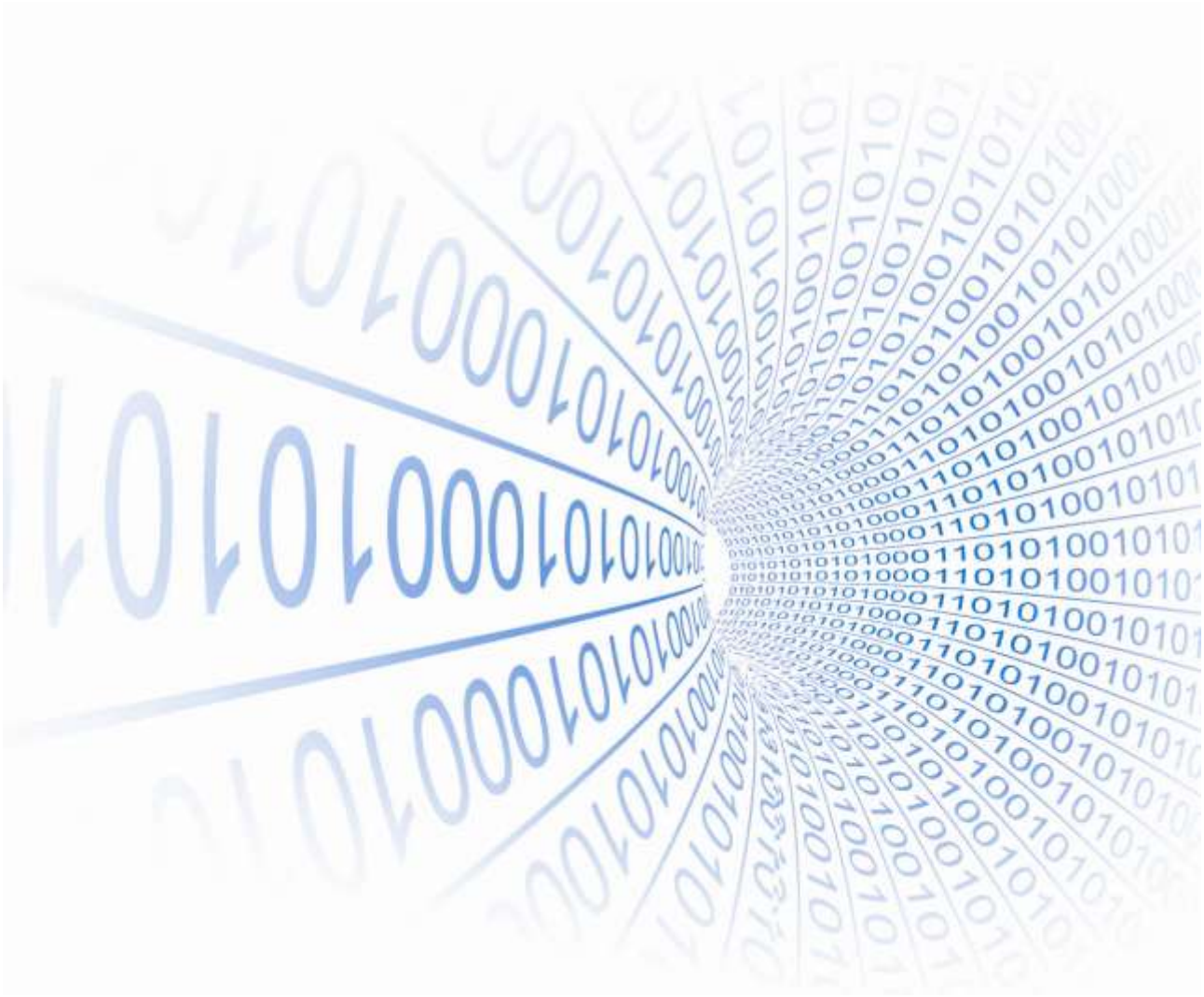
业务连续性容灾解决方案

主备容灾解决方案技术白皮书



HUAWEI

目 录

1 概述

1.1 单数据中心运行挑战

从传统IT 系统改造为数据中心模式之后，设备资源和管理都有了较大提升，但是数据的集中也引发了新的问题，故障域变得更大，任意一个小的设备故障或者灾难可能引发大面积的业务中断，甚至数据永久丢失，如果没有符合业务连续性要求的数据保护手段，保障各种灾难场景下的数据安全，用户随时可能面临着严重的业务中断和数据丢失风险。

1.2 方案简介

如今企业和政府部门越来越依赖信息化进行办公、服务、发展与决策，数据丢失和业务中断会造成巨大经济与信誉损失。911 事件和四川大地震证明，各种自然灾难（火灾、水灾、地震等）和人为灾难（误操作、病毒等）总是无法避免的。

数据和业务的灾难备份已成为信息系统建设的必然要求和发展趋势。《国家信息化领导小组关于加强信息安全保障工作的意见》(中办发2003[27]号)、《关于做好国家重要信息系统灾难备份的通知》(国信办[2004]11 号) 、《重要信息系统灾难恢复指南》、《信息系统灾难恢复规范》（GB/T20988-2007）等文件的相继出台，标志着容灾系统的建立正走向规范化。

主备容灾，指客户在生产中心之外，另一地点选址建设容灾中心，形成一对一的数据级或应用级保护。相对于两地三中心、集中容灾等复杂拓扑结构，主备容灾是目前市场上采用得最广泛的容灾方式。

1.3 方案亮点

强大的扩展性

- 方案采用虚拟化存储直接异构接管或者存储镜像卷技术，实现在网异构存储的资源整合，存储空间扩展和利旧更方便。
- SAN和NAS一体化容灾，数据库层容灾，容灾方案扩展更简单。

业务快速恢复

- 支持一键式演练和容灾切换，自动化脚本代替繁琐的手工操作，降低系统RTO。
- 提供基于应用的数据一致性保护，能够感知生产主机的关键应用，在异步远程复制开启时，先进行主机内存数据刷盘，保证应用数据一致性，确保灾备端的业务能够拉起。

便捷管理维护

- 一站式管理灾备中心的服务器、存储、网络资源，同时管理客户侧接入设备，使得整个系统管理简单、明确。
- 管理软件具备图形化操作和容灾拓扑显示，使用户感知更加直观，运维更加方便。完善的登录和操作日志，为事后审计提供良好支撑。
- 提供可自定义的Dashboard，展示容灾相关的图形报表，包括保护策略日程、保护策略执行情况等。

2 方案架构

2.1 解决方案概述

随着信息化的发展，企业以及政府行业对容灾系统建设的需求越来越明显，特别在医疗、政府、大企业、金融等领域，容灾系统建设已经成为信息化建设的必要步骤。传统容灾系统建设主要关注容灾设计和部署，忽略了容灾系统管理和灾难恢复的自动化流程。在部署交付完成后，客户往往发现系统维护困难，真正灾难发生时，需要非常繁琐的操作和很长的时间才能恢复数据和业务，实际使用效果与之前的容灾规划差距很大。

华为广泛汲取了在医疗、政府、金融等行业中多个容灾项目中的实践经验，深入挖掘客户在容灾建设中的需求，结合华为在融合存储架构、统一管理平台上的优势，推出主备容灾解决方案，帮助客户合理规划，建设易管理、易维护、支持快速数据和业务恢复的容灾系统。

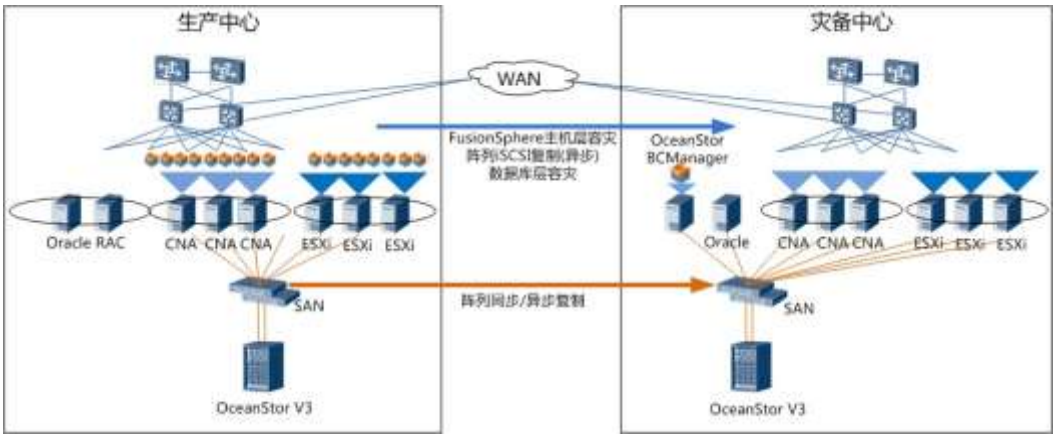
主备容灾解决方案主要针对两种客户场景：主备都是华为存储的场景和生产存储为友商阵列的场景。

2.2 主备存储同构场景

2.2.1 方案架构

2.2.1.1 SAN容灾方案

图2-1 方案架构设计图



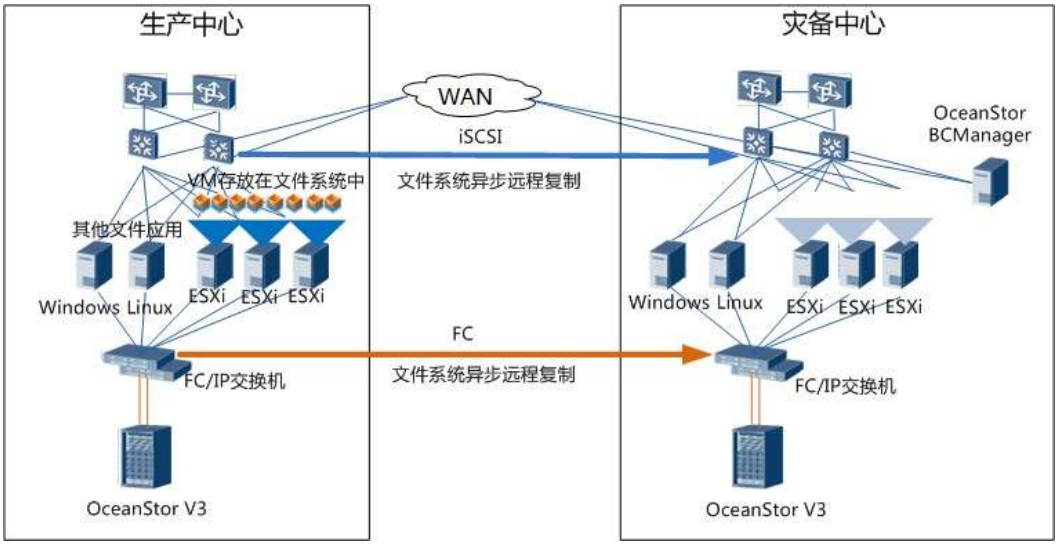
说明

- 阵列同步/异步复制均使用OceanStorV3的远程复制功能。
- 容灾管理软件OceanStorBCManager只需在灾备端部署1套即可，可部署在物理机或者虚拟机上。
- 方案支持的应用不限于图中OracleRAC、FusionSphere和VMware，具体支持请参见方案规格。

