中选择**启动**。

“状态”列将显示该集群管理域已启动。

相关概念

『同步受监视资源』

受监视资源在集群管理域中定义的节点上发生更改时, 会对受监视资源执行同步。

相关信息

启动集群管理域 (STRCAD) 命令

同步受监视资源:

受监视资源在集群管理域中定义的节点上发生更改时, 会对受监视资源执行同步。

在此同步过程期间, 集群管理域会尝试更改属性值与其全局值不匹配的每个资源, 除非该资源存在暂挂更改。

所有暂挂更改会分发至域中的所有活动节点, 并且应用于每个节点上每个受影响的资源。分发自暂挂更改后, 全局值会更改, 根据每个节点上针对该资源的更改操作的结果, 每个受影响资源的全局状态会更改为一致或不一致。如果受影响资源在域中的每个活动节点上成功更改, 那么该资源的全局状态为一致。如果该更改在任何节点上失败, 那么全局状态会设置为不一致。

如果在集群管理域处于不活动状态时对多个节点中的同一资源进行了更改, 那么在启动域时, 所有更改将作为同步进程的一部分传播至所有活动节点。尽管所有暂挂更改都是在集群管理域激活期间处理的, 但不能保证处理更改的顺序。如果在集群管理域处于不活动状态时对多个集群节点中的单个资源进行更改, 那么不能保证激活期间处理更改的顺序。

如果某个节点加入不活动的集群管理域 (即, 该节点在集群管理域关闭后启动), 那么直到集群管理域启动后, 受监视资源才会再同步。

注: 集群管理域及其关联出口程序都是 IBM 提供的对象。不应使用 `QcstChangeClusterResourceGroup` API 或“更改集群资源组” (CHGCRG) 命令进行更改, 否则会出现不可预测的结果。

集群管理域中的集群节点关闭后, 仍可在不活动的节点上更改受监视资源。再次启动该节点时, 这些更改将与集群管理域的余下部分再同步。执行再同步过程期间, 集群管理域会将不活动节点中的更改应用于域中的活动

节点，除非节点处于不活动状态时还在活动域中进行了更改。如果同时在活动域和不活动节点中对受监视资源进行了更改，那么在活动域中进行的更改将应用于加入节点。换句话说，不管节点的状态如何，不会丢失对任何受监视资源所作的任何更改。可指定同步选项来控制同步行为。

如果要关闭包含在集群管理域中的集群节点，并且不允许对不活动节点所作的更改在该节点启动时传播回活动域（例如，关闭集群节点以执行测试时），那么必须在关闭集群节点之前从管理域对等 CRG 中除去该节点。

相关概念

除去管理域节点项（RMVCADNODE）命令

相关任务

第 96 页的『启动集群管理域』

集群管理域为 i5/OS 高可用性解决方案中的资源提供了环境弹性。

相关信息

除去 CRG 节点项（RMVCRGNODE）命令

添加受监视资源项:

可以将受监视资源项（MRE）添加至集群管理域。受监视资源项定义关键的资源，以使对这些资源进行的更改在整个高可用性环境中保持一致。

要添加受监视资源项，请执行下列步骤:

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 在“集群资源服务”页中，单击**使用管理域**以显示该集群中集群管理域的列表。
4. 在“管理域”页中，单击集群管理域旁的上下文图标，然后选择**使用受监视资源类型**。

注意：仅当所管理的节点是集群管理域的组成部分时，**使用受监视资源类型**操作才可用。将显示受监视资源类型的当前列表。

5. 在受监视资源类型列表中，单击受监视资源类型旁的上下文图标，然后选择**添加受监视资源项**。将显示“添加受监视资源项”页。
6. 选择要监视的受监视资源项属性，然后单击**确定**。如果该 MRE 对象在库中，那么必须指定该对象的名称和库。新的受监视资源项将添加到集群管理域正在监视的资源列表中。当集群管理域处于活动状态时，对受监视资源所作的更改将在该域中所有活动节点上同步。缺省情况下，将监视所有与受监视资源类型相关联的属性；但是，通过进行选择，您可以控制所要监视的属性。

相关任务

第 126 页的『选择要监视的属性』

添加受监视资源项后，可选择与该资源相关联的属性以便由集群管理域进行监视。

相关信息

添加管理域 MRE（ADDCADMRE）命令

添加受监视资源项（QfpadAddMonitoredResourceEntry）API

配置切换式磁盘

切换式磁盘是已配置成 i5/OS 集群组成部分的独立磁盘池。切换式磁盘允许将独立磁盘池中存储的数据和应用程序切换到另一个系统。

创建独立磁盘池

要创建独立磁盘池，可以使用“新建磁盘池”向导。此向导可以帮助您创建新磁盘池以及将磁盘机添加至磁盘池。

借助“新建磁盘池”向导，可以将未配置的磁盘机包括在奇偶性校验集中，并且可以启动设备奇偶性校验保护和磁盘压缩功能。添加磁盘机时，请不要将同一奇偶性校验集中的磁盘机分到不同的磁盘池中，这样做会导致一个奇偶性校验集中的故障影响多个磁盘池。

要通过 IBM Systems Director Navigator for i5/OS 来使用“新建磁盘池”向导创建独立磁盘池，请执行下列步骤：

注意：要在 IBM Systems Director Navigator for i5/OS 中处理磁盘，必须为专用服务工具配置适当的密码。

IBM Systems Director Navigator for i5/OS

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**配置与服务**。
4. 选择**磁盘机**。
5. 从**选择操作**菜单中选择**新建磁盘池**。
6. 按照向导的指示信息执行操作，以将磁盘机添加至新磁盘池。
7. 打印磁盘配置，以便进行恢复时用作参考。
8. 记录独立磁盘池的名称与编号之间的关系。

System i 导航器

要通过 System i 导航器来使用“新建磁盘池”向导创建独立磁盘池，请执行下列步骤：

1. 在 System i 导航器中，展开**我的连接**（或您的活动环境）。
2. 展开要检查的系统，然后展开**配置与服务** → **硬件** → **磁盘机**。
3. 右键单击**磁盘池**，然后选择**新建磁盘池**。
4. 按照向导的指示信息执行操作，以将磁盘机添加至新磁盘池。
5. 打印磁盘配置，以便进行恢复时用作参考。
6. 记录独立磁盘池的名称与编号之间的关系。

注：请在系统重新启动完成后再添加独立磁盘池。如果必须以专用服务工具（DST）方式使用“新建磁盘池”向导，那么必须在系统重新启动完成后为该独立磁盘池创建相关联的设备描述。请使用“创建设备描述（ASP）”（CRTDEVASP）命令来创建设备描述。并且，请指定与独立磁盘池名称相同的设备描述名称和资源名称。可以使用“使用设备描述”（WRKDEV D）命令来验证设备描述名称与独立磁盘池名称是否匹配。

启动镜像保护

“添加磁盘机”和“新建磁盘池”向导会引导您完成向受保护磁盘池添加相似容量磁盘机对的过程。正确配置磁盘后，就可以启动镜像以获取镜像保护了。

与跨站点镜像不同，镜像保护仅针对单个系统进行。如果要对不可用的独立磁盘池启动镜像，那么可在系统完全重新启动时完成此操作。对于所有其他磁盘池，您需要在启动镜像保护前重新启动系统并进入专用服务工具（DST）方式。

- 1 要使用 IBM Systems Director Navigator for i5/OS 来启动镜像，请执行下列步骤：

- | 1. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**配置与服务**。
- | 2. 选择**磁盘池**。
- | 3. 选择要镜像的磁盘池。
- | 4. 从**选择操作**菜单中选择**启动镜像**。

| 要使用 System i 导航器来启动镜像，请执行下列步骤：

- 1. 在 System i 导航器中，展开**我的连接**（或您的活动环境）。
- 2. 展开要检查的 System i，然后选择**配置与服务** → **硬件** → **磁盘机** → **磁盘池**。
- 3. 右键单击要镜像的磁盘池，然后选择**启动镜像**。

| 停止镜像保护

| 停止镜像保护时，每个镜像对中的一个磁盘机会变为未配置状态。该磁盘池的每个镜像对中必须至少存在一个处于活动状态的磁盘机，才能对磁盘池停止镜像保护。

| 为控制将每个镜像对中的哪个磁盘机变为未配置状态，可暂挂要使其变为未配置状态的磁盘机。对于未暂挂的磁盘机，将自动进行选择。

| 如果要对不可用的独立磁盘池停止镜像，那么可在系统完全重新启动时完成此操作。对于所有其他磁盘池，您需要在停止镜像保护前重新启动系统并进入专用服务工具（DST）方式。

| 与跨站点镜像不同，镜像保护只针对单个系统。

| 要使用 IBM Systems Director Navigator for i5/OS 来停止镜像保护，请执行下列步骤：

- | 1. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**配置与服务**。
- | 2. 选择**磁盘池**。
- | 3. 选择要停止镜像保护的磁盘池。
- | 4. 从**选择操作**菜单中选择**停止镜像**。

| 要使用 System i 导航器来停止镜像保护，请执行下列步骤：

- | 1. 在 System i 导航器中，展开**我的连接**（或您的活动环境）。
- | 2. 展开要检查的 System i，然后选择**配置与服务** → **硬件** → **磁盘机** → **磁盘池**。
- | 3. 选择要对其停止镜像保护的磁盘机。
- | 4. 右键单击任意所选磁盘池，然后选择**停止镜像**。
- | 5. 从出现的确认对话框中单击**停止镜像**。

| 添加磁盘机或磁盘池

| “添加磁盘机”向导允许您使用现有磁盘池来添加新的磁盘机或未配置磁盘机。

| “添加磁盘机”和“磁盘池”向导通过将一些耗时很长的配置功能捆绑成一个高效进程来帮助您节省时间。因为一些向导了解您的系统功能并且只提供有效选项，所以使用它们您就不必猜测磁盘机的配置。例如，除非系统具有压缩功能，否则向导不会列示启动压缩的选项。

| 当您选择将磁盘机添加至受保护磁盘池时，向导会强制您将这些磁盘机加入到设备奇偶性校验保护中，或强制您添加足够的容量相同的磁盘机以启动镜像保护。向导还允许您选择在磁盘池中均衡分发数据或启动磁盘压缩，只要系统配置允许您这样做。由您决定选择哪些选项，所以此操作是完全为您的系统定制的。

| 要使用 IBM Systems Director Navigator for i5/OS 来添加磁盘机或磁盘池，请执行下列步骤：

- | 1. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**配置与服务**。
- | 2. 选择**磁盘机**。
- | 3. 从**选择操作**菜单中选择**添加磁盘机**。
- | 4. 请按照向导的指示信息将新磁盘机添加到磁盘池中。

| 要使用 System i 导航器来添加磁盘机或磁盘池，请执行下列步骤：

- | 1. 在 System i 导航器中，展开**我的连接**（或您的活动环境）。
- | 2. 展开要检查的 System i，然后选择**配置与服务** → **硬件** → **磁盘机**。
- | 3. 要添加磁盘机，请右键单击**所有磁盘机**，然后选择**添加磁盘机**。
- | 4. 请按向导中的指示信息来完成此任务。

评估当前配置

更改系统的磁盘配置之前，应准确了解现有磁盘机相对于磁盘池、IOA 及机架的位置。

System i 导航器的图形视图可用图形来表示您的系统配置，从而免去了编译所有这些信息的过程。可使用该图形视图来执行可通过 System i 导航器的“磁盘机列表”视图访问的任何功能，并且您还可以查看可视表示。如果右键单击该表中的任何对象（如特定磁盘机、磁盘池、奇偶性校验集或机架），那么会在 System i 导航器主窗口中见到相同的选项。

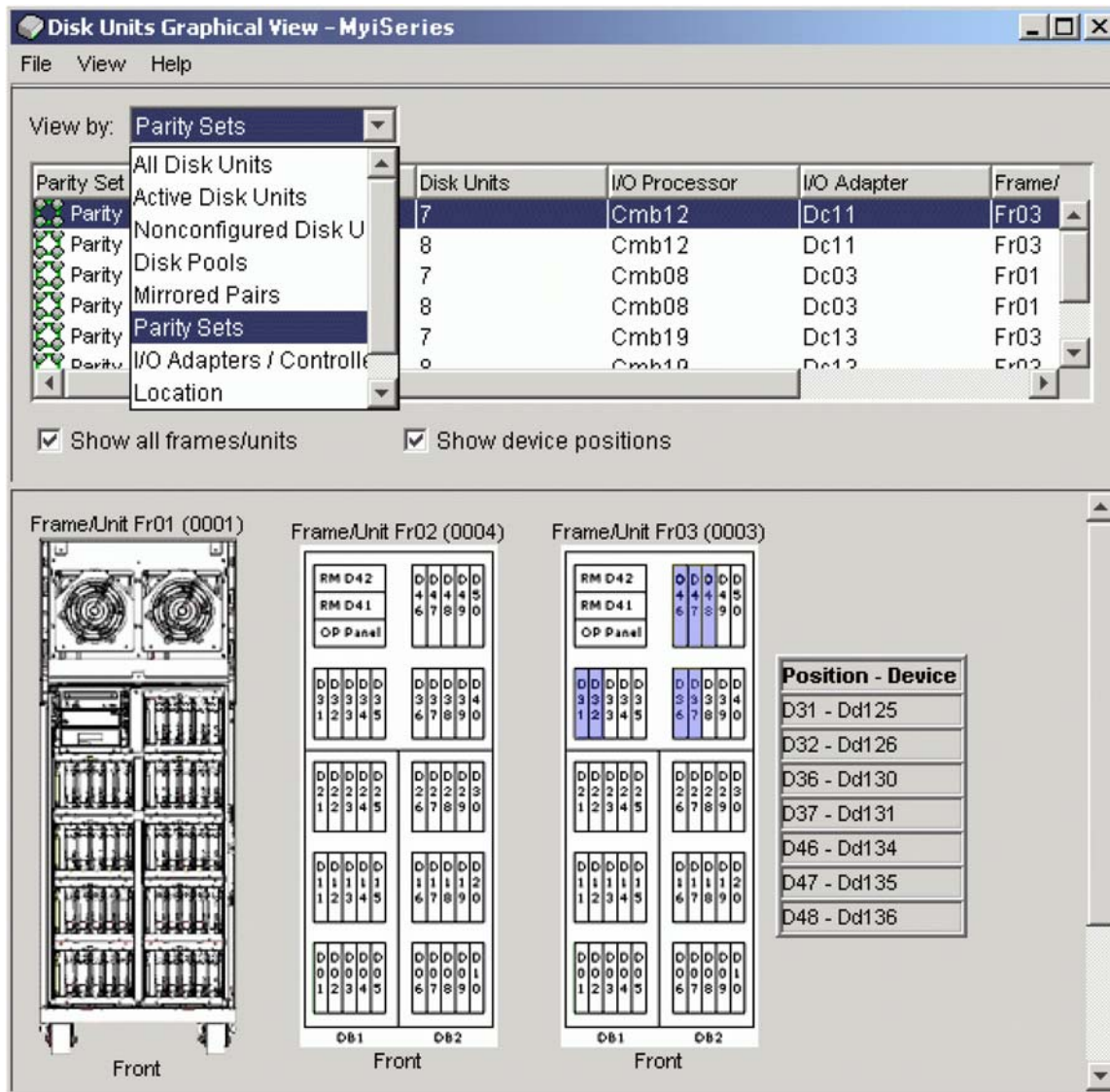
可选择在“磁盘机图形视图”窗口中显示硬件的方式。例如，可选择按磁盘池显示，然后在列表中选择某个磁盘池，以便仅显示包含构成所选磁盘池的磁盘机的机架。不管机架是否包含所选磁盘池中的磁盘机，都可选择“显示所有机架”来查看所有机架。还可选择“显示设备位置”来使磁盘机名称与插入磁盘机的设备位置相匹配。

可在该图形视图中右键单击任意磁盘机并选择要对该磁盘机执行的操作。例如，可选择对磁盘机启动或停止压缩、在奇偶性校验集中包括或排除该磁盘机，或者恢复该磁盘机。如果已对该磁盘机执行镜像保护，那么可对该磁盘机暂挂或恢复镜像。如果右键单击某个空磁盘机槽，那么可启动“安装磁盘机”向导。

要激活该图形视图，请执行下列步骤：

- 1. 在 System i 导航器中，展开**我的连接**（或您的活动环境）。
- 2. 展开要检查的对象，然后选择**配置与服务** → **硬件** → **磁盘机**。
- 3. 右键单击**所有磁盘机**，然后选择**图形视图**。

以下是 System i 导航器中的图形视图的示例。“显示依据”菜单会列示用于显示磁盘机的一些选项。



使磁盘池变为可用状态

要访问独立磁盘池中的磁盘机，必须使该磁盘池变为可用状态（使其联机）。

要访问独立磁盘池中的磁盘机以及相应数据库中的对象，必须使该磁盘池变为可用状态（使其联机）。如果正在使用地理镜像，那么必须使该磁盘池的生产副本变为可用状态。如果生产副本已拆离，那么只能使镜像副本可用。对于地理镜像的磁盘池，除非地理镜像已暂挂，否则还必须在尝试使磁盘池变为可用状态之前确保可切换硬件组已启动。

在多系统集群环境中，可使该磁盘池对当前节点或集群中的另一节点变为可用状态。该独立磁盘池一次只能对一个节点联机。如果要从另一节点访问独立磁盘池，那么必须将该独立磁盘池切换至备份集群节点。请参阅“执行切换”，以了解有关将设备 CRG（在 System i 导航器中又称为可切换硬件组）切换至备份节点的详细信息。

注：如果使主磁盘池或辅助磁盘池变为可用状态，那么该磁盘池组中的所有磁盘池也同时会变为可用状态。

使磁盘池变为可用状态或对独立磁盘池执行磁盘配置更改时，处理可能会停止。如果正在执行其他设备描述活动，那么使磁盘池变为可用状态和进行磁盘配置更改的操作会等待。

如果之前在使地理镜像的磁盘池可用的操作中发生故障，那么可能会导致下一次使该磁盘池变为可用状态或恢复操作时进行完全同步。

要使独立磁盘池变为可用状态，请执行下列步骤：

1. 在 System i 导航器中，展开**我的连接**（或您的活动环境）。
2. 展开要检查的对象，然后选择**配置与服务** → **硬件** → **磁盘机**。
3. 选择**磁盘池**。
4. 右键单击不可用的磁盘池，然后选择**使其变为可用状态**。可同时选择多个要使其变为可用状态的磁盘池。
5. 在显示的对话框中，单击**使其变为可用状态**以使该磁盘池变为可用状态。

可在字符界面中使用“联机配置”（VRYCFG）命令来使该磁盘池变为可用状态。

使用“显示 ASP 状态”（DSPASPSTS）命令来标出正在执行的步骤。

配置跨站点镜像

“跨站点镜像”是地理镜像、城域镜像和全局镜像等多种不同高可用性技术的统称。在这些技术中，每种技术都有特定的相关配置任务。

配置地理镜像

地理镜像是跨站点镜像的子功能。要使用地理镜像来配置高可用性解决方案，需要在生产系统与备份系统之间配置镜像会话。

在配置地理镜像前，必须要有活动的集群、节点和 CRG。并且，您计划用于地理镜像的独立磁盘池还必须处于脱机（不可用）状态才能完成配置。“场景：跨站点镜像与地理镜像配合”主题提供了有关设置基于地理镜像的高可用性解决方案的逐步指示信息。

IBM Systems Director Navigator for i5/OS

要使用 IBM Systems Director Navigator for i5/OS 来配置地理镜像，请执行下列步骤：

1. 在 Web 浏览器中输入 <http://mysystem:2001>，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**配置与服务**。
4. 选择**磁盘池**。
5. 选择要用作生产副本（源副本）的磁盘池。
6. 从**选择操作**菜单中选择**新建会话**。
7. 按照向导的指示信息完成任务。