如图2-1 所示，本场景下生产存储与灾备存储都部署华为阵列，采用阵列复制技术（HyperReplication/A 或者HyperReplication/S）实现容灾数据复制，当然针对数据库业务，也可以选择通过应用层复制方案完成数据同步，而对于华为云平台还可以选择通过主机层复制方式实现数据同步。

2.2.1.2 NAS容灾方案

图2-2 方案架构设计图



说明

- 文件系统复制使用OceanStor V3的文件系统异步远程复制功能。
- 容灾管理软件OceanStorBCManager只需在灾备端部署1套即可，可部署在物理机或者虚拟机上。
- 方案支持的应用不限于图中VMware，具体支持请参见方案规格。

如图2-2 所示，本场景下生产存储与灾备存储都部署华为阵列，采用文件系统异步远程复制技术（HyperReplication/A）实现生产数据的保护。

2.2.2 方案设计

2.2.2.1 SAN应用数据一致性设计

容灾数据的一致性是容灾建设的关键点，传统存储容灾方案只关注存储层的数据一致，对某些复杂应用，如数据库应用，由于丢失了主机内存数据，存在数据不一致的风险。

本方案提供的容灾管理软件，能够感知生产主机的关键应用，如Oracle 、SQL Server、DB2 等，在异步远程复制开启时，先进行主机内存数据刷盘，确保所有内存数据能够同步到灾备端，避免主机内存数据丢失导致的数据不一致情况发生。

2.2.2.2 RPO&RTO

RPO

本场景支持的最小RPO 为0，需要采用HyperReplication/S(同步复制)技术做容灾数据复制。

若客户采用HyperReplication/A(异步复制)技术做SAN 数据复制，最小可以做到秒级的RPO。若客户需要应用保护，采用刷内存功能保证数据一致性，不建议过于频繁进行刷内存操作，建议复制周期设置为15 分钟以上。

采用HyperReplication/A(异步复制)技术做NAS 数据复制，同步周期最短5 分钟。为了降低性能影响，不建议过于频繁进行周期复制，建议复制周期设置为15 分钟以上。

RTO

方案提供的容灾管理软件支持一键式容灾切换，使用自动化脚本代替繁琐的人工切换操作，加速业务恢复，降低RTO，简化容灾运维难度。

针对Oracle 数据库，自动切换脚本能够代替存储复制切换、LUN 映射、存储配置、启动应用、测试应用等手工操作。因此本方案支持分钟级别的RTO。

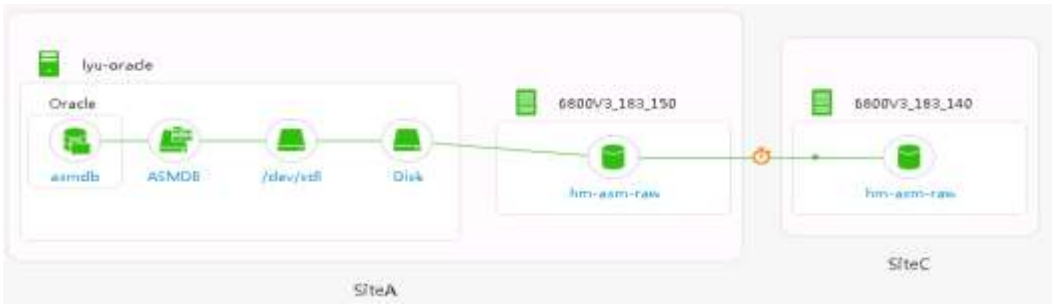
2.2.2.3 容灾管理

本方案提供的容灾管理软件提供针对应用的容灾管理，运维管理功能设计参见章节“4 容灾场景操作流程”。

方案运维管理功能的一大特点是提供可视化管理。

如图2-3 所示，客户可以从拓扑图中看到关键应用的数据文件对应的存储LUN，以及这些存储LUN 的容灾状态。基于应用的容灾视图有效展示了应用容灾状态，简化容灾配置，降低由于应用的部分关键文件没有保护导致灾备端业务无法拉起的风险。

图2-3 Oracle 应用可视化容灾拓扑



其他运维设计功能参见章节“4容灾场景操作流程”。

2.2.3 方案亮点

存储互连互通

- 华为存储系统支持高中低端存储容灾业务直接互通，灾备端按需配置，支持配置低端阵列，降低灾备投资50%以上。
- 采用阵列容灾复制技术，存储层完成容灾数据复制，减少容灾业务对生产服务器的性能影响。
- 支持同步、异步复制模式，可根据客户RPO、RTO需求，针对具体应用量身打造容灾方案。

业务快速恢复

- 支持一键式演练和容灾切换，自动化脚本代替繁琐的手工操作，降低系统RTO。
- 提供基于应用的数据一致性保护，能够感知生产主机的关键应用，在异步远程复制开启时，先进行主机内存数据刷盘，保证应用数据一致性，确保灾备端的业务能够拉起。

便捷管理维护

- 一站式管理灾备中心的服务器、存储、网络资源，同时管理客户侧接入设备，使得整个系统管理简单、明确。
- 管理软件具备图形化操作和容灾拓扑显示，使用户感知更加直观，运维更加方便。完善的登录和操作日志，为事后审计提供良好支撑。
- 提供可自定义的Dashboard，展示容灾相关的图形报表，包括保护策略日程、保护策略执行情况等。

2.2.4 典型配置

2.2.4.1 组网

本场景中使用OceanStor V3 作为数据存储，然后采用阵列复制技术（HyperReplication/A 或者HyperReplication/S）实现主备站点华为阵列的容灾数据复制。如下图所示，列举了三种典型的组网配置：

1. 如图2-4所示，生产中心采用FC SAN组网部署2节点OracleRAC，灾备中心部署单实例Oracle数据库，并在灾备中心的一台虚拟机上部署OceanStorBCManager 进行复制任务调度与容灾管理；
2. 如图2-5所示，生产中心采用SAN组网部署VMwareHA集群，灾备中心部署ESXi 主机，并在其中一个虚拟机上部署OceanStor BCManager 进行复制任务调度与容灾管理；
3. 如图2-6所示，生产中心采用NAS 组网部署VMwareHA集群，灾备中心部署ESXi主机，并在其中一个虚拟机上部署OceanStorBCManager进行复制任务调度与容灾管理。

图2-4 典型配置组网图（Oracle RAC）

图2-5 典型配置组网图（VMware vSphere SAN）

图2-6 典型配置组网图（VMware vSphere NAS）

2.2.4.2 规格

本章节以2.2.4典型配置为例，描述典型配置的核心参数，如果实际采用的是其他型号的华为存储，参数参见具体产品规格表。

- **硬件规格参数**

名称	参数
生产存储（OceanStor 6800 V3）	
最大硬盘数量	3200
硬盘参数	2.5 寸硬盘框： • 10kr/min SAS：300GB、600GB、900GB、1.2TB • 15kr/min SAS：300GB、600GB • SSD（SLC）：200GB • SSD（eMLC）：400GB、600GB、900GB、1.8TB 3.5 寸硬盘框： • 7200r/minNL-SAS：2TB、3TB、4TB、6TB • 7200r/min SATA：1TB、2TB、3TB • SSD（SLC）：200GB • SSD（eMLC）：400GB、600GB、900GB、1.8TB
灾备存储（OceanStor 5500 V3）	
最大硬盘数量	750

名称	参数
硬盘参数	2.5 寸硬盘框： • 10kr/min SAS：300GB、600GB、900GB、1.2TB • 15kr/min SAS：300GB、600GB • SSD（SLC）：200GB • SSD（eMLC）：400GB、600GB、900GB、1.8TB 3.5 寸硬盘框： • 7200r/minNL-SAS：2TB、3TB、4TB、6TB • 7200r/minSATA：1TB、2TB、3TB • SSD（SLC）：200GB • SSD（eMLC）：400GB、600GB、900GB、1.8TB

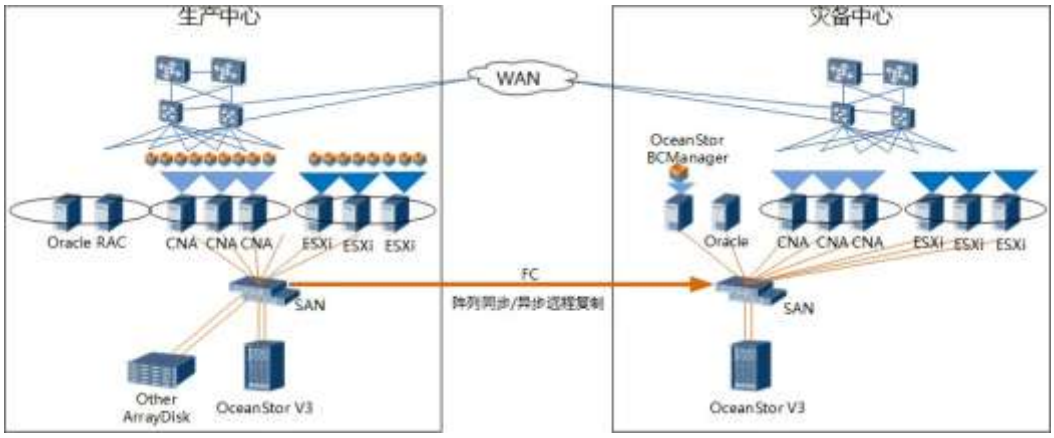
- **软件规格参数**

名称	参数
容灾管理软件（BCManager）	
站点数	32 个本地站点，32 个远程站点
保护组数	最大支持256 个保护组
保护策略数	最大支持256 个保护策略
并发执行的保护策略数	最多支持16 个保护策略同时执行
恢复计划数	最大支持256 个恢复计划
并发执行的恢复计划数	最大支持10 个恢复计划同时执行
存储阵列数	存储阵列最多支持32 个，其中2 高端存储或8 中端存储或32 低端存储
业务主机数	64 台

2.3 主备存储异构场景

华为存储通过卷镜像或直接接管异构存储LUN，对V3 镜像卷或者对接管后的LUN 使用OceanStor V3 远程复制，将数据从生产中心同步至灾备中心进行异地保护。

图2-7 架构设计图



说明

- 阵列同步/异步复制均使用OceanStorV3的远程复制功能。
- 此处生产中心业务LUN有两种存在形态：镜像和直接接管异构存储LUN，对V3镜像卷或者对接管后的LUN使用OceanStorV3复制。
- 生产中心可使用OceanStorV3接管异构存储阵列。
- 容灾管理软件OceanStorBCManager只需在灾备端部署1套即可，可部署在物理机或者虚拟机上。
- 方案支持的应用不限于图中OracleRAC、FusionSphere和VMware，具体支持请参见方案规格。

华为OceanStor V3 存储支持异构虚拟化接管，可以直接将现网其它品牌（兼容性列表内）的LUN 映射给OceanStor V3，然后映射给业务主机使用。或通过OceanStor V3 自身的卷镜像特性实现华为存储于异构存储的高可用，再通过华为存储的阵列远程复制功能，实现异地主备容灾。

2.3.2 方案设计

2.3.2.1 应用数据一致性设计

容灾数据的一致性是容灾建设的关键点，传统存储容灾方案只关注存储层的数据一致，对某些复杂应用，如数据库应用，由于丢失了主机内存数据，存在数据不一致的风险。

本方案提供的容灾管理软件，能够感知生产主机的关键应用，如Oracle 、SQL Server、DB2（具体支持请参见方案规格）等，在异步远程复制开启时，先进行主机内存数据刷盘，确保所有内存数据能够同步到灾备端，避免主机内存数据丢失导致的数据不一致情况发生。

2.3.2.2 RPO&RTO

RPO

本场景支持的最小RPO 为0，需要采用HyperReplication/S(同步复制)技术做容灾数据复制。

若客户采用HyperReplication/A(异步复制)技术做容灾数据复制，最小可以做到秒级RPO。若客户需要应用保护，采用刷内存功能保证数据一致性，不建议过于频繁进行刷内存操作，建议复制周期设置为15 分钟以上。

RTO

方案提供的容灾管理软件支持一键式容灾切换，使用自动化脚本代替繁琐的人工切换操作，加速业务恢复，降低RTO，简化容灾运维难度。

针对Oracle 数据库，自动切换脚本能够代替存储复制切换、LUN 映射、存储配置、启动应用、测试应用等手工操作。因此本方案支持分钟级别的RTO。

2.3.2.3 容灾运维管理

本方案提供的容灾管理软件提供针对应用的容灾管理，运维管理功能设计参见章节“4 容灾场景操作流程”。

方案运维管理功能的一大特点是提供可视化管理。

如图2-8 所示，客户可以从拓扑图中看到关键应用的数据文件对应的存储LUN，以及这些存储LUN 的容灾状态。基于应用的容灾视图有效展示了应用容灾状态，简化容灾配置，降低由于应用的部分关键文件没有保护导致灾备端业务无法拉起的风险。

图2-8 Oracle 可视化容灾拓扑

其他运维设计功能参见章节“4容灾场景操作流程”。

2.3.3 方案亮点

异构接管提高利用率

- 方案采用虚拟化存储直接异构接管或者存储卷镜像技术，实现在网异构存储的资源整合。
- 存储卷镜像技术，为生产存储提供双保险，接管的异构阵列故障或者需要停机维护时，华为阵列继续提供业务访问能力，上层业务无影响。
- 方案采用存储虚拟化技术兼容友商阵列，使友商存储和华为存储可以共存，同时对上层提供服务。

异地业务快速恢复

- 支持一键式演练和容灾切换，自动化脚本代替繁琐的手工操作，降低系统RTO。
- 提供基于应用的数据一致性保护，能够感知生产主机的关键应用，在异步远程复制开启时，先进行主机内存数据刷盘，保证应用数据一致性，确保灾备端的业务能够拉起。

便捷管理维护

- 一站式管理灾备中心的服务器、存储、网络资源，同时管理客户侧接入设备，使得整个系统管理简单、明确。
- 管理软件具备图形化操作和容灾拓扑显示，使用户感知更加直观，运维更加方便。完善的登录和操作日志，为事后审计提供良好支撑。
- 提供可自定义的Dashboard，展示容灾相关的图形报表，包括保护策略日程、保护策略执行情况等。

2.3.4 典型配置

2.3.4.1 组网

如图2-9、图2-10 所示，列举了两种应用的典型组网配置，本场景中首先使用OceanStorV3 异构虚拟化技术直接异构接管友商存储或者构造卷镜像，然后采用阵列复制技术（HyperReplication/A 或者HyperReplication/S）实现主备站点华为阵列的容灾数据复制。

图2-9 典型配置组网图（Oracle RAC）

图2-10 典型配置组网图（VMware vSphere）

2.3.4.2 规格

本章节以2.3.4典型配置中的配置为例，描述典型配置的核心参数，如果实际采用的是其他型号的华为存储，参数参见具体产品规格表。

- **硬件规格参数**

名称	参数
生产存储（OceanStor 6800 V3）	
最大硬盘数量	3200
硬盘参数	2.5 寸硬盘框： • 10kr/minSAS：300GB、600GB、900GB、1.2TB • 15kr/min SAS：300GB、600GB • SSD（SLC）：200GB • SSD（eMLC）：400GB、600GB、900GB、1.8TB 3.5 寸硬盘框： • 7200r/minNL-SAS：2TB、3TB、4TB、6TB • 7200r/minSATA：1TB、2TB、3TB • SSD（SLC）：200GB • SSD（eMLC）：400GB、600GB、900GB、1.8TB
灾备存储（OceanStor 5500 V3）	
最大硬盘数量	750
硬盘参数	2.5 寸硬盘框： • 10kr/min SAS：300GB、600GB、900GB、

名称	参数
	1.2TB • 15kr/min SAS：300GB、600GB • SSD（SLC）：200GB • SSD（eMLC）：400GB、600GB、900GB、1.8TB 3.5 寸硬盘框： • 7200r/minNL-SAS：2TB、3TB、4TB、6TB • 7200r/minSATA：1TB、2TB、3TB • SSD（SLC）：200GB • SSD（eMLC）：400GB、600GB、900GB、1.8TB

- **软件规格参数**

名称	参数
容灾管理软件（BCManager）	
站点数	32 个本地站点，32 个远程站点
保护组数	最大支持256 个保护组
保护策略数	最大支持256 个保护策略
并发执行的保护策略数	最多支持16 个保护策略同时执行
恢复计划数	最大支持256 个恢复计划
并发执行的恢复计划数	最大支持10 个恢复计划同时执行
存储阵列数	存储阵列最多支持32 个，其中2 高端存储或8 中端存储或32 低端存储
业务主机数	64 台

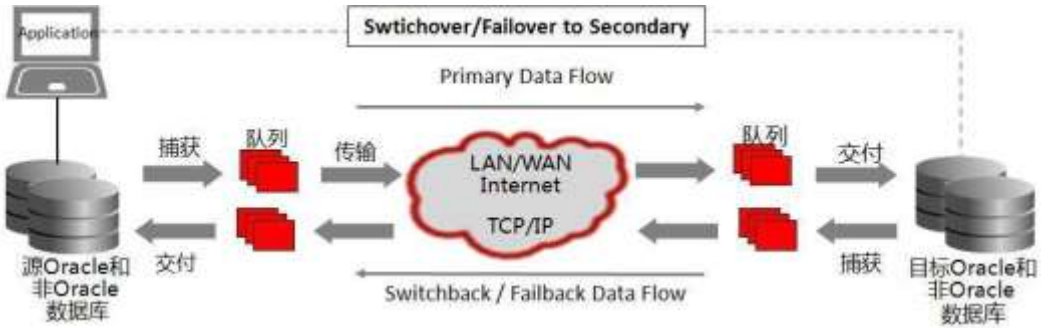
2.4 数据库容灾场景

本章节描述Oracle Data Guard 和GoldenGate 的容灾方案架构。

2.4.1 方案架构

2.4.1.1 服务器异构

GoldenGate 使用Active-Passive 模式来实现主备容灾，同时可以在备库分担查询负载压力。配置Active-Passive 模式的目的是为了主库因计划和非计划的中断后可以失败转移到一个完整复制主库数据的备份数据库上，业务架构如下：

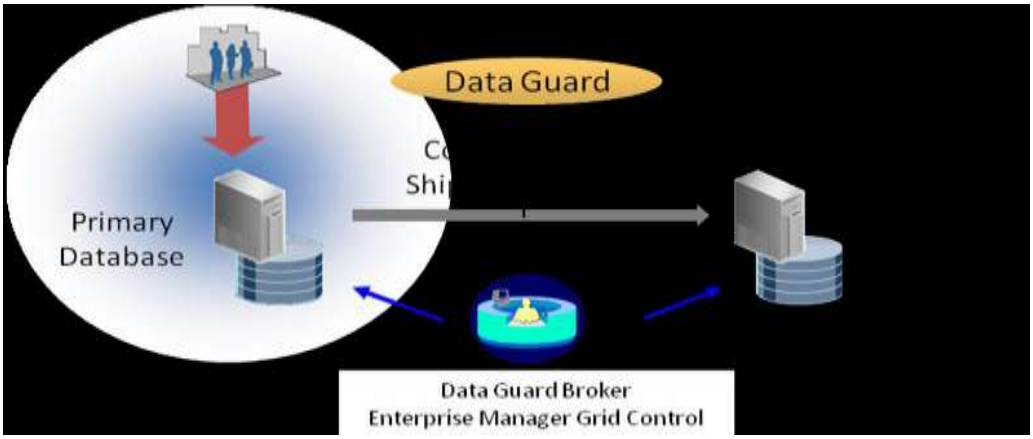


在这种配置中，备库拥有一个不活动的Extract group 和一个data pump。这些进程组在用户应用因switchover 或failover 切换到这个备用系统前都应保持stopped 状态。当用户活动切换到备用系统时，这些进程组开始捕获事务并将其写入到本地的trail 中，这些数据存储在磁盘上直到主库恢复使用。