System i 导航器

要使用 System i 导航器来配置地理镜像，请执行下列步骤：

1. 在 System i 导航器中，展开**我的连接**（或您的活动环境）。
2. 展开要用作生产副本的系统。
3. 展开**配置与服务** → **硬件** → **磁盘机** → **磁盘池**。
4. 右键单击要用作生产副本的磁盘池，然后选择**会话** → **新建**。
5. 按照向导的指示信息完成任务。

相关概念

第 73 页的『场景：切换式磁盘与地理镜像配合』

本场景描述一种 i5/OS 高可用性解决方案，此方案在三节点集群中将切换式磁盘与地理镜像配合使用。此解决方案提供了灾难恢复和高可用性功能。

配置城域镜像会话

对于使用 IBM System Storage 城域镜像技术的 i5/OS 高可用性解决方案而言，需要在 System i 机器与配置了城域镜像的 IBM System Storage 外部存储器部件之间配置会话。在 i5/OS 中，城域镜像会话不会在外部存储器部件上设置镜像，而是在 i5/OS 系统与外部存储器部件上的现有城域镜像配置之间建立关系。

在 i5/OS 上创建城域镜像会话之前，应该在 IBM System Storage 外部存储器部件上配置城域镜像。有关在 IBM System Storage DS6000 上使用城域镜像的信息，请参阅 IBM System Storage DS6000 信息中心。有关在 IBM System Storage DS8000 上使用城域镜像的信息，请参阅 IBM System Storage DS8000 信息中心。

要配置城域镜像会话，请执行下列步骤：

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**配置与服务**。
4. 选择**磁盘池**。
5. 选择要用作生产副本（源副本）的磁盘池。
6. 从**选择操作**菜单中选择**新建会话**。
7. 按照向导的指示信息完成任务。

相关信息

添加 ASP 副本描述（`ADDASPCPYD`）命令

启动 ASP 会话（`STRASPSN`）命令

配置全局镜像会话

对于使用 IBM System Storage 全局镜像技术的 i5/OS 高可用性解决方案而言，需要在 System i 机器与配置了全局镜像的 IBM System Storage 外部存储器部件之间配置会话。在 i5/OS 中，全局镜像会话不会在外部存储器部件上设置镜像，而是在 i5/OS 系统与外部存储器部件上的现有全局镜像配置之间建立关系。

IBM System Storage 全局镜像技术要求所有用户共享一个全局镜像连接。在给定的 System Storage 服务器上，i5/OS 高可用性全局镜像只允许一个 System i 分区配置全局镜像。其他平台上的任何 System i 分区或服务器都不能同时使用全局镜像。向全局镜像会话添加多个用户会导致不可预测的结果。

在 i5/OS 上创建全局镜像会话之前，应该在 IBM System Storage 外部存储器部件上配置全局镜像。有关在 IBM System Storage DS6000 上使用全局镜像的信息，请参阅 IBM System Storage DS6000 信息中心。有关在 IBM System Storage DS8000 上使用全局镜像的信息，请参阅 IBM System Storage DS8000 信息中心。

要配置全局镜像，请执行下列步骤：

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**配置与服务**。
4. 选择**磁盘池**。
5. 选择要用作生产副本（源副本）的磁盘池。
6. 从**选择操作**菜单中选择**新建会话**。
7. 按照向导的指示信息完成任务。

相关信息

添加 ASP 副本描述 (ADDASPCPYD) 命令

启动 ASP 会话 (STRASPCSSN) 命令

管理高可用性

配置 i5/OS 高可用性解决方案后，可使用与高可用性有关的一些界面来管理解决方案。

场景：管理高可用性解决方案

作为高可用性解决方案的系统操作员或管理员，您需要在高可用性环境中执行一些公共任务，例如备份和系统维护。

下列场景提供了有关执行公共系统任务（例如备份和升级）的指示信息以及管理高可用性事件（例如集群分区和故障转移）的示例。每个场景都选择了模型环境。每个场景的指示信息都与该特定高可用性解决方案相对应，并且仅作为示例供您参考。

场景：在高可用性环境中执行备份

根据高可用性解决方案以及备份策略的不同，数据的备份方法也会有所不同。但是，对高可用性环境中的系统执行备份操作时，有一组公共任务。

在多个高可用性解决方案中，您能够根据备份系统上存储的第二份数据执行远程备份。远程备份确保生产系统在您备份第二个系统时保持正常工作状态。这些场景中的每个场景都提供了两个高可用性解决方案的示例，在这些示例中，以远程方式在备份系统上执行备份。

第一个场景在使用地理镜像技术的高可用性解决方案中执行远程备份。第二个场景说明如何在使用 IBM System Storage 解决方案（例如城域镜像或全局镜像）的高可用性环境中使用 FlashCopy。

场景：在地理镜像环境中执行备份：

本场景概述在使用地理镜像的 i5/OS 高可用性解决方案中执行远程备份时必须完成的任务。

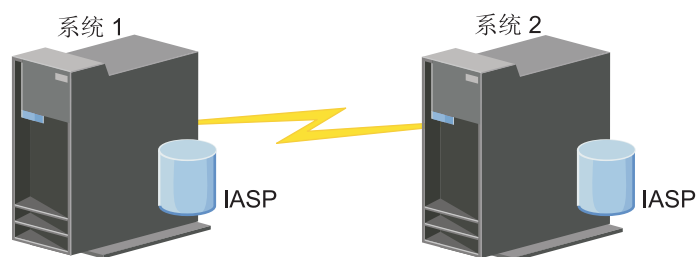
概述

在本示例中，系统管理员需要对高可用性解决方案中使用的独立磁盘池上存储的数据执行备份，该解决方案基于地理镜像技术。此管理员不希望由于执行备份期间将生产系统脱机而对其产生影响。而是，此管理员计划临时地将镜像副本拆离，然后根据远程位置的独立磁盘池上的第二份数据执行备份。

注：在本质上，拆离镜像副本将关闭地理镜像，直到将该副本重新连接至生产副本为止。在拆离期间，高可用性和灾难恢复功能不可用。如果生产系统在此期间发生运行中断，那么某些数据可能会丢失。

详细信息

下图对此环境作了说明:



配置步骤

1. 停止独立磁盘池
2. 第 142 页的『拆离镜像副本』
3. 使磁盘池变为可用状态
4. 备份独立磁盘池
5. 第 140 页的『恢复独立磁盘池』
6. 第 143 页的『重新连接镜像副本』

场景: 执行 *FlashCopy*:

在本示例中, 管理员希望根据备份站点上外部存储器部件中存储的远程数据副本执行备份。通过使用 IBM 存储器解决方案提供的 *FlashCopy* 功能, 管理员可以显著缩短备份时间。

概述

在本示例中, 系统管理员需要对 IBM System Storage 外部存储器部件上存储的数据执行备份。此管理员不希望由于执行备份期间将生产系统脱机而对其产生影响。而是, 他计划执行 *FlashCopy*, 即捕获某个时间点的数据。然后, 根据此数据将数据备份至外部介质。*FlashCopy* 仅需几秒钟即可完成, 这缩短了整个备份过程所需的时间。

在本示例中, 虽然 *FlashCopy* 仅用于执行备份操作, 但您应注意, *FlashCopy* 的用途并不仅限于此。例如, *FlashCopy* 可以用于数据仓储以降低生产系统上的查询工作负载, 也可以用于复制生产数据以创建测试环境。

配置步骤

1. 第 140 页的『停顿独立磁盘池』
2. 第 147 页的『配置 *FlashCopy* 会话』
3. 在 IBM System Storage 外部存储器部件上执行 *FlashCopy*。有关在 IBM System Storage DS6000 上使用 *FlashCopy* 的信息, 请参阅 IBM System Storage DS6000 信息中心。有关在 IBM System Storage DS8000 上使用 *FlashCopy* 的信息, 请参阅 IBM System Storage DS8000 信息中心。
4. 第 140 页的『恢复独立磁盘池』
5. 使磁盘池变为可用状态
6. 备份独立磁盘池

方案: 在高可用性环境中升级操作系统

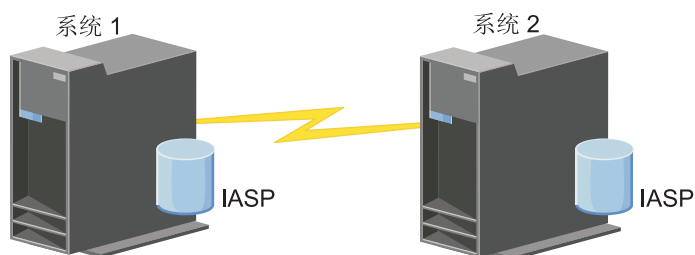
在此示例中, 系统管理员将对基于地理镜像的高可用性解决方案中的两个 i5/OS 系统升级操作系统。

概述

系统管理员需要对高可用性环境中的两个系统升级操作系统。此示例中有两个节点：系统 1 和系统 2。系统 1 是生产副本，而系统 2 是镜像副本。两个系统都使用 i5/OS V5R4。独立磁盘池处于联机状态，地理镜像处于活动状态，而两个系统已同步。系统管理员需要将两个系统都升级至 i5/OSV6R1。

详细信息

以下图形演示该环境：



配置步骤

1. 拆离镜像副本（系统 2）。
2. 关闭 CRG（系统 2）。
3. 停止节点（系统 2）。
4. 将系统 2 升级至新发行版。请参阅升级或替换 i5/OS 和相关软件以了解详细信息。
5. 安装 IBM System i High Availability Solutions Manager (iHASM) 许可程序 (5761-HAS)。
6. 使磁盘池变为可用状态并测试系统 2 上的应用程序。测试这些应用程序可确保它们在新发行版中按预期方式运行。完成应用程序测试后，可通过完成余下步骤来完成升级。
7. 在已拆离的镜像副本上使磁盘池变为不可用状态（系统 2）。
8. 重新连接镜像副本。这会启动镜像数据的再同步。完成再同步后，可继续升级进程。
9. 第 88 页的『执行切换』。这会使镜像副本（系统 2）成为新的生产副本，而原生产副本（系统 1）成为新的镜像副本。

注： 因为不能从执行从 V6R1 至 V5R4 的地理镜像，所以地理镜像会暂挂。可执行从 V5R4 至 V6R1 的地理镜像而不会出现任何问题。在此方案中，地理镜像会在切换完成后暂挂。因为有效备份系统不再存在，所以在升级进程的余下时间内数据没有镜像副本。
10. 关闭 CRG（系统 1）。
11. 停止节点（系统 1）。
12. 将系统 1 升级至新发行版。请参阅升级或替换 i5/OS 和相关软件以了解详细信息。
13. 安装 IBM System i High Availability Solutions Manager (iHASM) 许可程序 (5761-HAS)。
14. 启动节点（系统 1）。
15. 启动 CRG（系统 1）。
16. 恢复镜像
17. 执行切换。这会将当前镜像副本（系统 1）切换回生产副本，而生产副本（系统 2）会成为镜像副本。这是升级之前的初始配置。

示例： 升级操作系统：

在高可用性环境中，必须在升级操作系统前执行特定操作。

下列示例可帮助您确定在集群环境中执行升级前需要执行的操作。在执行升级或任何操作前，应先确定集群的当前版本。

示例 1: 要升级的节点的版本为 **i5/OS V5R4**。集群中的所有其他节点为 **i5/OS V5R4** 或更高版本。当前集群版本为 **V5**。

操作：将节点升级为 **i5/OS V6R1**。升级节点后，对已升级节点启动集群。

示例 2: 要升级的节点的版本为 **i5/OS V5R4**。集群中的所有其他节点的版本为 **i5/OS V5R4**。当前集群版本为 **V4**。

操作：将当前集群版本更改为 **V5**。将该节点升级为 **i5/OS V6R1**。对已升级节点启动集群。

示例 3: 要升级的节点的版本为 **i5/OS V5R3**。集群中的所有其他节点的版本为 **i5/OS V5R4**。当前集群版本为 **V4**。

操作：在升级前从集群中除去要升级为 **i5/OS V6R1** 的节点。将当前集群版本更改为 **V5**。将该节点升级为 **i5/OS V6R1** 并将其添加至该集群。

示例 4: 要升级的节点的版本为 **i5/OS V5R4**。当前仅版本为 **i5/OS V5R3** 和 **i5/OS V5R4** 的节点在集群中。当前集群版本为 **V4**。应先将停留在 **i5/OS V5R3** 的节点升级为 **i5/OS V6R1**，然后再将版本为 **i5/OS V5R4** 的节点升级为 **i5/OS V6R1**。

操作：

1. 从集群中除去节点（此节点将升级）。
2. 将该节点升级为 **i5/OS V6R1**。
3. 至少将停留在 **i5/OS V5R3** 的节点升级为 **i5/OS V5R4**。
4. 更改集群版本至 **V5**。
5. 将已升级节点添加至集群。

方案 5: 要升级的节点的版本为 **i5/OS V5R4**。当前仅版本为 **i5/OS V5R3** 和 **i5/OS V5R4** 的节点在集群中。当前集群版本为 **V4**。应先将版本为 **i5/OS V5R4** 的节点升级为 **i5/OS V6R1**，然后再将停留在 **i5/OS V5R3** 的节点升级为 **i5/OS V6R1**。

操作：

1. 从集群中除去所有 **i5/OS V5R3** 节点。
2. 更改集群版本至 **V5**。
3. 将该节点升级为 **i5/OS V6R1**。
4. 启动已升级节点。
5. 余下 **i5/OS V5R3** 节点升级为 **i5/OS V5R4** 以后，它们可添加到集群中。

方案 6: 要升级的节点的版本为 **i5/OS V5R3**。集群中至少还有一个节点的版本为 **i5/OS V5R3**。当前集群版本低于或等于 **V3**。

操作：将所有节点升级为 **i5/OS V5R4**。将集群版本更改为 **V4**，并将所有节点升级为 **i5/OS V6R1**。

下表提供在集群环境中升级时需要采取的操作。

表 7. 将节点升级为 **i5/OS V6R1**

要升级的节点的当前发行版	当前集群版本	操作
V5R4	5	<ol style="list-style-type: none">1. 将该节点升级为 i5/OS V6R1。2. 启动已升级节点。

表 7. 将节点升级为 i5/OS V6R1 (续)

要升级的节点的当前发行版	当前集群版本	操作
V5R4	4	<ol style="list-style-type: none"> 1. 更改集群版本至 V5。 2. 将该节点升级为 i5/OS V6R1。 3. 启动已升级节点。 <p>注：如果集群中的其他节点的版本为 i5/OS V5R3，那么请参阅“方案 4”和“方案 5”以了解详细信息。</p>
V5R3	低于或等于 V4	<p>选项 A</p> <ol style="list-style-type: none"> 1. 从集群中除去节点（此节点将升级）。 2. 更改集群版本至 V5。 3. 将该节点升级为 V6R1。 4. 向集群添加节点。 <p>选项 B</p> <ol style="list-style-type: none"> 1. 将所有节点升级为 V5R4。 2. 更改集群版本至 V5。 3. 将所有节点升级为 V6R1。

方案：使设备具备高可用性

除了独立磁盘池之外，还可为其他受支持的设备提供高可用性。在此情况下，高可用性管理员需要为以太网线路提供高可用性。

概述

系统管理员需要为高可用性解决方案中使用的以太网线路提供高可用性。当前配置通过使用切换式磁盘技术的两个系统为计划内中断提供高可用性。此解决方案还会使用集群管理域来管理和同步高可用性解决方案运营环境的更改。此示例假定在完成这些步骤之前已成功完成所有高可用性配置和以太网配置。而且还假定高可用性当前处于活动状态，并且所有受监视资源在该环境内保持一致。此示例提供为以太网线路配置高可用性时需要执行的步骤。

配置步骤

1. 第 117 页的『创建可切换设备』
2. 第 97 页的『添加受监视资源项』
3. 第 126 页的『选择要监视的属性』

管理集群

通过使用“集群资源服务”图形界面，可以执行许多与 i5/OS 高可用性解决方案所基于的集群技术相关的任务。这些任务帮助您管理和维护集群。

在配置集群后，可以对其进行的一些更改包括：

集群任务

- 将节点添加至集群
- 从集群中除去节点
- 启动集群节点
- 关闭集群节点
- 将集群的集群版本调整为最新级别
- 删除集群
- 更改集群节点

集群资源组任务

- 创建新的集群资源组
- 删除现有的集群资源组
- 启动集群资源组
- 将节点添加至集群资源组
- 从集群资源组中除去节点
- 关闭集群资源组
- 更改集群资源组的恢复域
- 执行切换
- 将节点添加至设备域
- 从设备域中除去节点

集群管理域任务

- 创建集群管理域
- 添加受监视资源
- 删除集群管理域

调整集群的集群版本

集群版本定义集群中的所有节点相互有效通信时采用的级别。

集群版本化是一种技术，它允许集群包含处于多个发行版级别的系统，并通过确定各个系统之间使用的通信协议来实现充分的协作。

要更改集群版本，集群中的所有节点必须处于相同的潜在版本。这样，就可以更改集群版本以使其与潜在版本相匹配。进行此更改后，就可以使用新功能。版本每次只能提高一级。要降低版本，必须删除集群并在较低版本上重新创建该集群。当前集群版本最初由集群中定义的第一个节点设置。添加到集群中的后续节点必须处于当前集群版本或下一级别的版本；否则无法将它们添加到集群中。

- | 如果正在将节点升级到新发行版，那么必须确保该节点具有适当的集群版本。集群只支持版本相差一级。如果
- | 集群中的所有节点处于同一个发行版，那么应该先升级到新发行版，然后再更改集群版本。这将确保所有与新
- | 发行版相关联的功能都可用。请参阅第 105 页的『方案：在高可用性环境中升级操作系统』主题以了解用于升
- | 级到新发行版的详细操作。

使用下列指示信息来验证和更改节点的集群版本。

- | 1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
- | 2. 使用您的用户概要文件和密码登录系统。

- | 3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
- | 4. 在“集群资源服务”页中，选择**显示集群属性任务**。
- | 5. 在“集群属性”页上，单击**常规选项卡**。
- | 6. 验证集群版本设置，或者将版本更改为期望的设置。

相关概念

集群版本

相关信息

更改集群版本（CHGCLUVER）命令

调整集群版本（QcstAdjustClusterVersion）API

删除集群

删除集群时，将在所有活动集群节点上关闭集群资源服务，并且将从该集群中除去那些集群节点。

- | 在可以删除集群之前，必须至少有一个活动节点。如果集群包含切换式磁盘或其他可切换设备，那么必须首先
- | 从设备域中除去每个节点，然后再删除集群。否则，可能无法将这些磁盘添加回到另一个集群。

要删除集群，请执行下列步骤：

- | 1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
- | 2. 使用您的用户概要文件和密码登录系统。
- | 3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
- | 4. 在**集群资源服务**页中，单击**删除集群**。
- | 5. 将显示**删除集群**确认窗口。选择**是**以删除集群。删除集群后，**集群资源服务**页将改为显示**新建集群任务**。

相关任务

第 115 页的『从设备域中除去节点』

设备域是集群中的一小部分节点，它们共享设备资源。

相关信息

删除集群（DLTCLU）命令

删除集群（QcstDeleteCluster）API

显示集群配置

- | 您可以显示用于提供集群配置信息的详细报告。集群配置报告提供有关集群、节点成员列表、配置参数和调整
- | 参数以及该集群中每个集群资源组的详细信息。

要显示集群配置，请执行下列步骤：

- | 1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
- | 2. 使用您的用户概要文件和密码登录系统。
- | 3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
- | 4. 在**集群资源服务**页中，选择**显示配置信息任务**。这将显示“集群配置和属性”页。可以将此页保存为文件，也可以进行打印。

相关信息

- | 显示集群信息（DSPCLUINF）命令

保存和恢复集群配置

即使在系统上使用了集群功能，创建备份和恢复策略以保护数据仍十分重要。

如果您计划使用集群功能作为备份策略，以便在关闭一个系统进行备份时运行另一个系统，那么建议您至少让集群包含三个系统。如果集群包含三个系统，那么始终能够在发生故障时切换到某个系统。

保存和恢复集群资源组

无论集群是否处于活动状态，您都可以保存集群资源组。恢复集群资源组时，存在下列限制：

- 如果该集群已启动并且不知道该集群资源组，那么无法恢复该集群资源组。
- 如果尚未对该节点配置集群功能，那么无法恢复集群资源组。

如果集群处于活动状态，集群资源组对该集群未知，该节点在该集群资源组的恢复域中，并且集群名与该集群资源组中的集群名匹配，那么可以恢复该集群资源组。如果集群已配置完毕但在该节点上未处于活动状态，并且该节点在该集群资源组的恢复域中，那么可以恢复该集群资源组。

准备应对灾难

发生灾难后，可能需要重新配置集群。为了应对这种情况，建议您保存集群配置信息并创建该信息的硬拷贝打印输出。

1. 更改集群配置后，请运行“保存配置”（SAVCFG）命令或“保存系统”（SAVSYS）命令，以更新所恢复的内部集群信息并使其与集群中其他节点上的该信息保持一致。有关执行 SAVCFG 或 SAVSYS 操作的详细信息，请参阅“保存配置信息”。
2. 每次更改集群配置信息后，请打印该信息的拷贝。可以使用“显示集群信息”（DSPCLUINF）命令来打印集群配置。请将拷贝与备份磁带存放在一起。发生灾难后，可能需要重新配置整个集群。

相关信息

保存配置信息

保存配置（SAVCFG）命令

保存系统（SAVSYS）命令

显示集群信息（DSPCLUINF）命令

| 监视集群状态

| “集群资源服务”图形界面监视集群状态，并且在参与高可用性解决方案的节点变为不一致时显示警告消息。

| 如果集群变为不一致，那么“集群资源服务”图形界面将在“节点”页中显示警告消息 HAI0001W。不一致消息意味着从此节点检索的信息可能与集群中其他活动节点上的信息不一致。当节点在集群中进入不活动状态时，它们将变为不一致。

| 要获取一致的信息，可以从集群中的活动节点访问集群信息，也可以启动此节点并再次尝试该请求。

| 要监视集群状态，请执行下列步骤：

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在“节点”页上，如果节点处于不一致状态，那么将显示 HAI0001W：本地集群节点未处于活动状态。在启动本地节点前，集群信息可能不准确。

| 相关任务

| 第 88 页的『启动节点』

| “启动集群节点”操作在 i5/OS 高可用性环境中的节点上启动集群功能和集群资源服务。

| 相关信息

- | 显示集群信息 (DSPCLUINF) 命令
- | 显示集群资源组信息 (DSPCRGINF) 命令
- | 列示集群信息 (QcstListClusterInfo) API
- | 列示设备域信息 (QcstListDeviceDomainInfo) API
- | 检查集群资源服务信息 (QcstRetrieveCRSInfo) API
- | 检查集群信息 (QcstRetrieveClusterInfo) API
- | 列示集群资源组 (QcstListClusterResourceGroups) API
- | 列示集群资源组信息 (QcstListClusterResourceGroupInf) API

| 指定消息队列

| 您可以指定集群消息队列或故障转移消息队列。这些消息队列可以帮助您确定 i5/OS 高可用性环境发生故障的原因。

| 集群消息队列用于存储集群级消息，并且，它提供了一条消息，该消息控制所有集群资源组 (CRG) 到特定节点的故障转移。故障转移消息队列用于存储 CRG 级消息，它为每个正在进行故障转移的 CRG 提供一条消息。

| 指定集群消息队列

| **注：**通过在运行“创建集群”向导时指定消息队列，还可以将集群配置为使用集群消息队列。

| 要指定集群消息队列，请执行下列步骤：

- | 1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
- | 2. 使用您的用户概要文件和密码登录系统。
- | 3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
- | 4. 在“集群资源服务”页中，单击**显示集群属性**。
- | 5. 在“集群属性”页中，单击**集群消息队列**。
- | 6. 指定以下信息，以创建集群消息队列：
 - | • 在**名称**字段中，指定一个消息队列的名称，该消息队列将接收用于处理集群级或节点级故障转移的消息。对于节点级故障转移而言，将发送一条消息以控制所有具有相同新主节点的集群资源组的故障转移。如果集群资源组独立地进行故障转移，那么将发送一条消息以控制该集群资源组的故障转移。将在新的主节点上发送该消息。如果设置了此字段，那么当该集群中的节点启动时，所有那些节点上都必须存在所指定消息队列。此消息队列不能在独立磁盘池中。
 - | • 在**库**字段中，指定用于接收故障转移消息的消息队列所在库的名称。库名不能是 *CURLIB、*QTEMP、*LIBL、*USRLIBL、*ALL 或 *ALLUSR。
 - | • 在**故障转移等待时间**字段中，选择**不等待**或**永久等待**，或者指定等待对集群消息队列中故障转移消息应答的分钟数。
 - | • 在**故障转移缺省操作**字段中，指定集群资源服务在故障转移消息响应超出故障转移等待时间值时执行的操作。可以将此字段设置为**继续进行故障转移**或**取消故障转移**。

| 指定故障转移消息队列

| 要指定故障转移消息队列，请执行下列步骤：

- | 1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
- | 2. 使用您的用户概要文件和密码登录系统。

3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在“集群资源服务”页中，选择**使用集群资源组**以显示该集群中集群资源组的列表。
5. 在集群资源组列表中，选择要处理的集群资源组。
6. 在“集群资源组”页中，单击**选择操作**菜单，然后选择**属性**。
7. 在“常规”页中指定下列值，以指定故障转移消息队列：
 - 在**故障转移消息队列**字段中，指定此集群资源组发生故障转移时用于接收消息的消息队列的名称。如果设置了此字段，那么在出口程序完成后，恢复域中的所有节点上都必须存在所指定消息队列。此故障转移消息队列不能在独立磁盘池中。
 - 在**库**字段中，指定用于接收故障转移消息的消息队列所在库的名称。库名不能是 *CURLIB、QTEMP 或 *LIBL。
 - 在**故障转移等待时间**字段中，指定等待对故障转移消息队列中故障转移消息应答的分钟数。您还可以指定集群资源服务在故障转移消息响应超出所指定故障转移等待时间时执行的操作。

集群取消配置核对表

要确保彻底地将集群取消配置，必须有系统地除去各个集群组件。

表 8. 集群的独立磁盘池取消配置核对表

独立磁盘池需求	
—	如果您正在使用切换式磁盘池，那么应该将塔式机柜切换到拥有 SPCN 的节点，然后再将集群资源组取消配置。可以使用“启动切换”（QcstInitiateSwitchOver）API 或“更改集群资源组主节点”（CHGCRGPRI）命令将 CRG 移回到 SPCN 所有者。如果未执行此步骤，那么无法将该塔式机柜标记为由该系统专用。
—	如果您计划除去一小部分独立磁盘池组或者除去可切换设备中的最后一个独立磁盘池，那么必须先关闭 CRG。请使用“关闭集群资源组”（ENDCRG）命令。
—	如果要删除参与了集群的独立磁盘池，那么强烈建议您先删除设备集群资源组（CRG）。有关详细信息，请参阅第 117 页的『删除 CRG』。 还可以使用“除去 CRG 设备项”（RMVCRGDEVE）命令从 CRG 中除去独立磁盘池的配置对象。
—	从集群的可切换设备中除去独立磁盘池的配置对象后，就可以删除独立磁盘池了。
—	通过完成下列步骤，删除独立磁盘池的设备描述： <ol style="list-style-type: none"> 1. 在命令行界面中，输入 WRKDEVD DEVD(*ASP) 并按 Enter 键。 2. 向下翻页，直到看到所要删除的独立磁盘池的设备描述为止。 3. 选择该设备描述名旁边的选项 4（删除），然后按 Enter 键。

表 9. 集群的集群资源组取消配置核对表

集群资源组需求	
—	通过完成下列其中一个步骤，删除集群资源组： <ol style="list-style-type: none"> 1. 如果集群功能在节点上未处于活动状态，那么请在命令行界面中输入 DLTCRG CRG(CRGNAME)。CRGNAME 是所要删除的 CRG 的名称。然后，按 Enter 键。 2. 如果集群功能在节点上处于活动状态，那么请在命令行界面中输入 DLTCRGCLU CLUSTER (CLUSTERNAME) CRG(CRGNAME)。CLUSTERNAME 是该集群的名称。CRGNAME 是所要删除的 CRG 的名称。然后，按 Enter 键。

管理节点

i5/OS 高可用性环境中包含的系统和逻辑分区被称为节点。您可以执行许多与节点相关的管理任务。

显示节点属性:

通过使用“集群资源服务”图形界面，可以显示和管理高可用性环境中已配置的节点的相关属性。

要显示节点属性，请完成下列任务:

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在**集群资源服务**页中，选择**使用集群节点**任务以显示该集群中节点的列表。
5. 在**节点**选项卡上，单击**选择操作**菜单，然后选择**属性**。单击**执行**。这将显示“节点属性”页。
 - “常规”页显示了节点的名称以及该节点的系统 IP 地址。
 - “集群”页显示了下列信息:
 - 集群功能与集群中其他节点进行通信时使用的集群接口 IP 地址。
 - 节点的潜在版本指定集群节点相互进行有效通信时采用的版本和修改级别。
 - 所选集群中配置的设备域。如果从列表中选择设备域，那么还将显示属于所选设备域的节点。

停止节点:

停止或关闭节点时，将在该节点上关闭集群功能和集群资源服务。

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在**节点**选项卡上，选择要停止的节点。
5. 单击**选择操作**菜单，然后选择**停止**。在所指定节点上成功地停止集群资源服务后，该节点的状态将设置为“已停止”。

相关信息

关闭集群节点 (ENDCLUNOD) 命令

关闭集群节点 (QcstEndClusterNode) API

除去节点:

- 1 如果正在对节点执行升级，或者该节点不再需要参与 i5/OS 高可用性环境，那么可能需要从集群中除去该节点。

要从现有集群中除去节点，请执行下列步骤:

- 1 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
- 1 2. 使用您的用户概要文件和密码登录系统。
- 1 3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
- 1 4. 在**集群资源服务**页中，选择**使用集群节点**任务以显示该集群中节点的列表。
- 1 5. 在“节点”页中，选择**选择操作**菜单，然后选择**除去**。
- 1 6. 在“确认除去集群节点”窗口中，单击**是**。

相关任务

第 143 页的『取消配置地理镜像』

如果您不再希望能够对特定磁盘池或磁盘池组使用地理镜像，那么可以选择**取消配置地理镜像**。如果取消配置地理镜像，那么系统将停止地理镜像并删除镜像副本站点中各个节点上的磁盘池镜像副本。

相关信息

除去集群节点项 (RMVCLUNODE) 命令

除去集群节点项 (QcstRemoveClusterNodeEntry) API

从设备域中除去节点:

设备域是集群中的一小部分节点，它们共享设备资源。

要点:

从设备域中除去节点时，务必十分谨慎。如果从设备域中除去节点，并且该节点是任何独立磁盘池的当前主访问点，那么那些独立磁盘池仍与所除去的节点在一起。这意味着该设备域中的其他节点不再能够访问那些独立磁盘池。

从设备域中除去节点后，如果一个或多个现有集群节点仍属于该设备域，那么无法将该节点添加回到该设备域。要将该节点添加回到该设备域，必须执行下列操作:

1. 删除正在添加至设备域的节点当前拥有的独立磁盘池。
2. 通过对该节点执行 IPL，重新启动系统。
3. 将该节点添加至设备域。
4. 重新创建步骤 1 中删除的独立磁盘池。

要从设备域中除去节点，请执行下列步骤:

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在**集群资源服务**页中，选择**使用集群节点任务**以显示该集群中节点的列表。
5. 在**节点**选项卡上，选择**选择操作**菜单，然后选择**属性**。单击**执行**。将显示节点属性表。
6. 在**集群**选项卡上，从**设备域**字段中删除该节点名，然后单击**确定**。

相关任务

第 110 页的『删除集群』

删除集群时，将在所有活动集群节点上关闭集群资源服务，并且将从该集群中除去那些集群节点。

相关信息

除去设备域项 (RMVDEVDME) 命令

除去设备域项 (QcstRemoveDeviceDomainEntry) API

管理集群资源组 (CRG)

集群资源组 (CRG) 用于管理 i5/OS 高可用性环境中的弹性资源。它们是一种集群技术，用于定义和控制发生中断时如何将资源切换至备份系统。

显示 CRG 状态:

- 1 您可以监视高可用性环境中集群资源组 (CRG) 的状态。可以使用这些状态消息来验证 CRG 中的更改或者确定 CRG 的问题。

要显示 CRG 状态，请执行下列步骤：

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在“集群资源服务”页中，选择**使用集群资源组**以显示该集群中集群资源组的列表。
5. 在“集群资源组”页的“状态”列中，查看 CRG 的当前状态。

CRG 的可能状态值如下所示：

表 10. CRG 的状态值

可能值	描述
已启动	此 CRG 当前已启动。
已停止	此 CRG 当前已停止。
不确定	高可用性解决方案中有关此 CRG 的信息可能不准确。如果使用操作码“撤销”来调用 CRG 出口程序，并且该出口程序未能成功完成，那么 CRG 将处于此状态。
已恢复	已在此 CRG 的节点上恢复此 CRG，但是尚未将其复制到集群中的其他节点。在节点上启动集群功能后，此 CRG 将与其他节点同步，并且它的状态将设置为“不活动”。
不活动	此 CRG 的集群资源服务在此节点上未处于活动状态。此节点可能已发生故障，此节点可能已关闭或者此节点上的 CRG 作业未处于运行状态。
正在删除	正在从集群中删除此 CRG。
正在更改	正在更改此 CRG。成功完成更改后，此 CRG 将复位为它先前所处的状态。
正在停止	正在停止此 CRG。
正在添加	正在将此 CRG 添加至集群。
正在启动	正在启动此 CRG。
正在切换	此 CRG 正在切换到另一个节点。
正在添加节点	正在将新节点添加至集群。成功添加节点后，此 CRG 将复位为它先前所处的状态。
正在除去节点	正在从此 CRG 中除去节点。成功除去节点后，此 CRG 将复位为它先前所处的状态。
正在更改节点状态	当前正在更改此 CRG 的恢复域中某个节点的状态。

停止 CRG:

集群资源组 (CRG) 用于管理 i5/OS 高可用性环境中的弹性资源。它们是一种集群技术，用于定义弹性资源以及在发生系统运行中断时将那些资源切换到备份系统。

您可能想停止 CRG，以关闭高可用性环境中的自动故障转移功能。例如，您可能正在 CRG 中定义的其中一个系统上执行 IPL。

要停止 CRG，请执行下列步骤：

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。

- | 3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
- | 4. 在“集群资源服务”页中，选择**使用集群资源组**以显示该集群中集群资源组的列表。
- | 5. 在“集群资源组”页中，选择要停止的 CRG。
- | 6. 从**选择操作**菜单中选择**停止**，然后单击**执行**。

| 相关信息

- | 关闭集群资源组 (ENDCRG) 命令
- | 关闭集群资源组 (QcstEndClusterResourceGroup) API

删除 CRG:

可以使用“集群资源服务”界面来删除集群资源组。

要删除 CRG，请执行下列步骤：

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在“集群资源服务”页中，选择**使用集群资源组**以显示该集群中集群资源组的列表。
5. 在“集群资源组”页中，选择要删除的 CRG。
6. 从**选择操作**菜单中选择**删除**，然后单击**执行**。
7. 在“确认删除集群资源组”窗口中，选择**是**。

| 相关信息

- | 从集群中删除集群资源组 (DLTCRGCLU) 命令
- | 删除集群资源组 (QcstDeleteClusterResourceGroup) API

| **创建可切换设备:**

- | 除了独立磁盘池之外，高可用性还支持许多其他设备。以太网线路、光学设备和网络服务器等设备现在可成为高可用性解决方案的一部分。

- | 设备集群资源组包含可切换设备列表。此列表中的每个设备都标识了一个可切换独立磁盘池或者另一种可切换设备，例如磁带机、线路描述、控制器和网络服务器。发生运行中断时，整组设备都将切换到备份节点。您还可以在切换或故障转移过程中使这些设备联机。

| 要创建可切换设备，请执行下列步骤：

- | 1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
- | 2. 使用您的用户概要文件和密码登录系统。
- | 3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
- | 4. 在“集群资源服务”页中，选择**使用集群资源组**以显示该集群中集群资源组的列表。
- | 5. 在“集群资源组”页上，单击要对其添加现有可切换设备的设备集群资源组旁的上下文图标，然后从上下文菜单中选择**添加现有设备**。
- | 6. 在“添加可切换设备列表”中，单击**添加**。
- | 7. 在“添加可切换设备”窗口中，填写可切换设备的配置对象类型和对象名。单击**确定**将新的可切换设备添加到列表中。例如，如果要添加可切换以太网线路，那么对该列表选择以太网线路。
- | 8. 单击列表窗口上的**确定**，以将新设备添加至设备集群资源组。

更改 **CRG** 的恢复域:

恢复域控制集群资源组 (CRG) 中定义的一小部分节点的恢复操作。

- 要更改设备集群资源组、应用程序集群资源组或数据集群资源组的恢复域，请执行下列步骤:
1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在“集群资源服务”页中，选择**使用集群资源组**以显示该集群中集群资源组的列表。
5. 在“集群资源组”页中，选择要更改的 **CRG**。
6. 从**选择操作**菜单中选择**属性**，然后单击**执行**。
7. 单击“恢复域”页，以更改恢复域的现有值。在此页中，可以更改节点在集群恢复域中的角色，也可以对恢复域添加和除去节点。对于设备集群资源组，还可以更改恢复域中某个节点的站点名和数据端口 IP 地址。

相关信息

添加集群资源组节点项 (ADDCRGNODE) 命令

更改集群资源组 (CHGCRG) 命令

除去集群资源组节点项 (RMVCRGNODE) 命令

向恢复域添加节点 (QcstAddNodeToRcvyDomain) API

更改集群资源组 (QcstChangeClusterResourceGroup) API

从恢复域中除去节点 (QcstRemoveNodeFromRcvyDomain) API

创建站点名和数据端口 IP 地址:

- 如果正在使用地理镜像，那么设备集群资源组的恢复域节点中定义的节点必须具有数据端口 IP 地址和站点名。
- 站点名与设备集群资源组的恢复域中某个节点相关联，并且仅适用于地理镜像。为高可用性配置地理镜像环境时，必须对不同站点上的每个节点指定不同的站点名。

要为恢复域中的节点创建数据端口 IP 地址和站点名，请执行下列步骤:

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在“集群资源服务”页中，单击**使用集群资源组**任务，以显示该集群中集群资源组的列表。
5. 在“集群资源组”选项卡上，单击设备集群资源组旁的上下文图标，然后选择**属性**。
6. 在“恢复域”页中，选择**编辑**。
7. 要使用现有的数据端口 IP 地址，请从列表中进行选择，然后单击**确定**。要添加新的数据端口 IP 地址，请单击**添加**。在“添加数据端口 IP 地址”窗口中，输入 IP 地址。
8. 在“编辑”窗口中，可以指定站点名。

管理故障转移中断事件

故障转移通常是节点中断造成的，但也可能有一些其他原因导致故障转移。不同系统或用户操作都可能导致发生故障转移。

只影响单个集群资源组 (CRG) 的问题可能导致该 **CRG** 故障转移，但不会导致任何其他 **CRG** 故障转移。

集群中可出现四种类别的中断。其中某些事件可能导致真正的故障转移（节点遇到中断），而其他事件需要经过调查才能确定导致故障转移的原因并作出适当的响应。下列各表描述每个类别的中断、归入该类别的中断事件的类型以及恢复时应采取的适当操作。