在主库失败的案例中，Oracle GoldenGate 的Manager 和Replicat 进程与一个数据库协同工作，在主系统恢复后从备库将数据恢复到主库，使两个系统相等。在适当的时候，将用户移回主系统，Oracle GoldenGate 再一次被配置为准备模式，以准备将来的failover。

2.4.1.2 服务器同构

Oracle Data Guard 物理备用数据库通过重做应用技术进行维护并与生产数据库保持同步。生产数据库的重做数据随附在物理备份中，该物理备份使用介质恢复将更改从重做数据应用到备用数据库。使用重做应用，备份数据库在物理上与生产数据库保持一致。物理备用数据库可帮助企业有效降低灾难和数据错误的风险。在发生错误或灾难时，物理备用数据库将被打开，并用来为应用和最终用户提供服务。典型的Data Guard配置如下图所示



该方案支持实时查询的物理备用(Physical Standby with Real Time Query)，自动故障切换和转换，自动修复数据库坏块等功能。

2.4.2 方案设计

2.4.2.1 应用数据一致性设计

Data Guard

Data Guard 是通过传输和运行数据库日志文件，来保持生产和容灾数据库的数据一致性。物理standby 数据库从物理上提供了与生产数据库在数据块级的一致性镜像。物理standby 数据库通过Redo Apply 技术来保障数据镜像能力。一旦数据库因某种情况而不可用时，standby 数据库将正常切换或故障切换为新的生产数据库，以达到无数据损失或最小化数据损失的目的，为业务系统提供持续的数据服务能力。

GoldenGate

Oracle GoldenGate 软件架构包含三个主要组件：捕获、跟踪文件和交付。这种模块化方式让每个组件可以独立运行，从而加快数据复制并确保数据完整性。

捕获模块只移动提交的事务，过滤掉中间活动和回滚的操作。这不但减少了基础架构负载，还避免了潜在的数据不一致。通过事务组合和可选的压缩特性，该模块还可获得进一步的优化。

交付模块按照每个事务提交的顺序及事务在源数据库中的事务环境将其应用到目标数据库，从而确保目标数据库的一致性和引用完整性。Oracle GoldenGate 交付模块从最新的跟踪文件读取更改的数据，然后使用相应关系数据库管理系统的原生SQL 将这些数据应用到目标数据库。交付可适用于任何使用开放数据库连接的数据库。交付模块按照每个事务提交的顺序及事务在源数据库中的事务环境将其应用到目标数据库，从而确保目标数据库的一致性和引用完整性。

2.4.2.2 RPO&RTO

Data Guard

灾难的影响通常根据恢复点目标（RPO — 即在出现灾难时，业务可以经受得住丢失多少数据）和恢复时间目标（RTO — 即在出现灾难时，业务可以经受得住停机多长时间）来衡量。使用Oracle Data Guard，当结合使用最大保护模式（确保即使在出现灾难时也不会丢失数据）和实时应用时，企业在出现灾难时不仅获得了零数据丢失的好处，同时还获得了最少停机时间的好处。

Oracle Data Guard 提供三种高水平的数据保护模式来平衡成本、可用性、性能和事务保护。

最大保护：零数据丢失；通过LGWRSYNC；最

高可用性：零数据丢失；通过LGWRSYNC；

最高性能：最小数据丢失，通常从0 到几秒；通过LGWR ASYNC 或ARCH。

对于RTO，使用Data Guard 快速启动故障切换（fast-start failover）特性来自动将故障数据库Failover 到远程容灾节点，RTO 小于30 秒。

GoldenGate

GoldenGate 可以提供秒级的大量数据实时捕捉和投递，异步复制方式，但是无法实现同步复制。所以GoldenGate 的RPO 可以达到近似于零，而不能达到0。RTO 则根据数据复制的情况，在秒级到分钟级之间。

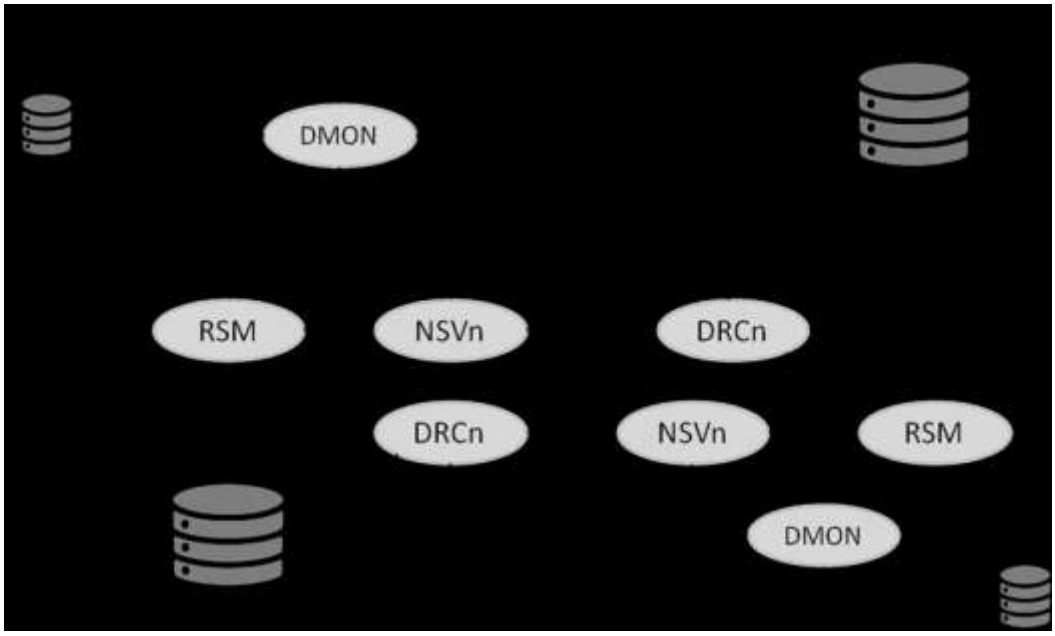
2.4.2.3 容灾管理

Data Guard Broker

DataGuard broker 是9i 开始引进的，是随同Oracle 企业版(Enterprise Edtion，EE)跟DataGuard 一起内置的管理监控工具。通过DataGuard Broker，DBA 能简化部署、监控DataGuard，以及进行主备角色切换。

Broker 由三部分组成：主备库各自的Broker 后台进程，一系列的配置文件和命令行工具dgmgrl。需要注意的是，如果使用Broker 管理或配置DataGuard，则不应该使用SQL 命令再进行配置管理，否则会导致Broker 参数配置或者数据库配置不一致的情况。

Broker 处理流程如下图所示，这些进程是在数据库及Broker 启动自动运行，DBA 无法介入。



进程说明如下：

DataGuard Monitor(DMON)：Broker 主进程，所有进程中，最先启动DMON，它的主要任务是协调Broker 的其他进程，包括维护Broker 配置文件，这个进程通过DG_BROKER_START 进行激活和取消。

Broker Resource Manager(RSM)：RSM 处理Broker 在任何一个数据库中进行配置所执行的所有SQL 命令。

DataGuard Net Server (NSVn)：NSVn 负责跟远程数据库连接。当创建配置文件时候需要指定连接符，NSVn 就是通过这个连接符去连接远程数据库。

DRCn：网络接收进程，负责建立与source 端NSVn 进程的联系。NSVn 到DRCn 类似于在日志传输时的LogWriter Network Service(LNS)到Remote File Server(RSF)的连接，当日志传输时，Broker 需要发送数据或者SQL 命令，会使用NSV 及DRC 进行连接，这些连接只会在需要的情况下启动。

Configuration files：配置文件为在主库和备库上的常规二进制文件，它储存了DataGuard 的所有参数配置。可以存储在OS 文件系统上或者ASM 存储上，它是通过Broker 去进行维护，而不是认为去维护，类似SPIFLE。

在Broker 配置中，主库的DMON 进程对DG 配置有拥有权，不管在哪一个节点上对配置进行修改，都是在主节点完成。任何时候DMON 需要执行一些SQL，它都会调用主库的RSM 进行辅助。如果SQL 执行目标是主库，它会直接执行；如果SQL 执行目标是备库，RSM 会要求NSVn 进程将这些SQL 命令传送到备库端，通过DRCn 进程执行。

使用Broker 可通过Enterprises Manager Grid Control 或者命令行工具dgmgrl。

GoldenGate GGSCI

GoldenGate的命令行管理界面-ggsci，它是GoldenGate配置和管理的基本工具。 当客户购买了GoldenGate软件后，可以通过其账户在edelivery.oracle.com网站上下载，一般安装包是zip或者tar文件。使用工具解压开GoldenGate安装包以后，在安装目录下可以看到ggsci的可执行文件，立即执行它即可：

[oracle@aprac1 /ogg]#./ggsci

Oracle GoldenGate Command Interpreter for Oracle

Version 11.2.1.0.1 OGGCORE_11.2.1.0.1_PLATFORMS_120423.0230

AIX 5L, ppc, 64bit (optimized), Oracle 11g on Apr 23 2012 05:03:51

Copyright (C) 1995, 2012, Oracle and/or its affiliates. All rights reserved.