第 1 类中断：导致故障转移的节点中断

发生节点级别的故障转移，导致出现下列情况：

- 对于每个 CRG，主节点被标记为不活动并成为最后一个备份节点。
- 以前充当第一个备份节点的节点变为新的主节点。

故障转移按以下顺序进行：

1. 所有设备 CRG
2. 所有数据 CRG
3. 所有应用程序 CRG

注意：

1. 如果任何 CRG 的故障转移检测到所有备份节点都未处于活动状态，那么该 CRG 的状态将设置为不确定，并且该 CRG 恢复域不会更改。
2. 如果所有集群资源服务失败，那么由这些集群资源服务管理的资源（CRG）将经历故障转移过程。

表 11. 第 1 类中断：导致故障转移的节点中断

故障转移中断事件
发出 ENDTCP (*IMMED 或带有时间限制的 *CNTRLD)。
发出 ENDSYS (*IMMED 或 *CNTRLD)。
发出 PWRDWN SYS (*IMMED 或 *CNTRLD)。
当集群资源服务在系统上处于活动状态时，按下“初始程序装入”（IPL）按钮。
在 CRG 恢复域中的主节点上调用“关闭集群节点”（API 或命令）。
在 CRG 恢复域中的主节点上调用“除去集群节点”（API 或命令）。
发出 HMC 分区延迟断电或面板选项 7。
发出 ENDSBS QSYSWRK (*IMMED 或 *CNTRLD)。
发出“取消作业” (*IMMED 或带有时间限制的 *CNTRLD)。
对 QCSTCRGM 作业发出“取消作业” (*IMMED 或带有时间限制的 *CNTRLD)。

第 2 类中断：导致分区的节点中断

这些中断会导致下列情况：

- 未通过集群消息传递进行通信的节点设置为“分区”状态。请参阅集群分区以了解有关分区的信息。
- 集群分区中未将主节点作为分区成员的所有节点将关闭活动的集群资源组。

注意：

1. 如果节点实际上已失败但仅被检测为分区问题，并且失败的节点为主节点，那么会丢失该节点上的所有数据和应用程序服务，并且不会启动自动故障转移。
2. 必须将该节点声明为已失败，或者启动该节点并再次对该节点启动集群。有关更多信息，请参阅将分区节点更改为“已失败”。

表 12. 第 2 类中断: 导致分区的节点中断

故障转移中断事件
发生 CEC 硬件中断 (例如 CPU)。
发生操作系统软件机器检查。
发出 HMC 立即断电或面板选项 8。
发出 HMC 分区重新启动或面板选项 3。
发生 CEC 掉电。

第 3 类中断: 导致故障转移的 CRG 故障

当 CRG 故障导致故障转移时, 会发生下列情况:

- 如果只有一个 CRG 受到影响, 那么只会对一个 CRG 进行故障转移。这是因为 CRG 是相互独立的。
- 如果某人取消了一些集群资源作业从而导致一些 CRG 同时受到影响, 那么不会在 CRG 之间执行协调故障转移。
- 每个 CRG 中的主节点会被标记为“不活动”并且成为最后一个备份节点。
- 以前充当第一个备份节点的节点变为新的主节点。
- 如果没有活动的备份节点, 那么 CRG 的状态将设置为“不确定”, 那么恢复域将保持不变。

表 13. 第 3 类中断: 导致故障转移的 CRG 故障

故障转移中断事件
CRG 作业存在软件错误, 此软件错误导致该作业异常关闭。
应用程序 CRG 的应用程序出口程序故障

第 4 类中断: 导致分区的通信中断

此类类似第 2 类。事件包括:

- 未通过集群消息传递进行通信的节点设置为“分区”状态。请参阅集群分区以了解有关分区的信息。
- 所有节点及节点上的集群资源服务仍然可操作, 但并非所有节点可相互通信。
- 集群已分区, 但每个 CRG 的主节点仍然提供服务。

对此分区状态的正常恢复应该会修复导致集群分区的通信问题。该集群将消除分区状态而无需任何额外干预。

注: 如果希望 CRG 进行故障转移以切换到新的主节点, 那么应确定旧的主节点在被标记为已失败之前未在使用该资源。有关更多信息, 请参阅将分区节点更改为“已失败”。

表 14. 第 4 类中断: 导致分区的通信中断

故障转移中断事件
集群脉动信号 IP 地址线路上发生通信适配器故障、线路故障或路由器故障。
ENDTCPIFC 将影响集群节点上的所有集群脉动信号 IP 地址。

与活动 CRG 有关的中断

- 如果 CRG 处于“活动状态”并且失败节点并非主节点, 那么会出现下列情况:
 - 故障转移会更新该 CRG 恢复域中失败恢复域成员的状态。
 - 如果失败节点是备份节点, 那么会对备份节点列表重新排序以使活动节点排在列表开头。
- 如果 CRG 处于“活动状态”并且该恢复域成员为主节点, 那么系统执行的操作取决于所发生中断的类型。

- | - 第 1 类中断: 导致故障转移的节点中断
- | - 第 2 类中断: 导致分区的节点中断
- | - 第 3 类中断: 导致故障转移的 CRG 故障
- | - 第 4 类中断: 导致分区的通信中断

与不活动 CRG 有关的中断

| 如果 CRG 发生中断, 那么会出现下列情况:

- | • 集群资源组恢复域中的失败节点的成员资格状态更改为“不活动”或“分区”。
- | • 节点角色未更改, 并且备份节点未自动重新排序。
- | • 调用启动集群资源组 (STRCRG) 命令或启动集群资源组 (QcstStartClusterResourceGroup) API 时, 备份节点将在“不活动 CRG”中重新排序。

| **注:** 如果主节点未处于活动状态, 那么“启动集群资源组”API 将失败。必须发出更改集群资源组 (CHGCRG) 命令或更改集群资源组 (QcstChangeClusterResourceGroup) API 以将活动节点指定为主节点, 然后再次调用“启动集群资源组”API。

管理集群管理域

创建集群管理域并添加适当的受监视资源项 (MRE) 之后, 集群管理员应该监视该管理域中的活动, 以确保受监视资源保持一致。通过使用“集群资源服务”图形界面, 您可以管理和监视集群管理域。

此图形界面能够列示 MRE 以及每个资源的全局状态。通过选择 MRE, 可以显示详细信息。此信息包括与 MRE 相关联的每个属性的全局值, 并指示了该属性在域中是否处于一致状态。如果受监视资源的全局状态不一致, 那么管理员应该执行必需的步骤来确定资源不一致的原因、解决问题并对该资源进行再同步。

如果该资源由于一个或多个节点上的更新操作失败而处于不一致状态, 那么为 MRE 保留的信息可以帮助您确定故障原因。在发生故障的节点上, 将随 MRE 一起记录有关更新失败原因的消息。在其他节点上, 将以内部方式记录一条参考消息, 此消息指示了故障的发生位置以及未能完成更新操作的节点的列表。通过使用“集群资源服务”界面或者调用“检索受监视资源信息” (QfpadRtvMonitoredResourceInfo) API, 可以获得这些消息。故障消息也将记录在对等 CRG 作业的作业记录中。

确定状态不一致的原因后, 通过在发生故障的节点上执行更新操作或者通过关闭并重新启动管理域, 可以对该资源进行再同步。例如, 如果在管理域中的一个节点上更改了用户概要文件的 UID, 但指定的 UID 已被其他某个节点上的另一个用户概要文件使用, 那么该用户概要文件的 MRE 将处于不一致状态。如果重新将 UID 值更改为未被管理域中任何用户概要文件使用的值, 那么集群管理域将在所有节点上进行该更改, 并且会将用户概要文件 MRE 的全局状态设置为“一致”。您不需要执行任何其他操作对用户概要文件 MRE 进行再同步。

在某些情况下, 必须关闭并重新启动集群管理域 CRG 才能对不一致的资源进行再同步。例如, 如果更改了具有相关 MRE 的用户概要文件的 UID, 但该用户概要文件在管理域中其他某个集群节点上的作业中处于活动状态, 那么更改操作在该节点上将失败, 从而导致该用户概要文件的相关 MRE 的全局值处于不一致状态。要更正这种情况, 您必须等待该作业关闭, 然后关闭集群管理域。当管理域再次启动时, 将使用每个不一致属性的全局值使该资源进入一致状态。

如果该资源在域中的任何节点上被删除、重命名或移动, 那么始终会将受监视资源的全局状态设置为“失败”。在这种情况下, 由于集群管理域不再对该资源进行同步, 因此应该除去该 MRE。

在集群管理域中的任何系统上恢复受监视资源时, 将重新使该资源与集群管理域中当前已知的全局值同步, 此全局值是代表该集群管理域的对等 CRG 处于活动状态时的全局值。

下列恢复命令将导致对系统对象进行再同步: RSTLIB、RSTOBJ、RSTUSRPRF 和 RSTCFG。此外, RSTSYSINF 和 UPDSYSINF 将导致对系统值和网络属性进行再同步。要在运行 RSTSYSINF 或 UPDSYSINF 命令后对系统环境变量进行再同步, 必须将代表该集群管理域的对等 CRG 关闭, 然后再次将其启动。

如果要受监视资源恢复到先前状态, 请除去代表所要恢复的资源的 MRE。接着, 在恢复资源后, 从执行恢复操作的系统添加该资源的 MRE。集群管理域将使用所恢复资源的值在整个域中对该受监视资源进行同步。

要监视集群管理域, 请执行下列步骤:

1. 在 Web 浏览器中输入 `http://mysystem:2001`, 其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 在“集群资源服务”页中, 单击**使用管理域**以显示该集群中集群管理域的列表。
4. 在**管理域**选项卡上, 选择**新建管理域**。
5. 在“新建管理域”页中, 指定有关集群管理域的信息。

停止集群管理域:

集群管理域为 i5/OS 高可用性解决方案中的资源提供了环境弹性。您可能需要停止集群管理域以临时关闭受监视资源的同步。

集群管理域停止时会变为不活动状态。当集群管理域处于不活动状态时, 因为对所有受监视资源的更改不会同步, 所以这些受监视资源将被视为不一致。尽管会跟踪对受监视资源的更改, 但全局值不会更改, 并且更改不会传播至管理域的余下部分。集群管理域重新启动后, 在集群管理域处于不活动状态时对任何受监视资源所作的任何更改将在所有活动节点间同步。

注: 集群管理域及其关联出口程序都是 IBM 提供的对象。不应使用 `QcstChangeClusterResourceGroup` API 或 `CHGCRG` 命令进行更改。进行这些更改会导致不可预测的结果。

要停止集群管理域, 请执行下列步骤:

1. 在 Web 浏览器中输入 `http://mysystem:2001`, 其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 在“集群资源服务”页中, 单击**使用管理域**以显示该集群中集群管理域的列表。
4. 在“管理域”页中, 选择**集群管理域**。
5. 从**选择操作**菜单中选择**停止**。
6. 在“确认停止管理域”页上, 单击**是**。

相关信息

关闭集群管理域 (ENDCAD) 命令

删除集群管理域:

通过使用“集群资源服务”界面, 可以删除集群管理域。删除集群管理域后, 将不再对该集群管理域中定义的受监视资源进行同步。

要删除集群管理域, 请执行下列步骤:

1. 在 Web 浏览器中输入 `http://mysystem:2001`, 其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 在“集群资源服务”页中, 单击**使用管理域**以显示该集群中集群管理域的列表。
4. 在“管理域”页中, 选择**集群管理域**。

5. 从**选择操作**菜单中选择**删除**。
6. 在“确认删除管理域”页中，单击**是**。

更改集群管理域的属性:

通过使用“集群资源服务”图形界面，可以更改现有集群管理域的属性。这些属性控制该集群管理域中定义的受监视资源项的同步。

要更改集群管理域的属性，请执行下列步骤:

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 在“集群资源服务”页中，单击**使用管理域**以显示该集群中集群管理域的列表。
4. 在“管理域”页中，选择集群管理域。
5. 从**选择操作**菜单中选择**属性**。
6. 在“属性”页中，可以更改有关集群管理域的下列信息:
 - 在**名称**字段中，输入集群管理域的名称。此名称限长 10 个字符。
 - 在**同步选项**字段中，指定节点加入集群管理域时的同步行为。仅当集群处于 V6 或更高版本时，才会启用此字段。可能的值如下所示:

“上次更改”选项（缺省）

如果对受监视资源所作的所有更改都应该应用于集群管理域，那么请选择此选项。当一个节点加入活动集群管理域时，在该节点处于不活动状态时对该节点上受监视资源所作的任何更改都将应用于该域中其他活动节点上的受监视资源，除非在活动域中已对该资源进行更新的更改。对受监视资源所作的最新更改将应用于所有活动节点上的资源。

“活动域”选项

如果只允许来自活动节点的受监视资源更改，那么请选择此选项。当该节点加入集群管理域时，将废弃对不活动节点上的受监视资源所作的更改。“活动域”选项不适用于网络服务器存储空间（*NWSSTG）或网络服务器配置（*NWSCFG）。这些资源的同步始终基于最后进行的更改。

- 从**管理域中的节点**列表中，可以通过选择**添加**将节点添加至集群管理域，也可以通过选择**除去**从该域中除去节点。

管理受监视资源项:

“集群资源服务”图形界面允许您管理集群管理域中的受监视资源项。集群管理域确保对这些受监视资源所作的更改在高可用性环境中的每个节点上保持一致。

使用受监视资源项状态:

“集群资源服务”图形界面为集群管理域中的受监视资源项提供状态消息。

向集群管理域添加 MRE 之后，会监视该资源以了解对所有管理域节点的更改，以使资源属性的值可在集群管理域中的各个节点上同步。同步行为取决于许多因素:

- 集群状态
- 集群管理域状态
- 节点状态
- 对资源执行的特定操作

要处理所有受监视资源项状态，请执行下列步骤：

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在“集群资源服务”页中，单击**使用管理域**以显示该集群中集群管理域的列表。
5. 在“管理域”页中，单击集群管理域旁的上下文图标，然后选择**使用受监视资源类型**。

注意：仅当所管理的节点是集群管理域的组成部分时，**使用受监视资源类型**操作才可用。将显示受监视资源类型的当前列表。

6. 在“使用受监视资源类型”页上，单击资源类型旁的上下文图标，然后选择**属性**。
7. 将显示“受监视资源项的属性列表”。“全局状态”列将显示活动集群管理域中此属性的当前状态。

这些值用于确定受监视资源在集群上的状态：

全局值 资源在所有管理域节点上应该使用的每个受监视属性的值。全局值在所有活动节点上相同，并且体现在域中已同步的最后一次更改。

全局状态

集群管理域上的资源的状态以及资源是否完全同步。可能的全局状态值如下：

一致 在集群管理域中的所有活动节点上，系统所监视的所有资源属性的值都相同。此状态出现在正常运营环境中，其中集群、集群管理域和所有节点都是可操作的，并且在集群中处于活动状态。在此环境中，对受监视资源的值的所有更改将传播至集群管理域中的所有其他节点。此处理与原始更改异步进行，但会使管理域上的已登记资源的值保持一致。在此情况下，全局状态为“一致”，并且会在每个节点上成功进行更改，而每个节点上的资源的值将与该资源的全局值相匹配。

不一致 在集群管理域中的所有活动节点上，系统所监视的所有资源属性的值并不相同。将记录用来描述状态为“不一致”的原因的消息。例如，如果在集群管理域处于不活动状态时更改受监视资源，那么受监视资源状态将为“不一致”。

暂挂 正在集群管理域中使受监视属性的值同步。

已添加 受监视资源已添加至集群管理域，但尚未同步。

已关闭 由于集群管理域已关闭，并且不再处理对资源所作的更改，因此受监视资源处于未知状态。集群管理域关闭后，当前设置为“一致”的所有 MRE 的全局状态将设置为“已关闭”。

已失败 集群管理域不再监视该资源，应该除去该 MRE。当集群管理域对资源进行同步时，建议您不要执行某些资源操作。如果 MRE 表示的资源是系统对象，应该先除去 MRE，然后才删除该资源、对其重命名或将其移至另一个库。如果某个资源被删除、重命名或移至另一个库，那么 MRE 的全局状态将是“已失败”，之后在任何节点上对该资源所作的任何更改都不会传播到集群管理域中的任何节点。

将受监视资源存储在集群管理域中的节点上时，会更改回受监视资源的值以匹配由集群管理域同步的全局值。

除去受监视资源项：

受监视资源项（MRE）是高可用性环境中当前正在使用并通过集群管理域监视更改的资源。当不再需要监视 MRE 时，可以将它们除去。可以使用“集群资源服务”图形界面来除去受监视资源项（MRE）。

要除去受监视资源项，请执行下列步骤：

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在“集群资源服务”页中，单击**使用管理域**以显示该集群中集群管理域的列表。
5. 在“管理域”页中，单击集群管理域旁的上下文图标，然后选择**使用受监视资源类型**。

注意：仅当所管理的节点是集群管理域的组成部分时，**使用受监视资源类型**操作才可用。将显示受监视资源类型的当前列表。

6. 在受监视资源类型列表中，单击受监视资源类型旁的上下文图标，然后选择**使用受监视资源项**。将显示 MRE 对象列表。
7. 单击要除去的 MRE 对象旁的上下文图标，然后选择**除去受监视资源项**。
8. 在“确认除去 MRE 对象”窗口中，单击**是**。将从集群管理域中除去该受监视资源项。

相关信息

除去管理域 MRE (RMVADMRE) 命令

除去受监视资源项 (QfpadRmvMonitoredResourceEntry) API

列示受监视资源类型:

受监视资源类型是可由集群管理域监视的系统对象，如用户概要文件和环境变量。可列示当前在集群管理域中指定的受监视资源的类型。

要列示受监视资源类型，请执行下列步骤:

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在“集群资源服务”页中，单击**使用管理域**以显示该集群中集群管理域的列表。
5. 在“管理域”页中，单击集群管理域旁的上下文图标，然后选择**使用受监视资源类型**。

注意：仅当所管理的节点是集群管理域的组成部分时，**使用受监视资源类型**操作才可用。将显示受监视资源类型的当前列表。

6. 对所选受监视资源类型执行必需的操作:

- 使用受监视资源属性
- 添加新的受监视资源项

l 列示受监视资源项:

l 受监视资源类型是已在集群管理域中定义的资源，如用户概要文件和环境变量。可使用“集群资源服务”图形界面来列示当前在集群管理域中定义的受监视资源项。

l 要列示受监视资源项，请执行下列步骤:

- l 1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
- l 2. 使用您的用户概要文件和密码登录系统。
- l 3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
- l 4. 在“集群资源服务”页中，单击**使用管理域**以显示该集群中集群管理域的列表。
- l 5. 在“管理域”页中，单击集群管理域旁的上下文图标，然后选择**使用受监视资源类型**。

- 注意：仅当所管理的节点是集群管理域的组成部分时，使用受监视资源类型操作才可用。将显示受监视资源类型的当前列表。
6. 在受监视资源类型列表中，单击受监视资源类型旁的上下文图标，然后选择使用受监视资源项。
 7. 查看并使用已登记受监视资源项列表。

选择要监视的属性:

添加受监视资源项后，可选择与该资源相关联的属性以便由集群管理域进行监视。

要选择针对受监视资源项（MRE）监视的属性，请执行下列步骤：

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在“集群资源服务”页中，单击**使用管理域**以显示该集群中集群管理域的列表。
5. 在“管理域”页中，单击集群管理域旁的上下文图标，然后选择**使用受监视资源类型**。

注意：仅当所管理的节点是集群管理域的组成部分时，使用受监视资源类型操作才可用。将显示受监视资源类型的当前列表。

6. 在受监视资源类型列表中，单击受监视资源类型旁的上下文图标，然后选择**使用受监视资源项...**。将显示 MRE 对象列表。
7. 单击 MRE 对象（如用户概要文件或系统值）旁的上下文图标，然后选择**使用属性**。将显示“MRE 属性列表”。
8. 在“MRE 属性列表”窗口中，选择要监视的属性，然后单击**关闭**。例如，如果要监视以太网线路描述以查看对其资源名称属性的更改，那么应选择资源名称作为属性。

相关任务

第 97 页的『添加受监视资源项』

可以将受监视资源项（MRE）添加至集群管理域。受监视资源项定义关键的资源，以使对这些资源进行的更改在整个高可用性环境中保持一致。

可以监视的属性:

对于不同类型的资源，可向集群管理域添加受监视资源项。本主题列示可监视的每种资源类型的属性。

资源类型

- 类 (*CLS)
- 以太网线路描述 (*ETHLIN)
- 独立磁盘池设备描述 (*ASPDEV)
- 作业描述 (*JOBDEV)
- 网络属性 (*NETA)
- 用于连接安全性的网络服务器配置 (*NWSCFG)
- 用于远程系统的网络服务器配置 (*NWSCFG)
- 用于服务处理器的网络服务器配置 (*NWSCFG)
- 用于 iSCSI 连接的网络服务器描述 (*NWSD)
- 用于集成网络服务器的网络服务器描述 (*NWSD)
- 网络服务器存储空间 (*NWSSTG)

- | • 网络服务器主机适配器设备描述 (*NWSHDEV)
- | • 光学设备描述 (*OPTDEV)
- | • 子系统描述 (*SBSD)
 - 系统环境变量 (*ENVVAR)
 - 系统值 (*SYSVAL)
- | • 磁带设备描述 (*TAPDEV)
- | • 令牌环线路描述 (*TRNLIN)
 - TCP/IP 属性 (*TCPA)
 - 用户概要文件 (*USRPRF)

表 15. 可针对类监视的属性

属性名	描述
CPUTIME	最长 CPU 时间
DFTWAIT	缺省等待时间
MAXTHD	最大线程数
MAXTMPSTG	最大临时存储空间
RUNPTY	运行优先级
TEXT	文本描述
TIMESLICE	时间片

表 16. 可针对线路描述监视的属性

属性名	描述
ASSOCPORT	相关联的端口资源名
AUTOCTRL	自动创建控制器
AUTODLCTRL	自动删除控制器
CMNRCYLMT	恢复限制
COSTBYTE	在线路上发送和接收数据的每字节相对成本
COSTCNN	在线路上进行连接的相对成本
DUPLEX	双工
GENTSTFRM	生成测试帧
GRPADR	组地址
LINESPEED	线路速度
MAXFRAME	最大帧大小
MAXCTL	最大控制器数
MSGQ	消息队列
ONLINE	在 IPL 时联机
PRPDLY	传播延迟
RSRCNAME	资源名
SECURITY	物理线路的安全级别
SSAP	源服务访问点 (SSAP) 信息列表
TEXT	文本描述
USRDFN1	第一个用户定义项

表 16. 可针对线路描述监视的属性 (续)

属性名	描述
USRDFN2	第二个用户定义项
USRDFN3	第三个用户定义项
VRYWAIT	联机等待

表 17. 可针对独立磁盘池设备描述监视的属性

属性名	描述
MSGQ	消息队列
RDB	关系数据库
RSRCNAME	资源名
TEXT	文本描述

表 18. 可针对作业描述监视的属性

属性名	描述
ACGCDE	记帐代码
ALWMLTTHD	允许多个线程
DDMCNV	DDM 对话
DEVRCYACN	设备恢复操作
ENDSEV	关闭严重性
HOLD	在作业队列中挂起
INLASGRP	初始 ASP 组
INQMSGRPY	查询消息应答
JOBMSGQFL	作业消息队列已满操作
JOBMSGQMX	作业消息队列最大大小
JOBPTY	在 JOBQ 中的作业优先级
JOBQ	作业队列
LOG	消息记录
LOGCLPGM	记录 CL 程序命令
OUTPTY	在 OUTQ 中的输出优先级
OUTQ	输出队列
PRTDEV	打印设备
PRTTXT	打印文本
RQSDTA	请求数据或命令
RTGDTA	路由数据
SPLFACN	假脱机文件操作
SWS	作业切换
SYNTAX	CL 语法检查
TEXT	文本描述
TSEPOOL	时间片结束池
USER	用户

表 19. 可针对网络属性监视的属性

属性名	描述
ALWADDCLU	允许添加至集群
DDMACC	DDM/DRDA 请求访问
NWSDOMAIN	网络服务器域
PCSACC	客户机请求访问
注意: 每个网络属性被视为它自己的受监视资源项。对于它们, 资源类型和属性名称是完全相同的。	

表 20. 可针对服务处理器的网络服务器配置监视的属性

属性名	描述
EID	机柜标识
ENBUNICAST	启用单点广播
INZSP	初始化服务处理器
SPAUT	服务处理器权限
SPCERTID	服务处理器证书标识
SPINTNETA	服务处理器因特网地址
SPNAME	服务处理器名称
TEXT	文本描述

表 21. 可针对远程系统的网络服务器配置监视的属性

属性名	描述
BOOTDEVID	引导设备标识
CHAPAUT	目标 CHAP 认证
DELIVERY	传送方法
DYNBOOTOPT	动态引导选项
INRCHAPAUT	发起方 CHAP 认证
RMTIFC	远程接口
RMTSYSID	远程系统标识
SPNWSCFG	用于管理远程服务器的服务处理器网络服务器配置
TEXT	文本描述

表 22. 可针对连接安全性的网络服务器配置监视的属性

属性名	描述
IPSECRULE	IP 安全性规则
TEXT	文本描述

表 23. 可针对集成网络服务器的网络服务器描述监视的属性

属性名	描述
CFGFILE	配置文件
CODEPAGE	表示要由此网络服务器使用的字符集的 ASCII 代码页
EVTLOG	事件记录
MSGQ	消息队列

表 23. 可针对集成网络服务器的网络服务器描述监视的属性 (续)

属性名	描述
NWSSTGL	存储空间链接
PRPDMNUSR	传播域用户
RSRCNAME	资源名
RSTDDEVRSC	受限设备资源
SHUTDTIMO	关闭超时
SYNCTIME	使日期和时间同步
TCPDMNNAME	TCP/IP 本地域名
TCPHOSTNAM	TCP/IP 主机名
TCPPORTCFG	TCP/IP 端口配置
TCPNAMSVR	TCP/IP 名称服务器系统
TEXT	文本描述
VRYWAIT	联机等待
WINDOWSNT	Windows® 网络服务器描述

表 24. 可针对 iSCSI 连接的网络服务器描述监视的属性

属性名	描述
ACTTMR	激活计时器
CFGFILE	配置文件
CMNMSGQ	通信消息队列
CODEPAGE	表示要由此网络服务器使用的字符集的 ASCII 代码页
DFTSECRULE	缺省 IP 安全规则
DFTSTGPTH	缺省存储路径
EVTLOG	事件记录
MLTPHGRP	多路径组
MSGQ	消息队列
NWSCFG	网络服务器配置
NWSSTGL	存储空间链接
PRPDMNUSR	传播域用户
RMVMEDPTH	可移动介质路径
RSRCNAME	资源名
RSTDDEVRSC	受限设备资源
SHUTDTIMO	关闭超时
STGPTH	网络服务器的 iSCSI 存储路径
SYNCTIME	使日期和时间同步
TCPDMNNAME	TCP/IP 本地域名
TCPHOSTNAM	TCP/IP 主机名
TCPNAMSVR	TCP/IP 名称服务器系统
TCPPORTCFG	TCP/IP 端口配置
TEXT	文本描述
VRTETHCTLP	虚拟以太网控制端口

表 24. 可针对 iSCSI 连接的网络服务器描述监视的属性 (续)

属性名	描述
VRTETHPTH	虚拟以太网路径
VRYWAIT	联机等待

表 25. 可针对网络服务器存储空间监视的属性

属性名	描述
SIZE	大小
TEXT	文本描述
TOTALFILES	总文件数

表 26. 可针对网络服务器主机适配器设备描述监视的属性

属性名	描述
CMNRCYLMT	恢复限制
LCLIFC	相关联的本地接口
MSGQ	消息队列
ONLINE	在 IPL 时联机
RSRCNAME	资源名
TEXT	文本描述

表 27. 可针对光学设备描述监视的属性

属性名	描述
MSGQ	消息队列
ONLINE	在 IPL 时联机
RSRCNAME	资源名
TEXT	文本描述

表 28. 可针对子系统描述监视的属性

属性名	描述
AJE	自动启动作业项
CMNE	在 IPL 时联机
JOBQE	作业队列
MAXJOBS	最大作业数
PJE	预启动作业项
RMTLOCNAME	远程位置名
RTGE	路由项
SGNDSPF	登录屏幕
SYSLIBLE	子系统库
TEXT	文本描述
WSNE	工作站名称项
WSTE	工作站类型项

表 29. 可针对系统环境变量监视的属性

可监视的任何 *SYS 级别环境变量。属性和资源名称都与环境变量的名称相同。
注意: 每个环境变量被视为它自己的受监视资源项。对于它们, 资源类型和属性名称是完全相同的。

表 30. 可针对系统值监视的属性

属性名	描述
QACGLVL	记帐级别
QACTJOBITP	允许中断作业
QALWOBJRST	阻止任何人恢复系统状态对象或沿用权限的对象
QALWUSRDMN	允许用户域对象
QASTLVL	辅助级别
QATNPGM	系统辅助程序
QAUDCTL	审计控制
QAUDENDACN	审计日志错误操作
QAUDFRCLVL	审计强制级别
QAUDLVL	审计级别
QAUDLVL2	审计级别扩展
QAUTOCFG	自动设备配置
QAUTORMT	远程控制器和设备
QAUTOVRT	自动虚拟设备配置
QCCSID	编码字符集标识
QCFGMSGQ	线路、控制器和设备的消息队列
QCHRID	用于显示或打印数据的缺省图形字符集和代码页
QCHRIDCTL	作业的字符标识控制
QCMNRCYLMT	自动通信错误恢复
QCNTRYID	国家或地区标识
QCRTAUT	新对象的权限
QCRTOBJAUD	审计新对象
QCTLSBSD	控制子系统或库
QCURSYM	货币符号
QDATFMT	日期格式
QDATSEP	日期分隔符
QDBRCVYWT	在完成重新启动前等待数据库恢复
QDECFMT	十进制格式
QDEVNAMING	设备命名约定
QDEVRCYACN	设备恢复操作
QDSCJOBITV	断开连接的作业的超时时间间隔
QDSPSGNINF	控制登录信息的显示
QENDJOB LMT	立即关闭的最长时间
QFRCCVNRST	恢复时强制转换
QHSTLOGSIZ	历史记录文件大小
QIGCCDEFNT	编码字体名称

表 30. 可针对系统值监视的属性 (续)

属性名	描述
QIGCFNTSIZ	编码字体磅值大小
QINACTITV	不活动作业超时时间间隔
QINACTMSGQ	超时时间间隔操作
QIPLTYPE	重新启动类型
QJOBMSGQFL	作业消息队列已满操作
QJOBMSGQMX	作业消息队列最大大小
QJOBMSGQSZ	作业消息队列的初始大小, 以千字节计 (KB)
QJOBMSGQTL	作业消息队列的最大大小 (以 KB 计)
QJOBSPLA	作业的假脱机控制块的初始大小 (以字节计)
QKBDBUF	键盘缓冲区
QKBDTYPE	键盘语言字符集
QLANGID	缺省语言标识
QLIBLCKLVL	用户作业库搜索列表中的锁定库
QLMTDEVSSN	限制设备会话
QLMTSECOFR	限制安全主管设备访问
QLOCALE	语言环境
QLOGOUTPUT	生成作业记录的打印机输出
QMAXACTLVL	系统的最大活动级别
QMAXJOB	系统上允许的最大作业数
QMAXSGNACN	达到 QMAXSIGN 系统值使用的限制时的系统响应
QMAXSIGN	允许的最多无效登录尝试次数
QMAXSPLF	最大打印机输出文件数
QMLTTHDACN	多线程作业中的函数并非线程安全时
QPASTHRSVR	可用的显示站联通服务器作业
QPRBFTR	问题记录过滤器
QPRBHLDITV	最短保留时间
QPRTDEV	缺省打印机
QPRTKEYFMT	打印密钥格式
QPRTTXT	最多可在列表和分隔符页底部打印 30 个字符的文本
QPWDCHGBLK	更改密码的最短时间间隔
QPWDEXPITV	密码有效天数
QPWDEXPWRN	密码到期时间间隔系统
QPWDLMTACJ	限制在密码中使用相邻的数字
QPWDLMTCHR	限制在密码中使用某些字符
QPWDLMTREP	限制在密码中使用重复的字符
QPWDLVL	密码级别
QPWDMAXLEN	密码中的最大字符数
QPWDMINLEN	密码中的最小字符数
QPWDPOSDIF	控制新密码中的字符的位置
QPWDRQDDGT	在新密码中需要数字

表 30. 可针对系统值监视的属性 (续)

属性名	描述
QPWDRQDDIF	控制新密码是否必须与先前的密码不同
QPWDRULES	密码规则
QPWDVLDPGM	密码批准程序
QPWRDWNLMT	立即关闭的最长时间
QRCLSPLSTG	自动清除未使用的打印机输出存储空间
QRETSVRSEC	保留服务器安全性数据指示符
QRMTSIGN	远程登录
QRMTSRVATR	远程服务属性
QSCANFS	扫描文件系统
QSCANFSCTL	扫描控制
QSCPFCONS	发生控制台问题
QSECURITY	系统安全性级别
QSETJOBATR	设置作业属性
QSFWERRLOG	软件错误记录
QSHRMEMCTL	允许使用具有写入功能的共享或映射内存
QSPCENV	缺省用户环境
QSPLFACN	假脱机文件操作
QSRTSEQ	排序顺序
QSRVDMP	未监视的脱离消息的服务记录
QSSLCSL	安全套接字层加密规范列表
QSSLCSLCTL	安全套接字层加密控制
QSSLPCL	安全套接字层协议
QSTRUPPGM	设置启动程序
QSTSMMSG	显示状态消息
QSYSLIBL	系统库列表
QTIMSEP	时间分隔符
QTSEPOOL	指示交互式作业在到达时间片结尾时是否应移至另一主存储池
注意: 每个系统值被视为它自己的受监视资源项。对于它们, 资源类型和属性名称是完全相同的。	

表 31. 可针对磁带设备描述监视的属性

属性名	描述
ASSIGN	在联机时指定设备
MSGQ	消息队列
ONLINE	在 IPL 时联机
RSRCNAME	资源名
TEXT	文本描述
UNLOAD	在脱机时卸装设备

表 32. 可针对令牌环描述监视的属性

属性名	描述
ACTLANMGR	激活 LAN 管理器
ADPTADR	本地适配器地址
AUTOCRTCTL	自动创建控制器
AUTODLTCTL	自动删除控制器
CMNRCYLMT	恢复限制
COSTBYTE	在线路上发送和接收数据的每字节相对成本
COSTCNN	在线路上进行连接的相对成本
DUPLEX	双工
ELYTKNRLS	早期令牌释放
FCNADR	工作地址
LINESPEED	线路速度
LINKSPEED	链路速度
LOGCFGCHG	记录配置更改
MAXCTL	最大控制器数
MAXFRAME	最大帧大小
MSGQ	消息队列
ONLINE	在 IPL 时联机
PRPDLY	传播延迟
RSRCNAME	资源名
SECURITY	线路的安全性
SSAP	源服务访问点 (SSAP) 信息列表
TRNINFBND	信标的令牌环通知
TRNLOGLVL	TRLAN 管理器记录级别
TRNMGRMODE	TRLAN 管理器方式
TEXT	令牌环线路的文本描述
USRDFN1	第一个用户定义项
USRDFN2	第二个用户定义项
USRDFN3	第三个用户定义项
VRYWAIT	联机等待

表 33. 可针对 TCP/IP 属性监视的属性

属性名	描述
ARPTIMO	地址解析协议 (ARP) 高速缓存超时
ECN	启用显式拥塞通知 (ECN)
IPDEADGATE	IP 失效网关检测
IPDTGFWD	IP 数据报转发
IPPATHMTU	路径最大传输单元 (MTU) 发现
IPQOSBCH	IP QoS 数据报批处理
IPQOSENB	IP QoS 启用
IPQOSTMR	IP QoS 计时器分辨率

表 33. 可针对 TCP/IP 属性监视的属性 (续)

属性名	描述
IPRSBTIMO	IP 重新装配超时
IPSRCRTG	IP 源路由
IPTTL	IP 生存时间 (中继段限制)
LOGPCLERR	Log 协议错误
NFC	网络文件高速缓存
TCPCLOTIMO	TCP 等待时间超时
TCPCNNMSG	TCP 关闭连接消息
TCPKEEPALV	TCP 保持活动
TCPMINRTM	TCP 最短重新传输时间
TCPR1CNT	TCP R1 重新传输计数
TCPR2CNT	TCP R2 重新传输计数
TCPRCVBUF	TCP 接收缓冲区大小
TCPSNDBUF	TCP 发送缓冲区大小
TCPURGPTR	TCP 紧急指针
UDPCKS	UDP 校验和
注意: 每个 TCP/IP 属性被视为它自己的受监视资源项。对于它们, 资源类型和属性名称是完全相同的。	

表 34. 可针对用户概要文件监视的属性

属性名	描述
ACGCDE	记帐代码
ASTLVL	辅助级别
ATNPGM	系统辅助程序
CCSID	编码字符集标识
CHRIDCTL	字符标识控制
CNTRYID	国家或地区标识
CURLIB	当前库
DLVRY	传送
DSPSGNINF	显示登录信息
GID	组标识编号
GRPAUT	组权限
GRPAUTYP	组权限类型
GRPPRF	组概要文件
HOMEDIR	主目录
INLMNU	初始菜单
INLPGM	要调用的初始程序
JOB	作业描述
KBDBUF	键盘缓冲
LANGID	语言标识
LCLPDMGT	本地密码管理
LMTCPB	限制功能

表 34. 可针对用户概要文件监视的属性 (续)

属性名	描述
LMTDEVSSN	限制设备会话
LOCALE	语言环境
MAXSTG	允许的最大存储空间
MSGQ	消息队列
OUTQ	输出队列
OWNER	所有者
PASSWORD	用户密码
PRTDEV	打印设备
PTYLMT	最高调度优先级
PWDEXP	将密码设置为已到期
PWDEXPITV	密码到期时间间隔
SETJOBATR	语言环境作业属性
SEV	严重性代码过滤器
SPCAUT	特权
SPCENV	特殊环境
SRTSEQ	排序顺序
STATUS	状态
SUPGRPPRF	补充组
TEXT	文本描述
UID	用户标识编号
USRCLS	用户类
USROPT	用户选项

显示受监视资源项消息:

可使用“集群资源服务”图形界面来显示与受监视资源项相关联的消息。

要显示和查看受监视资源项消息，请执行下列步骤:

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在“集群资源服务”页中，单击**使用管理域**以显示该集群中集群管理域的列表。
5. 在“管理域”页中，单击集群管理域旁的上下文图标，然后选择**使用受监视资源类型**。

注意: 仅当所管理的节点是集群管理域的组成部分时，**使用受监视资源类型**操作才可用。将显示受监视资源类型的当前列表。

6. 在受监视资源类型列表中，单击受监视资源类型旁的上下文图标，然后选择**使用受监视资源项**。将显示 MRE 对象列表。
7. 单击 MRE 对象（如用户概要文件或系统值）旁的上下文图标，然后选择**查看消息**。将显示“MRE 属性列表”。

管理切换式磁盘

切换式磁盘是作为设备集群资源组（CRG）的组成部分配置的独立磁盘池。切换式磁盘中存储的数据和应用程序的所有权可以切换至设备 CRG 中定义的其他系统。切换式磁盘技术能够在计划内和某些计划外系统运行中断期间提供高可用性。