如执行成功则会出现以上类似信息，可以看到GoldenGate 版本、操作系统和数据库版本等相关信息，并出现命令提示符，表示此时已进入GoldenGate 命令行管理界面，可以在此提示符下输入管理和配置命令。除命令行管理工具外，GoldenGate 也提供了图形管理界面Oracle GoldenGate Director。

2.4.3 方案亮点

Data Guard 主备数据中心

- 在任何中断意外的情况下保持生产数据库的高可用性，并在计划维护期间最大限度地减少停机时间
- 将只读负载分流到同步备用数据库，从而增加主数据库的容量并提高其性能，并且ActiveDataGuard保证只读查询与对主数据库执行的查询具有相同的读取一致性保证
- 保证数据库不受到坏块的影响，自动修复坏块对应用程序和用户来说是透明的
- 当实施Oracle补丁集或升级到新的Oracle版本时，DataGuard 备用数据库还可以用于最大限度地减少计划停机时间

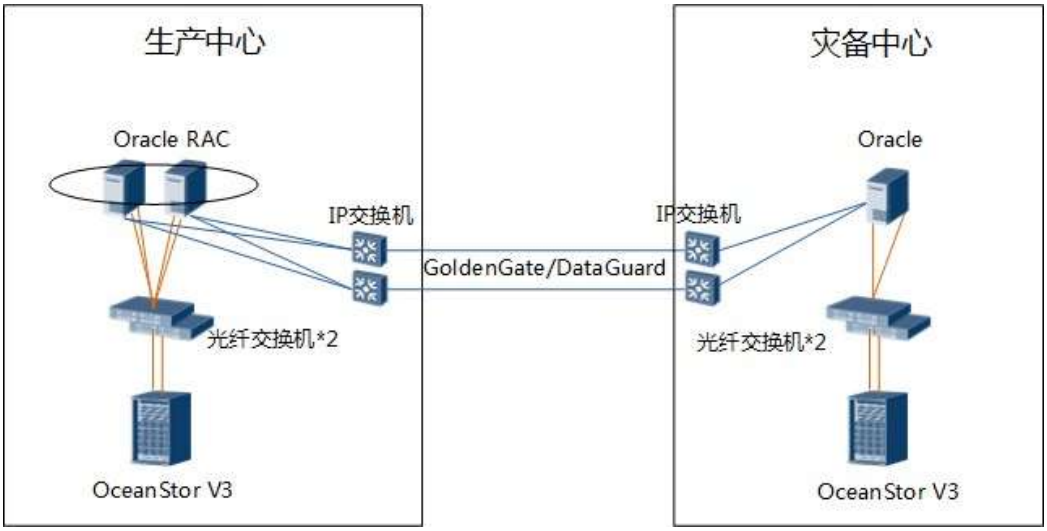
GoldenGate 主备数据中心

- 保证主备环境间事务数据的实时，低影响的捕获和传输，保持事务的完整性
- 支持异构源和目标，满足不同的延迟需求
- 在容灾站点支持实时查询，节省了生产站点的开销，提高了生产站点的性能
- 在线跨平台数据迁移和数据库升级，最大限度减少宕机时间

2.4.4 典型配置

2.4.4.1 组网

图2-11 方案架构图



如图2-11 所示，Site 1 为生产数据中心，Site 2 为灾备数据中心，实际部署时，可以互为灾备中心，以提高设备利用率。

- 数据复制通过GoldenGate或者DataGuard主机层复制完成；
- 灾备数据中心可以提供读功能，进行查询操作。

2.4.4.2 规格

Data Guard

- DataGuard物理备用数据库透明地支持所有Oracle数据类型、数据库特性和应用程序。
- DataGuard可在单一配置中支持多达30个备用数据库。

GoldenGate

- GoldenGate支持Windows 2000,2003,2008,XP，Sun Solaris，HP-UX，IBMAIX，Linux(Novell SuSE，RedHat Enterprise，Oracle Enterprise Linux)，HP TRU64，HP OpenVMS，HPNonStop，IBMz/OS 等32和64位操作系统平台，支持上述平台的主流版本。

- GoldenGate软件在进行数据复制时，不会受到数据库类型和版本的限制，基于该软件的数据复制方案不仅能够支持同构环境下的数据备份，还可以支持在异构的软硬件环境之间实现复制，如在不同厂商主机和存储之间、不同操作系统之间甚至不同数据库之间实现数据复制。OracleGoldenGate支持以下数据库的复制：HP NonStop SQL/MX，IBM DB2，IBM DB2 for i，IBM DB2 for z/OS，Microsoft SQL Server，MySQL Database Server，Oracle Database，PostgreSQL，Sybase Adaptive Server Enterprise，Teradata。

3 关键技术

3.1 异构虚拟化技术

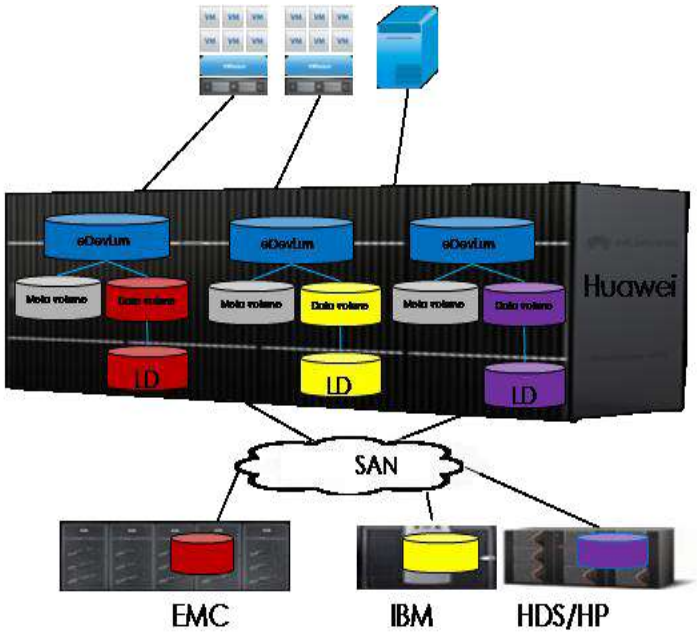
由于不同存储使用的容灾复制技术不同，不能直接进行容灾复制，这意味着用户一旦更换存储型号，如何选择利旧方案保护既有投资便成为一个首要面临的问题。

OceanStor V3 存储系统提供的SmartVirtualization 特性可以有效地解决客户遇到的问题。用户可以集中管理V3 存储系统（以下简称为本端存储系统）及第三方存储系统（以下简称异构存储系统）中的存储资源，同时用户仍然可以使用旧存储系统的存储资源，降低对原有投资的浪费。

SmartVirtualiztion 工作原理

异构虚拟化的工作原理就是把异构阵列映射到本端阵列的LUN，作为可为本端阵列提供存储空间的逻辑盘LD，再在该逻辑盘LD 上创建为可对主机映射的异构设备LUN eDevLun，逻辑盘LD 为eDevLun 的数据卷Data Volume 提供了全部的数据存储空间，eDevLun 的元数据卷Meta Volume 的存储空间由本地存储提供。异构虚拟化可保证外部LUN 数据完整性不被破坏。eDevLun 与原异构LUN 具有不同的全球唯一名称WWN。

图3-1 SmartVirtualization 工作原理示意图



由于eDevLun 与本地LUN 基本上具有相同的LUN 属性，所以，通过SmartMigration 技术为异构LUN 提供在线LUN 迁移功能，通过HyperReplication/S 技术为异构LUN 提供同步远程复制功能，通过HyperReplication/A 技术为异构LUN 提供异步远程复制功能，通过HyperSnap 为异构LUN 提供异构快照功能。同时通过SmartQos 和SmartPartion 技术，以及CACHE 可回写策略提升异构LUN 性能。

SmartVirtualization 技术特点

- 可靠的兼容性

由于不同厂商的异构阵列在实现的过程中，对SCSI 协议的理解和遵从性都不一样，导致接管异构阵列带来很大的兼容性挑战。异构虚拟化技术通过针对不同的阵列做兼容性的差异化处理，能很好的识别和处理与异构设备间的各种兼容性问题，如及时识别和处理异构设备的LUN 路径故障，以此达到与异构阵列间高可靠的兼容。对支持的异构设备，都通过了华为兼容性实验室认证。

- 后端多路径

异构虚拟化技术通过后端多路径软件支持以多路径冗余方式连接异构阵列，防止与异构阵列间的物理链路连接出现因为单个链路故障而导致业务中断，并在这些路径中选择最合适的路径进行IO 下发，并提供多种后端多路径负载均衡选路算法。

- 异构LUN迁移

异构虚拟化技术通过与SmartMigration 技术结合，为异构阵列间的LUN 提供高可靠的在线LUN 间数据迁移，异构LUN 数据迁移过程中不需要主机中断在线业务。

- 异构远程复制

异构虚拟化技术通过与HyperReplication 技术结合，为异构LUN 提供同步和异步远程复制功能。其主要特点如下：

1. 异构同步复制RPO为0；异构异步复制通过多时间片的缓存技术使RPO可达到秒级。
2. 支持分裂模式，在分裂模式下，生产主机的写请求只会写到主LUN，并通过差异日志来记录主，从LUN数据之间的差异。
3. 快速响应故障和故障恢复，系统在检测到系统故障情况下，复制进入断开状态，在断开状态下，生产主LUN记录IO差异，故障恢复后，根据策略自动完成主从LUN 数据差异同步。
4. 支持异构远程复制从LUN可写。从而实现用户在不影响主LUN业务的情况下，通过从LUN进行数据分析和挖掘等用户使用场景。
5. 支持异构远程复制的LUN主从切换。
6. 支持异构远程复制LUN一致性组功能。

- 异构快照

异构虚拟化技术通过和HyperSnap 技术结合，为异构LUN 提供快照功能。快照采用COW（Copy-On-Write）技术，保证快照不影响原LUN 读IO 性能。具有快照瞬间完成，占用磁盘空间小等特点。另外还支持对异构快照创建多个副本，用于进行数据分析和数据挖掘等应用。

3.2 卷镜像技术

随着企业的高速发展，业务数据量不断攀升，数据里记录着企业的运营情况，同时给决策者提供重要信息，成为企业最宝贵的财富之一。如何保证数据的可靠性成为了一个重要的挑战。并且随着异构虚拟化的广泛使用，如何提升对异构接管LUN 的可靠性保证，提出了不同于传统RAID 方式的新需求，卷镜像就是为了解决这些需求的一个可行办法。