使磁盘池变为不可用状态

可选择一個獨立磁盤池以使其變為不可用狀態（使其脫機）。直到該獨立磁盤池再次變為可用狀態（使其聯機），您才能訪問該獨立磁盤池中的任何磁盤機或對象或其相應數據庫。可在同一系統上使該磁盤池再次變為可用狀態，或在集群資源組恢復域中的另一系統上使其再次變為可用狀態。

要点： 当所有作业都不能在磁盘池上挂起保留项时，才能使独立磁盘池变为不可用状态。请参阅“在独立磁盘池上释放作业保留项”，以了解有关确定作业是否在使用独立磁盘池以及如何释放作业保留项的信息。

使用 System i 导航器来使 UDFS 磁盘池变为不可用状态时，字符界面中可能会产生需要响应的消息。System i 导航器不会提供任何指示信息来说明消息在等待响应。

要使独立磁盘池变为不可用状态，请执行下列步骤：

1. 在 System i 导航器中，展开**我的连接**（或您的活动环境）。
2. 展开要检查的 System i，然后选择**配置与服务** → **硬件** → **磁盘机**。
3. 选择**磁盘池**。
4. 右键单击要使其变为不可用状态的磁盘池，然后选择**使其变为不可用状态**。
5. 在显示的对话框中，单击**使其变为不可用状态**以使该磁盘池变为不可用状态。

可在字符界面中使用“联机配置”（VRYCFG）命令来使该磁盘池变为不可用状态。

使用“显示 ASP 状态”（DSPASPSTS）命令来标出正在执行的步骤。

使用“控制 ASP 访问”（QYASPCTLAA）API 来限制可访问 ASP 的进程。

使用“启动 DASD 管理操作”（QYASSDMO）API 来缩短使磁盘池变为不可用状态时所花的时间。

使硬件可切换

在 i5/OS 高可用性环境中，必须使外部扩展部件可切换。

在可切换环境中使用独立磁盘池时，相关联的硬件必须也有权进行切换。根据环境的不同，这些硬件可能包括机架或部件或者 IOP 及其相关资源。请参阅下列适用于可切换环境的步骤。

使机架或部件可切换

独立磁盘池可以包含多个扩展部件中的磁盘机。如果独立扩展部件包含某个独立磁盘池中包括的磁盘机，那么必须对该扩展部件进行授权，以允许其他系统对其进行访问。这称为使扩展部件可切换。如果您不希望其他系统能够访问该独立扩展部件，那么必须使该扩展部件成为私有扩展部件。

要使机架或部件可切换，请执行下列步骤：

1. 在 System i 导航器中，展开**我的连接**（或您的活动环境）。
2. 展开要检查的系统，然后展开**配置与服务** → **硬件** → **磁盘机** → **按位置**，然后选择应该可切换的机架或磁盘机。
3. 右键单击突出显示的机架或磁盘机，然后选择**设置为可切换**。
4. 按照所显示的对话框中的指示信息执行操作。

使 IOP 可切换

要使 IOP 可切换，包含该 IOP 的总线（此总线控制所要切换的磁盘机）必须由主节点拥有（拥有并共享）。备份节点还必须使用该总线（使用共享总线）。有关更多信息，请参阅“在分区之间动态切换 IOP”。

要完成此任务，您需要对专用服务工具中的“系统分区”功能具有管理权的“服务工具”用户概要文件。有关获取逻辑分区特权的更多信息，请参阅“逻辑分区权限”。

要使用“管理中心”来更改总线的所有权类型，请执行下列步骤：

1. 在 System i 导航器中，展开**我的连接**（或您的活动环境）。
2. 选择系统的主分区。
3. 展开**配置与服务**，然后选择**逻辑分区**。
4. 右键单击**逻辑分区**，然后选择**配置分区**。
5. 在“配置逻辑分区”窗口中，右键单击要更改所有权的总线，然后选择**属性**。
6. 选择分区页。
7. 从**拥有逻辑分区**中选择拥有该总线的分区，然后从**共享**中选择所有权类型。如果所有权类型为“共享”，那么共享该总线的分区将显示在列表中。如果您需要更多有关这些选项的信息，请单击“帮助”。
8. 单击**确定**。

使用硬件管理控制台使 I/O 池可切换

如果正在使用硬件管理控制台来管理逻辑分区，那么必须创建包含 IOP、IOA 和所有相连资源的 I/O 池，以使独立磁盘池能够在分区之间进行切换。您必须分配每个分区概要文件中的 I/O 池，从而授权访问每个要拥有该独立磁盘池的分区。

要创建可以在分区之间进行切换的 I/O 池，请执行下列步骤：

1. 打开“逻辑分区概要文件属性”窗口，以更改分区概要文件属性以及将资源分配到 I/O 池。
2. 单击**物理 I/O** 选项卡。
3. 在“概要文件 I/O 设备”列中，展开包含要变为可切换的 IOP 的总线。
4. 选择要分配给 I/O 池的 IOP。该 IOP 必须是**期望的**（**必需**列未包含选中标记）。
5. 单击 I/O 池列，以使光标转到要分配到 I/O 池的 IOP 所在的行，然后输入 I/O 池的编号。
6. 重复这些步骤，以便将该 IOP 控制的每个 IOA 和资源添加到 I/O 池。
7. 单击**确定**。

使 I/O 池与分区相关联

将资源添加至 I/O 池后，请完成下列步骤，以使该 I/O 池与每个应该能够拥有可切换环境中独立磁盘池的附加分区相关联。

1. 打开“逻辑分区概要文件属性”窗口，以更改每个需要访问该独立磁盘池的附加分区的分区概要文件属性。
2. 单击**物理 I/O** 选项卡。
3. 单击**高级**。
4. 在“I/O 池”窗口的**要添加的 I/O 池**字段中，输入 I/O 池的编号，随独立磁盘池一起切换的资源将分配到该 I/O 池。
5. 单击**添加** → **确定**。

要使 I/O 池更改生效，请对每个已更改分区概要文件的分区完成下列步骤：

1. 关闭该分区。请参阅“重新启动和关闭逻辑分区中的 i5/OS”。
2. 通过激活分区概要文件启动该逻辑分区，以使更改生效。

相关概念

在分区间动态切换 IOP

逻辑分区权限

I/O 池

相关任务

更改分区概要文件属性

激活分区概要文件

在逻辑分区中重新启动并关闭 i5/OS™。

1 停顿独立磁盘池

在 i5/OS 高可用性解决方案中，独立磁盘池用于存储弹性数据和应用程序。一些系统功能（如执行备份）要求您在操作进行时临时暂挂对该数据的更改。

要缩短停顿独立磁盘池所花的时间，您可能需要挂起批处理作业队列，停止某些子系统，或向交互式用户发送中断消息以通知他们将新任务延后。

要停止独立磁盘池，请执行下列步骤。

在命令行界面中，输入以下命令：`CHGASPACT ASPDEV(name) OPTION(*SUSPEND) SSPTIMO(30) SSPTIMOACN(*CONT)`，其中 *name* 是要暂挂的独立磁盘池的名称。在此命令中，您将指定暂挂独立磁盘池并设置 30 秒超时，然后继续完成下一步，即使已超过超时限制。

恢复独立磁盘池

在 i5/OS 高可用性环境中停顿独立磁盘池以执行备份操作后，需要恢复独立磁盘池以确保已更新停顿期间对数据所作的更改。

完成下列步骤以恢复独立磁盘池：

在命令行界面中，输入以下命令：`CHGASPACT ASPDEV(name) OPTION(*RESUME)`，其中 *name* 是要恢复的独立磁盘池的名称。

管理跨站点镜像

您可以管理三种跨站点镜像技术：地理镜像、城域镜像和全局镜像。这些跨站点镜像技术将关键数据从生产站点的磁盘机复制至备份位置的磁盘机，从而提供灾难恢复功能。

管理地理镜像

请使用以下信息来帮助您管理地理镜像。地理镜像是跨站点镜像的子功能，用于将数据镜像到 i5/OS 环境中的独立磁盘池。

暂挂地理镜像：

如果因为任何原因需要关闭 TCP 通信（例如，将系统置于受限状态），应先暂挂地理镜像。此操作会暂时停止高可用性解决方案中各个系统间的镜像。

暂挂镜像时，对独立磁盘池的生产副本所作的任何更改不会传输至镜像副本。

注：恢复地理镜像时，生产副本与镜像副本之间需要同步。如果已暂挂地理镜像而未进行跟踪，那么会进行完全同步。这可能需要花很长时间。

使用 IBM Systems Director Navigator for i5/OS 暂挂地理镜像

要使用 IBM Systems Director Navigator for i5/OS 暂挂地理镜像，请执行下列步骤：

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**配置与服务**。
4. 选择**磁盘池**。
5. 选择要暂挂的**磁盘池**生产副本。
6. 从**选择操作**菜单中选择**会话**。
7. 选择要暂挂的会话。
8. 从**选择操作**菜单中选择**暂挂并进行跟踪**或**暂挂而不进行跟踪**。

使用 System i 导航器暂挂地理镜像

要使用 System i 导航器暂挂地理镜像，请执行下列步骤：

1. 在 System i 导航器中，展开**我的连接**（或您的活动环境）。
2. 展开拥有地理镜像的磁盘池生产副本的系统，您想要对该副本暂挂地理镜像。
3. 展开**配置与服务** → **硬件** → **磁盘机** → **磁盘池**。
4. 右键单击要暂挂的**磁盘池**生产副本，然后选择**地理镜像** → **暂挂地理镜像**。

如果选择暂挂并进行跟踪，那么系统会尝试跟踪对这些磁盘池所作的更改。通过在恢复地理镜像时执行部分同步，这会缩短同步过程所花的时间。如果跟踪空间不足，那么在恢复地理镜像时需要完全同步。

注：如果选择暂挂地理镜像而不跟踪更改，那么恢复地理镜像时需要在生产副本与镜像副本之间进行完全同步。如果选择暂挂地理镜像并跟踪更改，那么只需要部分同步。完全同步可能需要花很长时间，有时可能要花一个小时到几个小时甚至更长时间。同步所需时间取决于要同步的数据量、TCP/IP 连接的速度以及用于地理镜像的线路数目。

恢复地理镜像：

如果暂挂地理镜像，那么必须恢复它才能再次重新激活生产副本与镜像副本之间的镜像。

注：恢复地理镜像时，生产副本与镜像副本会通过执行地理镜像并行同步。同步可能需要花很长时间。如果变为不可用状态的磁盘池中中断同步，那么同步会在磁盘池再次变为可用时从中断处继续。中断的同步继续时，第一条消息（CPI0985D）声明同步完成 0%。

使用 IBM Systems Director Navigator for i5/OS 恢复地理镜像

要使用 IBM Systems Director Navigator for i5/OS 恢复地理镜像，请执行下列步骤：

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**配置与服务**。
4. 选择**磁盘池**。
5. 选择要恢复的**磁盘池**生产副本。

- | 6. 从**选择操作**菜单中选择**会话**。
- | 7. 选择要恢复的会话。
- | 8. 从**选择操作**菜单中选择**恢复**。

| 使用 **System i** 导航器恢复地理镜像

- | 要使用 System i 导航器恢复地理镜像，请执行下列步骤：
 1. 在 System i 导航器中，展开**我的连接**（或您的活动环境）。
 2. 展开拥有磁盘池生产副本的系统，您想要对该副本恢复地理镜像。
 3. 展开**配置与服务** → **硬件** → **磁盘机** → **磁盘池**。
 4. 右键单击要恢复的**磁盘池**生产副本，然后选择**地理镜像** → **恢复地理镜像**。

使用“启动 DASD 管理操作”（QYASSDMO）API 来缩短使磁盘池变为不可用状态时所花的时间。

拆离镜像副本：

如果正在使用地理镜像，并且要访问镜像副本以执行保存操作或数据挖掘操作，或者要创建报告，那么必须将镜像副本与生产副本拆离。

要将镜像副本拆离，请访问磁盘池的生产副本。

注：重新连接已拆离的镜像副本时，需要在生产副本与镜像副本之间进行完全同步。同步可能需要花很长时间。

| 使用 **IBM Systems Director Navigator for i5/OS** 拆离镜像副本

- | 要使用 IBM Systems Director Navigator for i5/OS 来拆离镜像副本，请执行下列步骤：
 - | 1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
 - | 2. 使用您的用户概要文件和密码登录系统。
 - | 3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**配置与服务**。
 - | 4. 选择**磁盘池**。
 - | 5. 选择所要拆离的**磁盘池**的生产副本。
 - | 6. 从**选择操作**菜单中选择**会话**。
 - | 7. 选择要拆离的会话。
 - | 8. 从**选择操作**菜单中选择**拆离并跟踪**或**拆离而不跟踪**。

| 使用 **System i** 导航器拆离镜像副本

建议您使独立磁盘池变为不可用状态，以确保执行拆离期间生产副本不会被更改。

- | 要使用 System i 导航器来拆离镜像副本，请执行下列步骤：
 1. 在 System i 导航器中，展开**我的连接**（或您的活动环境）。
 2. 展开要与镜像副本拆离的磁盘池生产副本所属的系统。
 3. 展开**配置与服务** → **硬件** → **磁盘机** → **磁盘池**。
 4. 右键单击所要拆离的**磁盘池**生产副本，然后选择**地理镜像** → **拆离镜像副本**。

如果**地理镜像** → **拆离镜像副本**处于禁用状态，导致您无法单击此选项，那么表示该镜像副本与生产副本不同步。在这种情况下，必须恢复地理镜像，将磁盘池联机并对生产副本和镜像副本进行同步，然后才能拆离镜像副本。

在使拆离的镜像副本变为可用状态之前，应该为独立磁盘池创建第二个唯一的设备描述，以便将其与生产副本区分开。镜像副本的独立设备描述用于防止网络中存在同一个数据库的两个实例。这还可以简化 System i 导航器外部的的工作。请使用所拆离的镜像副本的设备描述使该镜像副本可用。

重新连接镜像副本:

如果已拆离镜像副本并且已处理完拆离的镜像副本，那么必须重新连接已拆离的镜像副本才能继续使用地理镜像。

通过访问磁盘池的生产副本来重新连接已拆离的镜像副本。将已拆离的镜像副本重新连接至生产副本时，已拆离的镜像副本必须处于不可用状态。

注: 重新连接已拆离的镜像副本时，生产副本与镜像副本之间需要完全同步。同步可能需要花很长时间。

使用 IBM Systems Director Navigator for i5/OS 重新连接镜像副本

要使用 IBM Systems Director Navigator for i5/OS 重新连接镜像副本，请执行下列步骤:

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**配置与服务**。
4. 选择**磁盘池**。
5. 选择要暂挂的**磁盘池**生产副本。
6. 从**选择操作**菜单中选择**会话**。
7. 选择要暂挂的会话。
8. 从**选择操作**菜单中选择**连接**。

使用 System i 导航器重新连接镜像副本

要使用 System i 导航器重新连接镜像副本，请执行下列步骤:

1. 在 System i 导航器中，展开**我的连接**（或您的活动环境）。
2. 展开拥有磁盘池生产副本的系统，您想要将已拆离的镜像副本重新连接至该生产副本。
3. 展开**配置与服务** → **硬件** → **磁盘机** → **磁盘池**。
4. 右键单击要重新连接的**磁盘池**生产副本，然后选择**地理镜像** → **重新连接镜像副本**。

取消配置地理镜像:

如果您不再希望能够对特定磁盘池或磁盘池组使用地理镜像，那么可以选择**取消配置地理镜像**。如果取消配置地理镜像，那么系统将停止地理镜像并删除镜像副本站点中各个节点上的磁盘池镜像副本。

要取消配置地理镜像，磁盘池必须处于脱机状态。

要取消配置地理镜像，请执行下列步骤:

1. 在 System i 导航器中，展开**我的连接**（或您的活动环境）。
2. 展开要检查的系统，然后展开**配置与服务** → **硬件** → **磁盘机** → **磁盘池**。
3. 右键单击要取消配置的**磁盘池**的生产副本，然后选择**地理镜像** → **取消配置地理镜像**。

4. 更新集群配置，如下所示：
 - a. 从设备集群资源组（CRG）恢复域中除去与该镜像副本相关联的节点。
 - b. 从集群的其余节点中除去该站点名和数据端口 IP 地址。

相关任务

第 114 页的『除去节点』

如果正在对节点执行升级，或者该节点不再需要参与 i5/OS 高可用性环境，那么可能需要从集群中除去该节点。

更改地理镜像属性:

您可以更改与地理镜像相关联的信息以及编辑相关联的副本描述。

使用 IBM Systems Director Navigator for i5/OS 更改地理镜像属性

要使用 IBM Systems Director Navigator for i5/OS 来编辑地理镜像会话，请执行下列步骤:

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**配置与服务**。
4. 选择**磁盘池**。
5. 选择与会话相关联的磁盘池。
6. 从**选择操作**菜单中选择**会话**。
7. 选择该会话。
8. 从**选择操作**菜单中选择**属性**。要更改相关联的副本描述，请选择副本描述，然后单击**编辑**。

使用 System i 导航器更改地理镜像属性

要使用 System i 导航器来更改地理镜像属性，请执行下列步骤:

1. 在 System i 导航器中，展开**我的连接**（或您的活动环境）。
2. 展开要编辑属性的地理镜像会话的相关地理镜像磁盘池生产副本所属的系统，然后展开**配置与服务** → **硬件** → **磁盘机** → **磁盘池**。
3. 右键单击要编辑属性的**磁盘池**生产副本，然后选择**会话** → **打开**。
4. 右键单击要编辑属性的**会话**生产副本，然后选择**属性**。要更改相关联的副本描述，请选择副本描述，然后单击**编辑**。

管理城域镜像会话

在使用 IBM System Storage 城域镜像技术的 i5/OS 高可用性环境中，必须在 i5/OS 系统与配置了城域镜像的外部磁盘机之间配置全局镜像会话。从该系统，可以管理这些会话。

暂挂城域镜像会话:

可能需要暂挂城域镜像会话以便在系统上执行维护。

要暂挂城域镜像会话，请执行下列步骤:

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**配置与服务**。

4. 选择**磁盘池**。
5. 选择要暂挂的磁盘池。
6. 从**选择操作**菜单中选择**会话**。
7. 选择要暂挂的会话。
8. 从**选择操作**菜单中选择**暂挂**。

恢复城域镜像会话:

完成例行操作（如在系统上执行维护）后，需要恢复暂挂的城域镜像会话以重新启用高可用性。

要恢复暂挂的城域镜像会话，请执行下列步骤:

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**配置与服务**。
4. 选择**磁盘池**。
5. 选择暂挂的磁盘池。
6. 从**选择操作**菜单中选择**会话**。
7. 选择暂挂的会话。
8. 从**选择操作**菜单中选择**恢复**。

删除城域镜像会话:

可以删除城域镜像会话，以便不再将该会话用于高可用性和灾难恢复功能。

要删除城域镜像会话，请执行下列步骤:

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**配置与服务**。
4. 选择**磁盘池**。
5. 选择与所要删除的会话相关联的磁盘池。
6. 从**选择操作**菜单中选择**会话**。
7. 选择要删除的会话。
8. 从**选择操作**菜单中选择**删除**。

显示城域镜像属性:

显示有关城域镜像会话的信息，以更改相关联的副本描述。

要使用 IBM Systems Director Navigator for i5/OS 来更改城域镜像属性，请执行下列步骤:

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**配置与服务**。
4. 选择**磁盘池**。
5. 选择与会话相关联的磁盘池。
6. 从**选择操作**菜单中选择**会话**。

7. 选择该会话。
8. 从**选择操作**菜单中选择**属性**。要更改相关联的副本描述，请选择副本描述，然后单击**编辑**。

管理全局镜像

在使用 IBM System Storage 全局镜像技术的 i5/OS 高可用性环境中，必须在 i5/OS 系统与配置了全局镜像的外部磁盘机之间配置全局镜像会话。从该系统，可以管理这些会话。

暂挂全局镜像会话:

可能需要暂挂全局镜像会话以便在系统上执行维护。

要暂挂全局镜像会话，请执行下列步骤:

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**配置与服务**。
4. 选择**磁盘池**。
5. 选择要暂挂的磁盘池。
6. 从**选择操作**菜单中选择**会话**。
7. 选择要暂挂的会话。
8. 从**选择操作**菜单中选择**暂挂**。

恢复全局镜像会话:

完成例行操作（如在系统上执行维护）后，需要恢复暂挂的全局镜像会话以重新启用高可用性。

要恢复暂挂的全局镜像会话，请执行下列步骤:

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**配置与服务**。
4. 选择**磁盘池**。
5. 选择暂挂的磁盘池。
6. 从**选择操作**菜单中选择**会话**。
7. 选择暂挂的会话。
8. 从**选择操作**菜单中选择**恢复**。

删除全局镜像会话:

可以删除全局镜像会话，以便不再将该会话用于高可用性和灾难恢复功能。

要删除全局镜像会话，请执行下列步骤:

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**配置与服务**。
4. 选择**磁盘池**。
5. 选择与所要删除的会话相关联的磁盘池。
6. 从**选择操作**菜单中选择**会话**。

7. 选择要删除的会话。
8. 从**选择操作**菜单中选择**删除**。

更改全局镜像会话属性:

显示有关全局镜像会话的信息，以更改相关联的副本描述。

要使用 IBM Systems Director Navigator for i5/OS 来更改全局镜像属性，请执行下列步骤:

1. 在 Web 浏览器中输入 <http://mysystem:2001>，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**配置与服务**。
4. 选择**磁盘池**。
5. 选择与会话相关联的磁盘池。
6. 从**选择操作**菜单中选择**会话**。
7. 选择该会话。
8. 从**选择操作**菜单中选择**属性**。要更改相关联的副本描述，请选择副本描述，然后单击**编辑**。

管理 FlashCopy

FlashCopy 是一种 IBM System Storage 技术，它允许对外部磁盘机进行时间点复制。在使用城域镜像或全局镜像的 i5/OS 高可用性解决方案中，通过使用 FlashCopy 创建数据副本，然后将该副本备份至介质，可以缩小备份窗口。要使用 FlashCopy，必须在系统与外部存储器部件之间创建会话。

配置 FlashCopy 会话

对于使用 IBM System Storage 技术的 i5/OS 高可用性环境而言，您可以配置 FlashCopy 会话以创建数据的时间点副本。

有关在 IBM System Storage DS6000 上使用 FlashCopy 的信息，请参阅 IBM System Storage DS6000 信息中心。有关在 IBM System Storage DS8000 上使用 FlashCopy 的信息，请参阅 IBM System Storage DS8000 信息中心。

要配置 FlashCopy 会话，请执行下列步骤:

1. 在 Web 浏览器中输入 <http://mysystem:2001>，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**配置与服务**。
4. 选择**磁盘池**。
5. 选择要用作源副本的磁盘池。
6. 从**选择操作**菜单中选择**新建会话**。
7. 按照向导的指示信息完成任务。

更新 FlashCopy

在对 IBM System Storage 外部存储器部件上的 FlashCopy 卷进行再同步时，可以更新 FlashCopy 会话。再同步操作允许您在不重新复制整个卷的情况下创建副本。只有在存在持久关系的情况下才有可能执行此过程，在这种情况下，存储器部件将持续不断地跟踪源卷和目标卷的更新。持久关系意味着，源卷与目标卷之间的关系在后台复制完成后继续保持。在 i5/OS 上创建的 FlashCopy 会话使您能够管理和监视 IBM System Storage 部件上与 FlashCopy 相关的活动。

要更新 FlashCopy 会话，请执行下列步骤：

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**配置与服务**。
4. 选择**磁盘池**。
5. 选择与所要更新的会话相关联的磁盘池。
6. 从**选择操作**菜单中选择**会话**。
7. 选择要更新的会话。
8. 从**选择操作**菜单中选择**更新 FlashCopy**。

重新连接 FlashCopy

重新连接 FlashCopy 会话。

要重新连接 FlashCopy 会话，请执行下列步骤：

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**配置与服务**。
4. 选择**磁盘池**。
5. 选择与要重新连接的会话相关联的磁盘池。
6. 从**选择操作**菜单中选择**会话**。
7. 选择要重新连接的会话。
8. 从**选择操作**菜单中选择**重新连接 FlashCopy**。

拆离 FlashCopy

可以将所选 FlashCopy 会话的目标卷与源卷拆离。

要将所选 FlashCopy 会话的目标卷与源卷拆离，请执行下列步骤：

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**配置与服务**。
4. 选择**磁盘池**。
5. 选择与所要拆离的会话相关联的磁盘池。
6. 从**选择操作**菜单中选择**会话**。
7. 选择要将目标卷与源卷拆离的会话。
8. 从**选择操作**菜单中选择**拆离 FlashCopy**。

删除 FlashCopy

删除 FlashCopy 会话。

要删除 FlashCopy 会话，请执行下列步骤：

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**配置与服务**。

4. 选择**磁盘池**。
5. 选择与所要删除的会话相关联的磁盘池。
6. 从**选择操作**菜单中选择**会话**。
7. 选择要删除的会话。
8. 从**选择操作**菜单中选择**删除**。

通过 FlashCopy 恢复数据

在 IBM System Storage 单元上完成 FlashCopy 后，可在数据源副本中断时将目标卷上的数据恢复至源卷。为此，需要逆向执行 i5/OS 上创建的 FlashCopy 会话。但是，逆向执行该会话会将目标上的数据复制回源并使源返回至其较早版本。

警告： 逆向执行 FlashCopy 会话通过将目标上的数据复制回源来回退在源副本上所作的更改。这会使源返回至较早时间点的状态。

要逆向执行 FlashCopy 会话，请执行下列步骤：

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**配置与服务**。
4. 选择**磁盘池**。
5. 选择源副本的磁盘池。
6. 从**选择操作**菜单中选择**打开会话**。
7. 选择该会话。
8. 从**选择操作**菜单中选择**逆向执行 FlashCopy**。

更改 FlashCopy 属性

显示有关 FlashCopy 会话的信息，以更改相关联的副本描述。

要更改有关 FlashCopy 会话的信息，请执行下列步骤：

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**配置与服务**。
4. 选择**磁盘池**。
5. 选择与会话相关联的磁盘池。
6. 从**选择操作**菜单中选择**会话**。
7. 选择该会话。
8. 从**选择操作**菜单中选择**属性**。要更改相关联的副本描述，请选择副本描述，然后单击**编辑**。

诊断高可用性解决方案

配置 i5/OS 高可用性解决方案后，可能会遇到有关不同技术的问题，包括集群和跨站点镜像。

集群故障诊断

查找错误恢复解决方案以了解是否存在特定于集群的错误。

有时可能会出现集群未正常工作的情况。本主题包含有关您可能遇到的集群问题的信息。

确定是否存在集群问题

可从此处着手诊断集群问题。

有时可能会出现集群未正常工作的情况。如果您认为存在问题，可使用下列操作来帮助您确定是否存在问题以及该问题的本质。

- **确定集群在系统上是否处于活动状态。**

要确定集群资源服务是否处于活动状态，请在系统作业列表中查找以下两个作业：QCSTCTL 和 QCSTCRGM。如果这些作业处于活动状态，那么集群资源服务处于活动状态。可使用 IBM Director Navigator for i5/OS 或 System i 导航器中的“工作管理”功能来查看作业或使用 WRKACTJOB（使用活动作业）命令来完成此任务。还可使用 DSPCLUINF（显示集群信息）命令来查看集群的状态信息。

- 集群资源服务的其他作业也可能处于活动状态。集群作业提供了有关如何格式化集群资源服务作业的信息。

- **确定出现 CPFBB26 消息的原因。**

消息 : 集群资源服务未处于活动状态或未响应。

原因 : 因为资源不可用或者可能已被破坏，所以集群资源服务未处于活动状态或不能响应此请求。

此错误可能意味着 CRG 作业或集群未处于活动状态。使用 DSPCLUINF（显示集群信息）命令来确定该节点是否处于活动状态。如果该节点未处于活动状态，那么启动该集群节点。如果它处于活动状态，那么还应检查 CRG 以确定 CRG 是否存在问题。

在系统作业列表中查找 CRG 作业。可使用 IBM Director Navigator for i5/OS 或 System i 导航器中的“工作管理”功能来查看作业或使用 WRKACTJOB（使用活动作业）命令来完成此任务。还可使用 DSPCRGINF（显示 CRG 信息）命令（通过在命令中指定 CRG 名称）来查看特定 CRG 的状态信息。如果 CRG 作业未处于活动状态，那么查找 CRG 作业记录以确定导致该作业关闭的原因。修正该问题后，可使用 CHGCLURCY（更改集群恢复）命令或通过对该节点关闭并重新启动集群来重新启动 CRG 作业。

- **查找指示存在问题的消息。**

- 选择 F10 会在“包括详细消息”与“排除详细消息”之间切换显示，这样可确保您能够查看与集群命令相关的所有消息。选择包括所有详细消息并查看它们以确定是否需要其他操作。

- 查找 QSYSOPR 中等待响应的查询消息。

- 查找 QSYSOPR 中指示集群问题的错误消息。通常它们在 CPFBB00 到 CPFBBFF 范围内。

- 显示历史记录（DSPLOG CL 命令）以查找指示集群问题的消息。通常它们在 CPFBB00 到 CPFBBFF 范围内。

- **查找集群作业的作业记录以了解是否存在严重错误。**

这些作业一开始设置的记录级别为（4 0 *SECLVL），所以可查看必要的错误消息。应确保这些作业和出口程序作业适当设置了记录级别。如果集群未处于活动状态，您仍可查找集群作业和出口程序作业的假脱机文件。

- **如果怀疑存在某种中止情况，那么应查看集群作业的调用堆栈。**

确定是否有任何程序处于某种 DEQW（解队等待）状态。如果出现此情况，那么应检查每个线程的调用堆栈并了解其中任何线程的调用堆栈中是否包含 getSpecialMsg。

- **检查集群纵向许可内码（VLIC）记录项。**

这些记录项包含 4800 主代码。

- **使用 NETSTAT 命令来确定通信环境中是否存在任何异常情况。**

NETSTAT 会返回有系统上的 TCP/IP 网络路由、接口、TCP 连接和 UDP 端口状态的信息。

- 使用 Netstat 选项 1 (使用 TCP/IP 接口状态) 来确保选择用于集群的 IP 地址显示“活动”状态。还应确保 LOOPBACK 地址 (127.0.0.1) 也处于活动状态。
- 使用 Netstat 选项 3 (使用 TCP/IP 连接状态) 以显示端口号 (F14)。本地端口 5550 应处于“侦听”状态。在已确认“活动作业”列表中存在 QTOGINTD (用户 QTCP) 的情况下, 必须通过 STRTCPSVR *INETD 命令打开此端口。如果已对节点启动集群, 那么本地端口 5551 必须已打开并且处于“*UDP”状态。如果未启动集群, 那么端口 5551 一定不能打开, 否则它会导致无法对目标节点成功启动集群。
- 使用 PING 操作。如果尝试启动集群节点并且不能 PING 通该集群节点, 那么会接收到内部集群错误 (CPFBB46)。

收集集群的恢复信息

可使用使用集群 (WRKCLU) 命令来收集集群的完整信息。此信息可用来帮助您分析错误。

使用集群 (WRKCLU) 命令用于显示和使用集群节点和对象。运行此命令时, 将显示“使用集群”屏幕。除了显示集群中的节点和集群信息以外, 还可使用此命令来查看集群信息和收集有关集群的数据。

要收集错误恢复信息, 请执行下列步骤:

1. 在字符界面上, 输入 WRKCLU OPTION(OPTION)。可指定下列选项来指示要处理的集群状态信息。

*SELECT

显示“使用集群”菜单。

*CLUINF

显示集群信息。

*CFG 显示集群的性能和配置参数。

*NODE

显示“使用集群节点”面板, 此面板显示了集群中的节点列表。

*DEVDMN

显示“使用设备域”面板, 此面板显示了集群中的设备域列表。

*CRG 显示“使用集群资源组”面板, 此面板显示了集群中的集群资源组列表。

*ADMDMN

显示“使用管理域”面板, 此面板显示了集群中的管理域列表。

*SERVICE

收集集群中所有集群资源服务作业的相关跟踪和调试信息。此信息将被写入文件, 在该文件中, 每个集群资源服务作业都有一个成员。仅当得到服务供应商的指示时, 才应该使用此选项。它会显示转储集群跟踪 (DMPCLUTRC) 的提示面板。

常见集群问题

列示集群中可能发生的一些最常见问题以及避免这些问题和从这些问题恢复的方法。

下列常见问题很容易避免或恢复。

不能启动或重新启动集群节点

一些通信环境问题通常会造成此状况。为避免此状况, 应确保网络属性设置正确, 包括集群通信的回送地址、INETD 设置、ALWADDCLU 属性和 IP 地址。

- 如果尝试启动远程节点, 那么必须在目标节点上适当设置 ALWADDCLU 网络属性。根据您的环境, 它应该设置为 *ANY 或 *RQSAUT。
- 选择用于在本地和目标节点上集群的 IP 地址必须显示活动状态。

- 本地和目标节点上的 LOOPBACK 地址（127.0.0.1）也必须处于活动状态。
- 本地节点和任何远程节点必须能够使用要用于集群的 IP 地址执行 PING 操作以确保网络路由处于活动状态。
- INETD 在目标节点上必须处于活动状态。INETD 处于活动状态时，目标节点上的端口 5550 应处于侦听状态。请参阅 INETD 服务器以了解有关启动 INETD 服务器的信息。
- 在尝试启动节点之前，该节点上的端口 5551 一定不能打开，否则它会导致无法对目标节点成功启动集群。

结束时有一些散乱的单节点集群

如果要启动的节点不能与余下集群节点通信，那么会发生此状况。检查通信路径。

出口程序的响应很慢。

出现此状况的常见原因是出口程序使用的作业描述的设置不正确。例如，MAXACT 参数可能设置得太低，导致任何时间点只能有一个出口程序实例处于活动状态。建议将此项设置为 *NOMAX。

总体性能太差。

此症状有一些常见原因。

- 最可能的原因是共享通信线路上的通信流量太大。
- 另一个可能原因是通信环境与集群消息调整参数之间存在不一致。可使用检索集群资源服务信息（QcstRetrieveCRSInfo）API 来查看调整参数的当前设置，并使用更改集群资源服务（QcstChgClusterResourceServices）API 来更改这些设置。如果使用旧的适配器硬件，那么使用缺省集群调整参数设置的集群性能可能会下降。包括在旧的适配器定义中的适配器硬件类型为 2617、2618、2619、2626 和 2665。在此情况下，最好将性能类调整参数设置为正常。
- 导致此状况的另一个常见原因是 IP 多点广播组问题。如果一些节点的主集群地址（创建集群或添加节点时对给定节点输入的第二个地址）在公共 LAN 上，那么集群将利用 IP 多点广播功能。使用 NETSTAT 命令来确保主集群地址显示多点广播主机组 226.5.5.5。可对目标地址使用选项 14 显示多点广播组来显示。如果多点广播组不存在，请通过使用检索集群资源服务信息（QcstRetrieveCRSInfo）API 来验证启用多点广播集群调整参数设置的缺省值是否仍为 TRUE。
- 如果集群的所有节点都在本地 LAN 上，或者包含能够在整个网络路由中处理大小超过 1,464 字节的最大传输单元（MTU）包的路由功能，那么通过增加对应消息片段大小的集群调整参数值以更好地与路由 MTU 匹配，可以极大地提高大型集群消息（超过 1,536K 字节）的传输速度。

不能使用新发行版的任何功能。

如果您尝试使用新发行版功能但见到错误消息 CPFBB70，那么表示当前集群版本仍然设置为先前的版本级别。必须将所有集群节点升级至新发行版级别，然后使用调整集群版本界面将当前集群版本设置为新级别。请参阅“调整集群的版本”以了解更多信息。

不能向设备域添加节点或访问 System i 导航器集群管理界面。

要访问 System i 导航器集群管理界面或使用可切换设备，必须先系统在系统上安装 i5/OS 选项 41: HA 可切换资源。还必须具有此选项的有效许可证密钥。

已应用集群 PTF，但它无法正常工作。

在应用该 PTF 后，应确保已完成下列任务：

1. 关闭集群
2. 注销，然后登录

直到激活组被破坏，旧程序才会在激活组变为不活动状态。所有集群节点（包括集群 API）都在缺省激活组中运行。

3. 启动集群

大多数集群 PTF 要求对节点关闭并重新启动集群才能激活 PTF。

CEE0200 出现在出口程序作业记录中。

在此错误消息中，源模块为 QLEPM，而源过程为 Q_LE_leBdyPeilog。出口程序调用的所有程序必须在 *CALLER 或指定激活组中运行。必须更改您的出口程序或存在错误的程序来更正此问题。

集群资源服务作业记录中出现 CPD000D 并后跟 CPF0001。

接收到此错误消息时，应确保 QMLTTHDACN 系统值设置为 1 或 2。

集群中止。

确保集群资源组出口程序正在运行。要检查出口程序，请使用 WRKACTJOB（使用活动作业）命令，然后查看“函数”列以了解 PGM-QCSTCRGEXT 是否存在。

分区错误

可轻松更正某些集群问题。如果已发生集群分区，可了解如何恢复。本主题还会告诉您如何避免集群分区，并会给出如何将分区合并到一起的示例。

每当集群中一个或多个节点间的连接中断并且无法确认断开连接的节点的故障时，就表示集群中发生了集群分区。不要将它与逻辑分区（LPAR）环境中的分区相混淆。

如果在历史记录（QHST）或 QCSTCTL 作业记录中接收到错误消息 CPFBB20，那么表示发生了集群分区并且您需要知道如何恢复。以下示例显示涉及由四个节点（A、B、C 和 D）组成的集群的集群分区。该示例显示集群节点 B 与 C 之间发生了通信中断，从而导致该集群分为两个集群分区。在发生集群分区之前，一共有四个集群资源组（它们可能是任何类型），分别称为 CRG A、CRG B、CRG C 和 CRG D。该示例显示每个集群资源组的恢复域。

表 35. 集群分区期间的恢复域示例

节点 A	节点 B	x	节点 C	节点 D
CRG A (备份节点 1)	CRG A (主节点)			
	CRG B (主节点)		CRG B (备份节点 1)	
	CRG C (主节点)		CRG C (备份节点 1)	CRG C (备份节点 2)
CRG D (备份节点 2)	CRG D (主节点)		CRG D (备份节点 1)	
分区 1			分区 2	

如果通信路径中任何位置处的最大传输单元（MTU）小于消息参数片段大小这一集群通信可调整参数，那么会发生集群分区。可在目标节点上使用使用 TCP/IP 网络状态（WRKTCPSTS）命令来验证集群 IP 地址的 MTU。还可在每个步骤中沿整条通信路径验证 MTU。如果 MTU 小于消息片段大小，那么应提高路径的 MTU 或降低消息片段大小。可使用检索集群资源服务信息（QcstRetrieveCRSInfo）API 来查看调整参数的当前设置，并使用更改集群资源服务（QcstChgClusterResourceServices）API 来更改这些设置。

更正集群分区问题的原因后，集群会检测重新建立的通信链路并在历史记录（QHST）或 QCSTCTL 作业记录中发出 CPFBB21 消息。它会通知操作员集群已从集群分区问题恢复。要注意的是，更正集群分区问题后，可能需要几分钟的时间集群才能重新合并到一起。

确定主集群分区和辅助集群分区:

为了确定可在集群分区中采取的集群资源组操作类型，需要知道该分区是主集群分区还是辅助集群分区。检测到分区时，每个分区被指定为集群中定义的所有集群资源组的主分区或辅助分区。

对于主-备份模型，主分区包含作为当前主节点的节点。所有其他分区为辅助分区。所有集群资源组的主分区不能相同。

对等模型包含下列分区规则:

- 如果恢复域节点完全包含在一个分区中，那么该分区将是主分区。
- 如果恢复域节点跨分区，那么没有主分区。两个分区都是辅助分区。
- 如果集群资源组处于活动状态，并且给定分区中没有对等节点，那么将在该分区中关闭该集群资源组。
- 只要符合可操作更改的限制，就允许在辅助分区中进行可操作更改。
- 不允许在辅助分区中进行任何配置更改。

对每个集群资源组 API 的限制如下:

表 36. 集群资源组 API 分区限制

集群资源组 API	允许在主分区中进行的操作	允许在辅助分区中进行的操作
向恢复域添加节点	X	
添加 CRG 设备项		
更改集群资源组	X	
更改 CRG 设备项	X	X
创建集群资源组		
删除集群资源组	X	X
分发信息	X	X
关闭集群资源组 ¹	X	
启动切换	X	
列示集群资源组	X	X
列示集群资源组信息	X	X
从恢复域中除去节点	X	
除去 CRG 设备项	X	
启动集群资源组 ¹	X	

注:

1. 允许在对等集群资源组的所有分区中进行，但只影响运行该 API 的分区。

通过应用这些限制，可在集群不再分区时使集群资源组同步。当节点从分区状态重新加入集群时，主分区中的集群资源组版本将复制至辅助分区中的节点。

合并对等模型的两个辅助分区时，集群资源组状态为“活动”的分区将被声明为主分区。如果两个分区的集群资源组状态相同，那么包含集群资源组恢复域中列示的第一个节点的分区将被声明为主分区。成为主分区的分区中的集群资源组版本将复制至另一分区中的节点。

检测到分区时，不能在任何分区中运行添加集群节点项、调整集群版本和创建集群 API。仅当设备域中没有任何节点被分区时，才能运行添加设备域项 API。所有其他集群控制 API 可在任何分区中运行。但是，该 API 执行的操作仅影响运行该 API 的分区。

将分区的节点更改为“已失败”:

有时实际是节点中断时会报告分区情况。如果集群资源服务已断开与一个或多个节点的通信，但不能检测这些节点是否仍然工作，那么会发生此情况。出现此情况时，可使用一个简单的机制来指示节点已失败。

警告: 当您告诉集群资源服务某个节点已失败时，从分区状态恢复更为简单。但是，如果节点实际上处于活动状态并且确实已进行分区，那么不应该将节点状态更改为“已失败”。这样做会导致多个分区中的节点充当集群资源组的主节点。如果两个节点都认为它们是主节点，并且两个节点分别更改了它们的文件副本，那么可能文件或数据库之类的数据会变得散乱或遭到破坏。此外，在将每个分区中的一个节点指定为主节点后，就无法再将两个分区合并到一起。

将节点状态更改为“已失败”后，可对分区中每个集群资源组的恢复域中的节点角色进行重新排序。设置为“已失败”的节点将被指定为最后一个备份节点。如果多个节点发生故障并且需要更改它们的状态，那么节点的更改顺序将影响恢复域的备份节点的最终顺序。如果失败的节点是 CRG 的主节点，那么会将第一个活动备份节点指定为新的主节点。

当集群资源服务已断开与节点的通信，但无法检测该节点是否仍然工作，则该集群节点的状态为**未在通信**。您可能需要将该节点的状态从**未在通信**更改为**已失败**。然后您可以重新启动该节点。

要将该节点的状态从**未在通信**更改为**已失败**，请执行下列步骤:

1. 在 Web 浏览器中输入 `http://mysystem:2001`，其中 `mysystem` 是系统的主机名。
2. 使用您的用户概要文件和密码登录系统。
3. 从 IBM Systems Director Navigator for i5/OS 窗口中选择**集群资源服务**。
4. 在**集群资源服务**页中，选择**使用集群节点任务**以显示该集群中节点的列表。
5. 单击**选择操作**菜单，然后选择**更改状态**。将节点的状态更改为“已失败”。

相关信息

更改集群节点 (CHGCLUNODE) 命令

更改集群节点项 (QcstChangeClusterNodeEntry) API

分区集群管理域:

使用分区集群管理域时，应注意下列信息。

如果集群管理域已分区，那么更改会在每个分区中的活动节点间继续同步。节点再次合并到一起后，集群管理域会传播在每个分区中所作的所有更改，以使资源在活动域中保持一致。集群管理域的合并处理有一些注意事项:

- 如果所有分区处于活动状态并且对不同分区中的相同资源进行更改，那么在合并期间最新更改会应用于所有节点上的资源。最新更改通过开始更改的每个节点的全球标准时间 (UTC) 确定。
- 如果所有分区处于不活动状态，那么会根据任何分区处于活动状态时进行的最后一次更改来解析每个资源的全局值。直到表示集群管理域的对等 CRG 启动后，才会实际应用这些对受监视资源的更改。
- 如果在合并之前某些分区处于活动状态而某些分区处于不活动状态，那么表示在活动分区中所作更改的全局值将传播至不活动分区。然后不活动分区会启动，从而使得对不活动分区中的节点所作的所有暂挂更改传播至合并域。

技巧: 集群分区:

可使用下列技巧来处理集群分区。

1. 用于限制分区内操作的规则是按照能够合并分区的思路设计的。如果没有这些限制，那么重新构造集群需要大量工作。
2. 如果主分区中的节点已被破坏，那么可能需要在辅助分区中进行特殊处理。导致此状况的最常见情况是构成主分区的站点丢失。我们在从分区错误恢复时使用此示例并假定分区 1 已被破坏。在此情况下，集群资源组 B、C 和 D 的主节点必须在分区 2 中。最简单的恢复方法是使用“更改集群节点项”同时将节点 A 和节点 B 设置为已失败。请参阅“将分区节点更改为已失败”以了解有关如何完成此操作的更多信息。还可手工恢复。为此，请执行下列操作：
 - a. 从分区 2 内的集群中除去节点 A 和节点 B。分区 2 现在成为集群。
 - b. 建立新集群中需要的所有逻辑复制环境。例如，使用“启动集群资源组”API/CL 命令等等。

因为已从分区 2 内的集群定义中除去节点，所以尝试合并分区 1 和分区 2 的操作会失败。为更正集群定义中的不匹配问题，应对分区 1 中的每个节点运行 删除集群 (QcstDeleteCluster) API，然后将分区 1 中的节点添加至集群，并重新建立所有集群资源组定义、恢复域和逻辑复制。这需要大量工作，并且很容易发生错误。您仅应在站点崩溃的情况下执行此过程。

3. 应根据要启动的节点的状态来处理启动节点操作：

该节点已失败或“关闭节点”操作已关闭该节点：

- a. 对要添加的节点启动集群资源服务
- b. 将集群定义从集群中的活动节点复制至要启动的节点。
- c. 将恢复域中包含要启动的节点的所有集群资源组从集群中的活动节点复制至要启动的节点。不要将集群资源组从要启动的节点复制至集群中的活动节点。

该节点为分区节点：

- a. 比较活动节点的集群定义与要启动的节点的集群定义。如果两个定义完全相同，那么启动操作将作为合并操作继续执行。如果两个定义不相同，那么合并操作会停止，并且需要用户干预。
- b. 如果合并操作继续，那么要启动的节点将设置为活动状态。
- c. 将恢复域中包含要启动的节点的所有集群资源组从集群资源组主分区复制至备份分区。可将集群资源组从要启动的节点复制至在集群中已处于活动状态的节点。

集群恢复

了解有关如何从可能发生的其他集群故障恢复的信息。

从集群作业故障恢复：

集群资源服务作业故障通常意味着存在一些其他问题。

应查找与失败的作业相关联的作业记录以及描述作业失败原因的消息。更正所有错误状况。

可使用更改集群恢复 (CHGCLURCY) 命令来重新启动已关闭的集群资源组作业，而不必对节点关闭并重新启动集群。

1. CHGCLURCY CLUSTER(EXAMPLE)CRG(CRG1)NODE(NODE1)ACTION(*STRCRGJOB) 此命令会导致在作业 NODE1 上提交集群资源组作业 CRG1。要在 NODE1 上启动集群资源组作业，要求集群在 NODE1 上处于活动状态。
2. 对该节点重新启动集群。

如果要使用 IBM 业务合作伙伴集群管理产品，请参阅随该产品提供的文档。

相关信息

更改集群恢复 (CHGCLURCY) 命令

恢复被破坏的集群对象:

虽然您可能未遇到被破坏的对象, 但集群资源服务对象有可能会被破坏。

如果系统是活动节点, 那么系统会尝试从集群中的另一个节点恢复。该系统将执行下列恢复步骤:

用于被破坏的内部对象

1. 关闭被破坏的节点。
2. 如果集群中至少还有一个活动节点, 那么被破坏的节点将自己自动重新启动并重新加入集群。重新加入的过程将更正被破坏的状况。

用于被破坏的集群资源组

1. 包含被破坏的 CRG 的节点会导致所有正在进行并且与该 CRG 相关联的操作失败。然后系统会尝试从另一活动节点自动恢复该 CRG。
2. 如果恢复域至少有一个活动成员, 那么 CRG 恢复会起作用。否则该 CRG 作业会关闭。

如果系统无法找到或访问任何其他活动节点, 那么您需要执行下列恢复步骤。

用于被破坏的内部对象

您接收到内部集群错误 (CPFBB46、CPFBB47 或 CPFBB48)。

1. 关闭对包含破坏的节点的集群。
2. 重新启动对包含破坏的节点的集群。从集群中的另一个活动节点完成此操作。
3. 如果第 1 步和第 2 步未解决该问题, 那么应从集群中除去被破坏的节点。
4. 将系统添加至集群和适当集群资源组的恢复域。

用于被破坏的集群资源组

您接收到一个错误, 指示某个对象已被破坏 (CPF9804)。

1. 对包含被破坏的集群资源组的节点关闭集群。
2. 使用 DLTCRG 命令删除该 CRG。
3. 如果包含该 CRG 对象的集群中没有其他活动节点, 那么应从介质恢复。
4. 对包含被破坏的集群资源组的节点启动集群。可从任何活动节点完成此操作。
5. 启动集群时, 系统会使所有集群资源组再同步。如果集群中的所有其他节点都未包含该 CRG, 那么您可能需要重新创建该 CRG。

在系统完全崩溃后恢复集群:

在系统意外断电导致系统完全崩溃后, 应将此信息与“恢复系统”主题中的适当核对表配合使用来恢复整个系统。

方案 1: 恢复为同一系统

1. 为了避免许可内码与 i5/OS 间的设备域信息存在不一致, 建议使用选项 3 (安装许可内码并恢复配置) 来安装许可内码。

注: 为了成功执行“安装许可内码并恢复配置”操作, 必须具有相同的磁盘机, 但装入源磁盘机例外 (如果它已经失败)。您还必须恢复同一发行版。

2. 安装许可内码后，请执行恢复系统主题中的恢复磁盘配置过程。如果使用这些步骤，您就不必重新配置磁盘池。
3. 恢复系统信息并准备好对刚刚恢复的节点启动集群后，必须从该活动节点启动集群。这会将最新的配置信息传播至已恢复节点。

方案 2: 恢复为不同系统

如果您恢复了系统信息并且检查了作业记录以确保所有对象已恢复，那么必须执行下列步骤以获取正常的集群设备域配置。

1. 从刚刚恢复的节点中删除该集群。
2. 从活动状态中执行下列步骤:
 - a. 从该集群中除去已恢复节点。
 - b. 将已恢复节点添加至集群。
 - c. 将已恢复节点添加至设备域。
 - d. 创建集群资源组或将该节点添加至恢复域。

在发生灾难后恢复集群:

如果发生灾难并且所有节点都已丢失，那么需要重新配置集群。

为了应对这种情况，建议您保存集群配置信息并创建该信息的硬拷贝打印输出。

从备份磁盘恢复集群:

在正常操作期间，您决不应从备份磁带恢复。

只有在发生灾难并且集群中的所有节点都已丢失的情况下，才需要这样做。如果发生灾难，那么应按照创建备份与恢复策略后设置的适当正常恢复过程来恢复。

诊断跨站点镜像

将您的简短描述放在此处用作第一段和摘要。

在此处输入您的正文。

- 关键点
- 另一个关键点

小标题

此处是概念中的一小段。

示例

此处是概念中的一小段示例。

地理镜像消息

查看地理镜像消息描述和恢复操作，以解决地理镜像问题。

0x00010259

描述: 由于系统找不到镜像副本，因此操作失败。

恢复：并非设备域中的所有节点都进行了响应。请确保集群功能处于活动状态。必要时，在节点上启动集群功能。有关详细信息，请参阅第 88 页的『启动节点』。然后，再次尝试该请求。如果问题仍然存在，请与技术支持提供者联系。

0x0001025A

描述：并非磁盘池组中的所有磁盘池都已进行地理镜像。

恢复：如果对磁盘池组中的一个磁盘池进行地理镜像，那么必须对该磁盘池组中的所有磁盘池进行地理镜像。请执行下列其中一项操作：

1. 对尚未进行地理镜像的磁盘池配置地理镜像。
2. 对已进行地理镜像的磁盘池取消配置地理镜像。

0x00010265

描述：拆离的镜像副本可用。

恢复：使拆离的镜像副本变为不可用状态，然后再次尝试执行重新连接操作。

0x00010380

描述：镜像副本的配置中缺少磁盘机。

恢复：查找或修复该镜像副本中缺少的磁盘机。检查目标节点上的产品活动记录。回收 IOP 高速缓存存储器。

0x00011210

描述：未对磁盘池组的建议辅助磁盘池进行地理镜像。

恢复：如果对磁盘池组中的一个磁盘池进行地理镜像，那么必须对该磁盘池组中的所有磁盘池进行地理镜像。您必须立即或者在完成此操作后对尚未进行地理镜像的建议辅助磁盘池配置地理镜像。

0x00011211

描述：存在重复的镜像副本。

恢复：检查可能存在于 Enterprise Storage Server® 和 FlashCopy 系统上并以本地方式进行镜像的磁盘机或者后备级独立磁盘池副本。有关更多信息，请参阅镜像副本节点上的产品活动记录。消除重复，然后再试该请求。如果问题仍然存在，请与技术支持提供者联系，或者参阅 i5/OS 技术支持以获取有关 IBM 支持和服务的信息。

代码许可证和免责声明

IBM 授予您使用所有编程代码示例的非专有版权许可，您可以由此生成根据您的特定需要而定制的相似功能。

关于程序或技术支持（如果有的话），除了不能排除的法定保证外，IBM、其程序开发商和供应商不进行任何明示或暗含的保证和条件，包括但不限于暗含的有关适销性、适用于某种特定用途和非侵权的保证和条件。

在任何情况下，IBM、其程序开发商或供应商不负责以下任何损失，即使告知了其可能性：

1. 数据丢失或损坏；
2. 直接的、特殊的、附带的或间接的损坏，或任何具有经济后果的损坏；
3. 损失利润、交易、收入、信誉或预期节约。

某些管辖区域不允许排除或限制直接的、偶然的或随之发生的损失，因此某些或全部上述限制或排除可能不适用于您。

| 安装 IBM System i High Availability Solutions Manager (iHASM) 许可程序

| 在可以实现 i5/OS 高可用性解决方案之前，必须在每个参与高可用性的系统上安装 IBM System i High Availability Solutions Manager (iHASM) 许可程序 (5761-HAS)。

| 在安装 iHASM 许可程序前，应该完成下列安装任务：

- | 1. 安装或升级到 V6R1 i5/OS。
- | 2. 安装 i5/OS 选项 41 (HA 可切换资源)。

| 要安装 iHASM 许可程序，请执行下列步骤：

- | 1. 在命令行上输入 GO LICPGM。
- | 2. 在“使用许可程序”屏幕上，选择选项 11 (安装许可程序)。
- | 3. 选择“产品 5761-HAS，选项 *BASE”以安装 System i 产品的 iHASM。然后，按 Enter 键。
- | 4. 在“安装选项”屏幕上，按照要求输入安装设备的名称。然后，按 Enter 键以开始进行安装。

附录. 声明

本信息是为在美国提供的产品和服务编写的。

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

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

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

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

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

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

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

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

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

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

IBM Corporation
Software Interoperability Coordinator, Department YBWA
3605 Highway 52 N
Rochester, MN 55901
U.S.A.

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

- 1 本文档中描述的许可程序及其所有可用的许可资料均由 IBM 依据 IBM 客户协议、IBM 国际软件许可协议、
- 1 IBM 机器代码许可协议或任何同等协议中的条款提供。

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

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

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

所有 IBM 的价格均是 IBM 当前的建议零售价，可随时更改而不另行通知。经销商的价格可与此不同。

本信息仅用于规划目的。在所描述的产品可用之前，此处的信息可能更改。

本信息包含日常业务经营中使用的数据和报告的示例。为了尽可能完整地说明这些示例，这些示例中包括个人、公司、品牌和产品的名称。所有这些人或名称均系虚构，如有实际的企业名称和地址与此雷同，纯属巧合。

版权许可：

本信息包括源语言形式的样本应用程序，这些样本说明不同操作平台上的编程方法。如果是为按照在编写样本程序的操作平台上的应用程序编程接口（API）进行应用程序的开发、使用、经销或分发为目的，您可以任何形式对这些样本程序进行复制、修改、分发，而无须向 IBM 付费。这些示例并未在所有条件下作全面测试。因此，IBM 不能担保或暗示这些程序的可靠性、可维护性或功能。

凡这些样本程序的每份拷贝或其任何部分或任何衍生产品，都必须包括如下版权声明：

©（贵公司的名称）（年）。此部分代码是根据 IBM 公司的样本程序衍生出来的。© Copyright IBM Corp. _（输入年份）。All rights reserved.

如果您正以软拷贝格式查看本信息，图片和彩色图例可能无法显示。

1 编程接口信息

此《通过基于任务的方法实现高可用性》出版文档适用于允许客户编写程序以获取 IBM i5/OS 服务的编程接口。

商标

下列各项是 International Business Machines Corporation 在美国和 / 或其他国家或地区的商标：

i5/OS

IBM

IBM（徽标）

System i

System i5

IBM Systems Storage

TotalStorage

FlashCopy

162 System i: 可用性 通过基于任务的方法实现高可用性

- | Adobe、Adobe 徽标、PostScript 和 PostScript 徽标是 Adobe Systems Incorporated 在美国和 / 或其他国家或地区的注册商标或商标。
- | Microsoft、Windows、Windows NT 和 Windows 徽标是 Microsoft Corporation 在美国和 / 或其他国家或地区的商标。
- | Java 和所有基于 Java 的商标是 Sun Microsystems,Inc. 在美国和 / 或其他国家或地区的商标。

其他公司、产品或服务名称可能是其他公司的商标或服务标记。

条款和条件

如果符合以下条款和条件，则授予您使用这些出版物的许可权。

个人使用: 只要保留所有的专有权声明，您就可以为个人、非商业使用复制这些出版物。未经 IBM 明示同意，您不可以分发、显示或制作这些出版物或其中任何部分的衍生产品。

商业使用: 只要保留所有的专有权声明，您就可以仅在企业内复制、分发和显示这些出版物。未经 IBM 明示同意，您不可以制作这些出版物的衍生产品，或者在您的企业外部复制、分发或显示这些出版物或其中的任何部分。

在本许可权中除明示地授权以外，没有把其他许可权、许可证或权利（无论是明示的，还是默示的）授予其中包含的出版物或任何信息、数据、软件或其他知识产权。

一旦使用这些出版物损害了 IBM 的利益，或者 IBM 确定以上指令未被正确遵守，则 IBM 保留自行决定撤销此处授予的许可权的权利。

您不可以下载、出口或再出口此信息，除非完全符合所有适用的法律和法规，包括所有美国出口法律和法规。

IBM 对这些出版物的内容不作任何保证。这些出版物以“按现状”的基础提供，不附有任何形式的（无论是明示的，还是默示的）保证，包括（但不限于）对非侵权性、适销性和适用于某特定用途的默示保证。



中国印刷