通过使用卷镜像，一个LUN 可以拥有多个物理副本。每个副本的空间可以来源于本地存储池，也可以来源于外部LUN。每个副本都具有与LUN 相同的虚拟容量。当服务器对镜像LUN 执行写操作时，系统会将数据同时写入每个副本。当服务器对镜像LUN 执行读操作时，系统会选取其中一个副本进行读取。如果其中一个镜像副本暂时不可用（例如，由于提供存储池的存储系统不可用），那么服务器仍然可以访问LUN。系统会记住执行写操作的LUN 区域，并会在镜像副本恢复后，对这些区域进行再同步。

OceanStor 存储系统的卷镜像软件名称为HyperMirror。

HyperMirror 工作原理

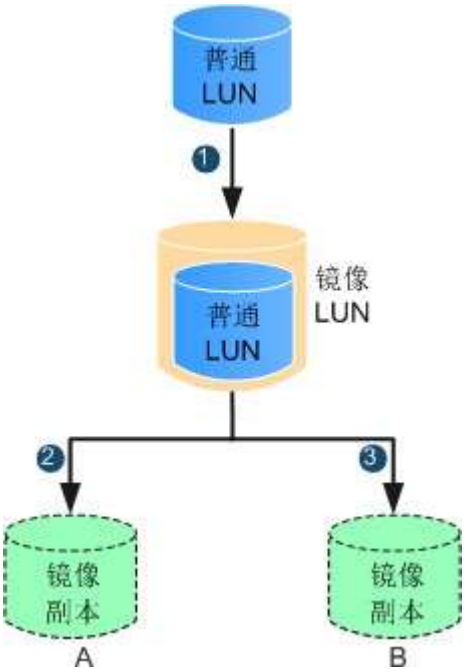
卷镜像的主要用途是为本地LUN 或外部LUN 提供多个可用的镜像副本。如果其中一个镜像副本故障不可用，主机仍然可以正常访问LUN，主机侧业务无任何影响；同时，待故障镜像副本从故障中恢复后，镜像副本会自动同步镜像LUN 的数据，最终达到镜像副本与镜像LUN 的数据完全一致。

卷镜像的实现过程分为三个阶段：创建镜像LUN、同步和分裂。

- 创建镜像LUN

镜像LUN 的创建过程如图3-2 所示。

图3-2 镜像LUN 的创建过程



1. 对一个普通LUN（本地LUN或外部LUN）执行创建镜像LUN操作，此时镜像LUN 完全继承普通LUN 的存储空间；同时继承普通LUN 的基本属性和业务，主机侧不中断业务。
2. 创建镜像LUN过程中会在本地自动生成一个镜像副本A，普通LUN变为镜像LUN，并将数据存储空间交换到镜像副本A，镜像LUN 从镜像副本A 中同步数据。
3. 此后需再给镜像LUN添加一个镜像副本B，创建之初从镜像副本A同步数据。此时普通LUN具有空间镜像功能，同时拥有镜像副本A 和镜像副本B两份镜像数据。

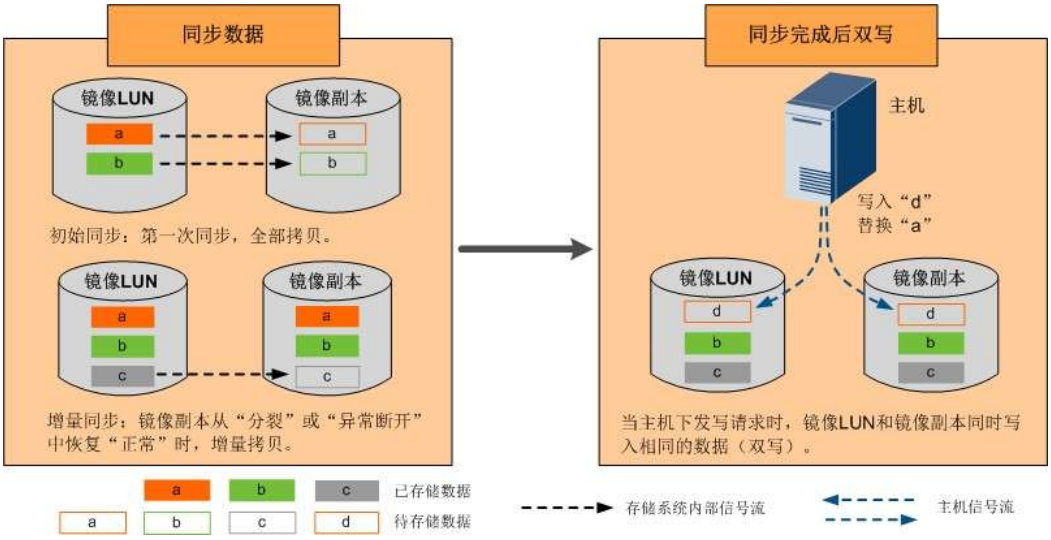
   镜像LUN 创建完成后，主机下发I/O 的情况：

4. 当主机对镜像LUN下发读请求时，存储系统会以轮询方式在镜像LUN和镜像副本之间进行读操作。当镜像LUN或者某个镜像副本故障时，主机侧业务不受影响。
5. 当主机对镜像LUN下发写请求时，存储系统会以双写方式对镜像LUN和镜像副本进行写操作。

- 同步

同步过程的原理如图3-3 所示。当一个副本从故障恢复或从数据不完整恢复到数据完整的过程中，增量从完整的镜像副本同步数据，最终达到镜像副本间的数据完全一致。

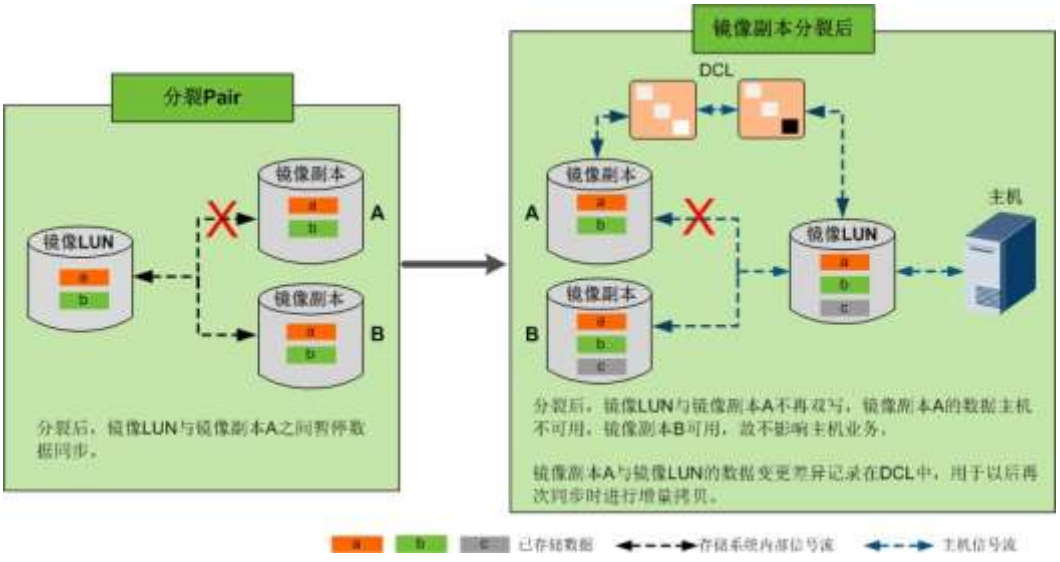
图3-3 镜像同步过程原理



- 分裂

在业务需要隔离一个副本时，可执行分裂操作，将副本从镜像中隔离，此时镜像不具备数据镜像功能，同时镜像副本记录此后的数据差异，当需要重新建立镜像关系时，根据差异增量同步差异数据。

图3-4 镜像分裂过程原理



HyperMirror 技术特点

- 存储系统内数据可靠性

通过对业务LUN 创建卷镜像，使其同时拥有两分完全独立的数据副本，其中一份数据副本故障，不影响主机业务的运行，大大提高的数据的可靠性。

- 异构阵列的可靠性

通过SmartVirtualization 特性，将异构阵列的LUN 接管，对其创建镜像LUN，创建本一个本地镜像副本，防止由于异构阵列不稳定或链路异常导致的业务中断。

- 主机性能影响小

镜像数据副本位于业务LUN 的CACHE 之下实现，且镜像空间之间实现并发写和轮询读技术，不影响主机业务性能。

- 主机业务连续性

通过在线将正在运行业务的LUN 创建镜像副本，主机业务不感知LUN 数据空间的变化。

3.3 存储远程复制技术

华为公司HyperReplication 是基于华为OceanStor 存储系统的远程复制特性，可以在OceanStor 存储系统之间实现同步或异步的数据复制，支撑用户构建同城或异地的容灾解决方案。

HyperReplication 至少需要在两套OceanStor 存储系统上运行，这两套存储系统可以放在同一个机房、同一个城市或者相距上千公里的两地。一般提供生产业务数据访问的一端称为主端，提供数据备份的另一端称为从端。

HyperReplication 包含以下几种模式：

- LUN同步远程复制

  主从端的LUN 数据实时同步，灾难发生时无数据丢失，代价是生产业务的性能受到主从端间数据传输时延影响。

- LUN异步远程复制

  主从端的LUN 数据周期性同步，生产业务的性能不受主从端间数据传输时延影响，代价是灾难发生时可能会有一定数据丢失。

- 文件系统异步远程复制

  主从端的文件系统数据周期性同步，生产业务的性能不受主从端间数据传输时延影响，代价是灾难发生时可能会有一定数据丢失。

HyperReplication 对LUN 同步远程复制、LUN 异步远程复制提供基于存储阵列的一致性组功能，保证跨多个LUN部署的应用在容灾复制时的一致性（consistency）。一致性组功能会在复制时保证跨多个LUN 的主机写IO 的依赖关系，从而保证灾备端的LUN 间的数据一致性。

HyperReplication 提供基于FC 网络和IP 网络的复制。主从端存储系统间可以经FC 链路传输数据，也可以经IP 链路传输数据。

3.3.1 LUN同步远程复制

简介

HyperReplication LUN 同步远程复制提供对LUN 的近距离的数据容灾功能。它适用于需要在同城范围内的进行容灾，不容许有数据丢失的场景。

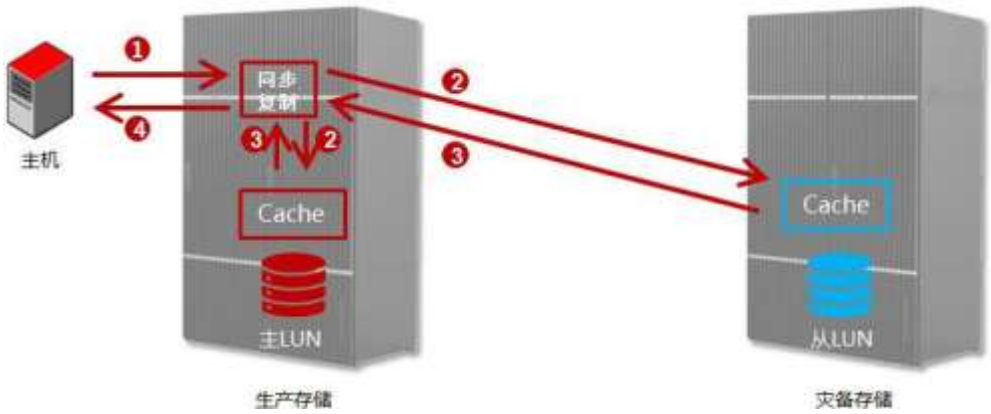
HyperReplication LUN 同步远程复制对于每个主机的写IO，都会同时写到主LUN 和从LUN，直到主LUN 和从LUN 都返回处理结果后，才会返回主机处理结果。因此，同步远程复制可以实现RPO 为0。

基本原理

HyperReplication LUN 同步远程复制的工作原理如下：

1. 生产中心的主LUN和灾备中心的从LUN建立同步远程复制关系，启动初始同步，将主LUN数据全量拷贝到从LUN；
2. 初始同步中主LUN收到主机写请求也会同样写到从LUN；
3. 初始同步完成以后，进入正常状态，此时主、从LUN数据相同。正常状态下的IO 处理流程如下：
   - 生产存储收到主机写请求。HyperReplication将该请求记录日志。日志中只记录地址信息，不记录数据内容。
   - 将该请求写入主LUN和从LUN。通常情况下LUN是回写状态，数据会写入Cache。
   - HyperReplication等待主LUN和从LUN的写处理结果都返回。如果都写成功，清除日志；否则保留日志，进入异常断开状态，后续启动同步时重新复制该日志地址对应的数据块。
   - 返回主机写请求处理结果，以写主LUN的处理结果为准。

图3-5 LUN 同步远程复制原理



3.3.2 LUN异步远程复制

简介

HyperReplication LUN 异步远程复制提供对LUN 的远距离数据容灾功能。它适用于需要在跨异地的数据中心间进行容灾，同时降低对生产业务的性能影响的场景。

HyperReplication LUN 异步远程复制基于多时间片缓存技术，周期性的同步主、从LUN 的数据，上一次同步以来主LUN 上发生的所有变化会在下一次同步时写到从LUN 上。

基本原理

OceanStor 存储系统异步远程复制采用了创新的多时间片缓存技术（专利号：PCT/CN2013/080203），其实现原理如下：

1. 与同步远程复制类似，当主站点的主LUN和远端复制站点的从LUN建立异步远程复制关系以后，默认情况下会启动一个初始同步，将主LUN数据全量拷贝到从LUN；
2. 初始同步完成后，从LUN数据状态变为完整（即从LUN为主LUN的过去某个时刻的一致性拷贝），然后开始按照下面的流程进行I/O处理：

图3-6 LUN 异步远程复制原理

- 每当间隔一个同步周期（由用户设定，范围为3s～1440min），系统会自动启动一个将主站点数据增量同步到从站点的同步过程（如果同步类型为手动，则需要用户来触发同步）。每个复制周期启动时在主LUN（LUNA）和从LUN（LUNB）的缓存中产生新的时间片（TP_{N+1}和TP_{X+1}）；
- 主站点接收生产主机写请求；
- 主站点将写请求的数据写入Cache时间片TP_{N+1}中，立即响应主机写完成；
- 同步数据时，读取前一个周期主LUN（LUNA）Cache时间片TP_N的数据，传输到从站点，写入从LUN（LUNB）Cache时间片TP_{X+1}中；若主站点Cache 写缓存达到高水位时会自动将数据从Cache写入硬盘中，此时时间片TP_N的数据会在盘上生成快照，同步时已写入硬盘的数据从快照中读取并复制到从LUN（LUN B）；

- 同步数据完成后，按照刷盘策略将主LUN（LUN A）和从LUN（LUNB）Cache 中时间片TP_N和TP_{X+1}的数据下盘（生成的快照自动删除），等待下一个同步的到来。
  - 时间片：在Cache中管理一段时间内写入数据的逻辑空间（数据大小没有限定）

说明

- 在低RPO的应用场景下，异步远程复制周期很短，OceanStor存储系统Cache中能缓存多个时间片中的全部数据；如果主机业务带宽或容灾带宽出现异常或故障，造成复制周期变长或中断，此时Cache中的数据会按照刷盘策略自动刷盘并进行一致性保护，复制时再从盘上进行读取。

3.3.3 文件系统异步远程复制

简介

HyperReplication 文件系统异步远程复制提供对文件系统的远距离数据容灾功能。它适用于需要在跨异地的数据中心间进行容灾，同时降低对生产业务的性能影响的场景。

HyperReplication 文件系统异步远程复制基于文件系统对象层，周期性的同步主、从FS的数据，上一次同步以来主FS 上发生的所有变化会在下一次同步时写到从FS 上。

HyperReplication 文件系统异步远程复制是对文件系统级别的复制，它将生产端的文件系统的全部内容复制到灾备端的文件系统中。目前不支持对目录级别的复制。

基本原理

- 基于对象层的复制

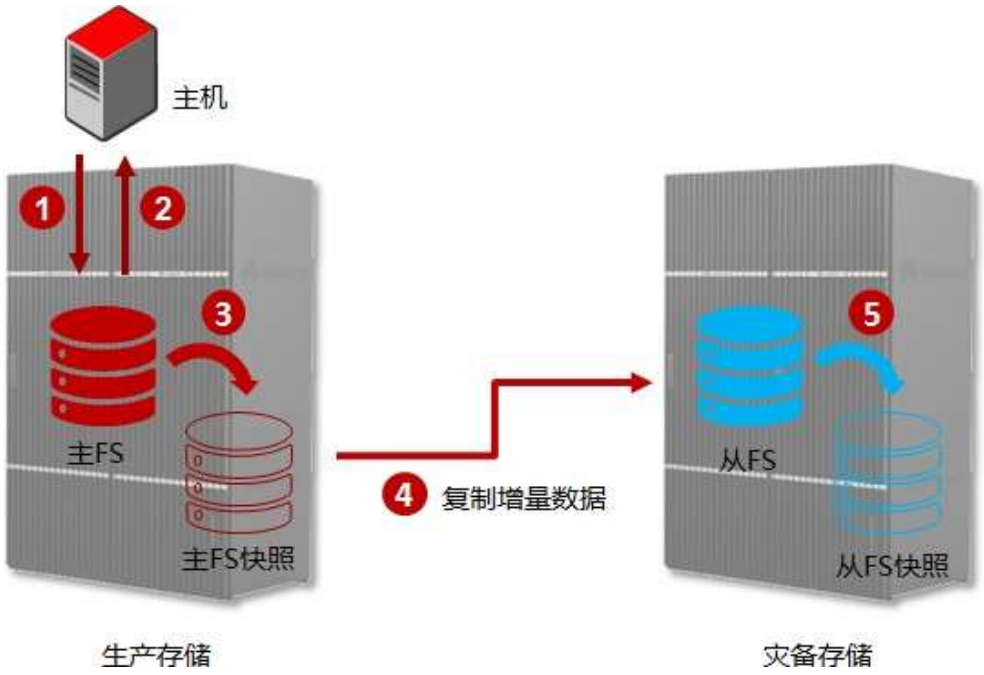
HyperReplication 文件系统异步远程复制采用基于对象层的方式进行数据复制。文件系统的所有内容，比如文件、目录、文件属性，都是由对象构成。基于对象层的复制直接将对象从主文件系统复制到从文件系统，不需要关心复杂的文件层的信息，比如文件与目录间的依赖关系、各种文件操作，从而使复制变得更加简单高效。

- 基于ROW快照的周期性复制

HyperReplication 文件系统异步远程复制采用基于ROW 快照的周期性的方式进行数据复制。

异步复制中，主机数据写入主FS即可返回。写入的数据在后台周期性地复制到从FS。复制周期由用户设定，每个周期内数据的变化会记录增量信息，增量信息记录数据变化的地址，不会记录数据内容。每个周期开始时，文件系统异步远程复制创建主FS（主文件系统）的快照，根据上一周期复制完成到本周期开始这段时间内的增量信息，读取快照的数据复制到从FS，增量复制完成后，从FS 的内容与主FS 的快照内容相同，从FS 形成数据一致性点。每次周期复制过程中，当增量数据没有传输完成时，从FS 还不能构成完整的文件系统，因此每次周期复制完成时，从FS 形成数据一致性点后，会创建从FS 的快照，如果下一次周期复制过程中断（生产端发生故障、链路发生故障等原因），当用户需要使用从FS时，文件系统异步远程复制可以将从FS回滚到上个周期完成时的快照点，获得一致性数据。

图3-7 文件系统异步远程复制原理



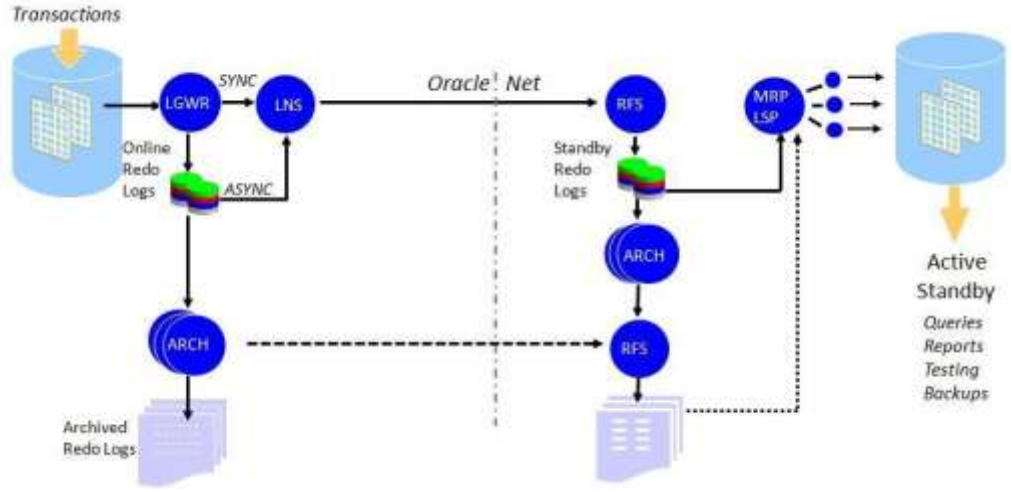
周期性复制可以提高复制效率和带宽利用效率。在一个周期中，如果主机重复写入相同地址的数据（比如对同一文件相同地址的重复修改），只需要将最后一次写入的数据进行复制。

文件系统及其快照都是采用ROW 方式处理数据写入（详细请参考文件系统技术白皮书、快照技术白皮书）,不管文件系统是否带有快照，数据都是写入新分配的地址空间，创建快照后几乎不会带来性能影响。因此，文件系统异步远程复制对生产业务的性能影响也很小。

3.4 数据库复制技术

Data Guard

图3-8 Data Guard 复制技术示意图

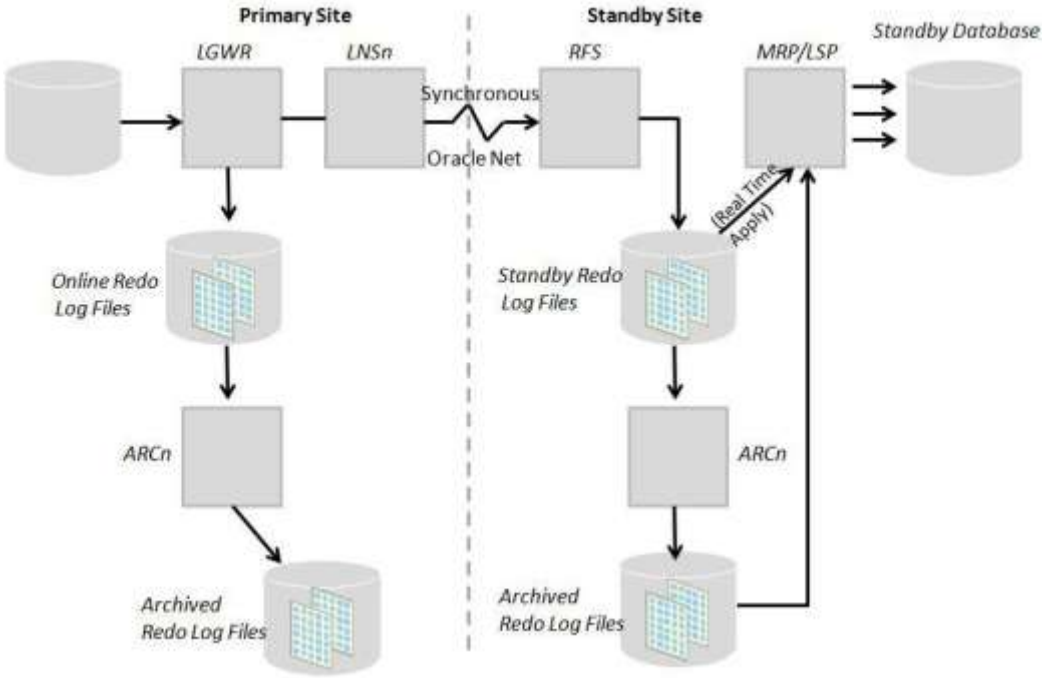


Data Guard 在主数据库上使用日志写入器网络服务器(LNS) 的专门的后台进程来捕获日志写入器写入的重做数据，并以同步或异步方式将重做数据传输到备用数据库。
LNS进程将日志写入器与传输开销及网络中断隔离开来。

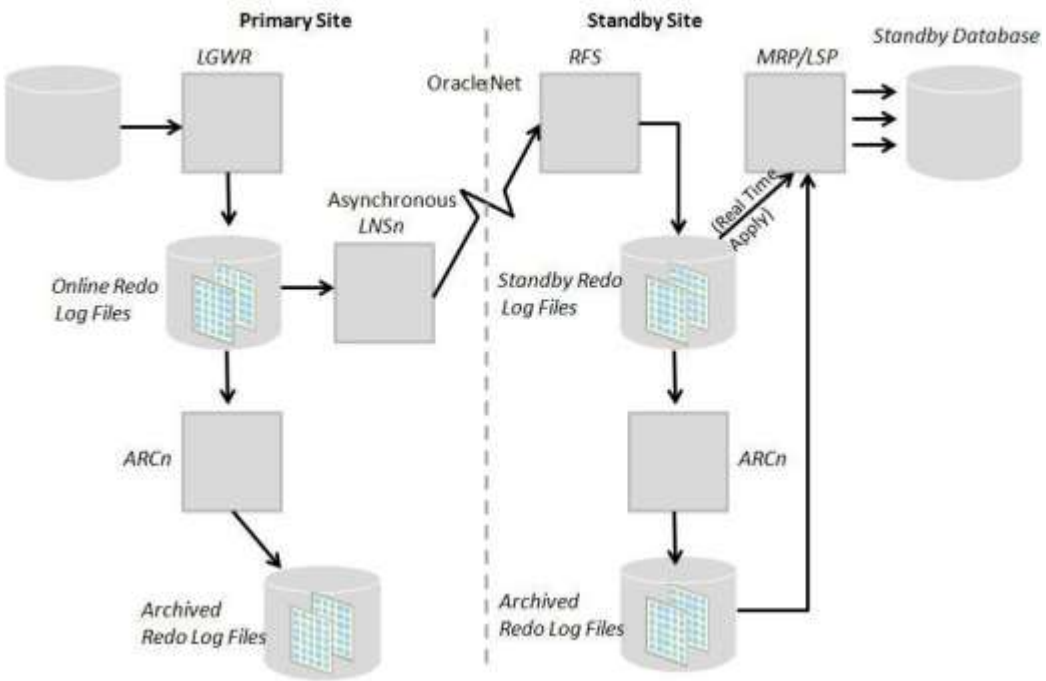
备用数据库接受来自主数据库LGWR 进程触发的日志信息并且通过RFS 进程写入到备用重做日志或者备用归档日志中。如果写到备用重做日志文件中，则当主数据库发生日志切换时，也会触发备用数据库上的备用重做日志的日志切换，并把这个备用重做日志归档。如果是写到备用归档日志，那么这个动作本省也可以看作是个归档操作。

备用数据库端的MRP(Managed Recovery Process)进程或者LSP 进程在备用数据库上应用这些日志，进而同步数据。

同步重做传输(SYNC) 要求主数据库上的日志写入器等待LNS 确认备用数据库已经接收重做数据并已将其写入备用重做日志，然后才能确认到客户端应用程序的提交。这确保了提交的所有事务都在磁盘上，并在备用位置受到保护。



异步重做传输(ASYNC) 不要求主数据库上的日志写入器等待备用数据库确认重做已被写入磁盘。确认到客户端应用程序的提交与重做传输是异步进行的。



此外，ASYNC 可以非常高效地将重做传输至备用数据库，以消除将导致网络吞吐量降低的网络确认的开销。

如果主数据库和备用数据库断开连接（网络故障或备用服务器故障），那么根据选择的保护模式，主数据库将继续处理事务并累积不能传输到备用数据库的重做数据积压，直到能建立一个新的网络连接为止（称为归档日志差异）。在这种状态下，DataGuard 持续监视备用数据库状态、检测连接重建时间、并自动重新同步备用数据库与主数据

库，以尽快将配置返回到受保护状态。始终使用归档器(ARCn) 进程来传输解析差异所需的日志文件。

Data Guard 提供了三种数据保护模式，以实现成本、可用性、性能和数据保护的平衡。

保护模式	主数据库出现故障时的数据丢失风险	重做传输
最大保护	零数据丢失	SYNC
最高可用性	零数据丢失—假设出现故障之前主数据库提交事务时的同步通信没有中断	SYNC
最高性能	最小数据丢失—只有几秒，具体取决于网络带宽	ASYNC

以下小节更详细地说明了这些保护模式。

最大保护

最大保护提供最高级别的数据保护。最大保护使用SYNC 重做传输将重做记录同步传输到备用数据库。只有当Data Guard 确认事务数据安全保存到至少一台备用服务器的磁盘上之后，主数据库上的日志写入器进程才会确认到客户端应用程序的提交。由于重做传输的同步特性，最大保护模式可能影响主数据库的响应时间。通过配置一个带宽足够应付高峰事务负载的低延迟网络，可以将这种影响降至最小。例如，金融机构和彩票系统就部署了这种最高级别的数据保护模式。强烈建议，最大保护模式应至少配置两个备用数据库。这样，如果一个备用目标出现故障，另外一个可用的目标仍可向主数据库做出确认，而不中断生产。考虑使用这种保护模式的用户必须接受的事实是，如果至少一个备用数据库无法返回重做已被接收并写入磁盘的确认，主数据库将停滞（并最终崩溃）。最大保护的规则规定这一行为的目的是在有多个故障的情况下
（例如，主数据库和备用数据库之间的网络出现故障，然后又有一个事件导致主站点发生故障）实现零数据丢失。如果您无法接受这一要求，并希望在得到零数据丢失保护的同时，主数据库即使在备用数据库无法确认数据受到保护的情况下也能保持可用，可以使用接下来讨论的最高可用性保护模式。

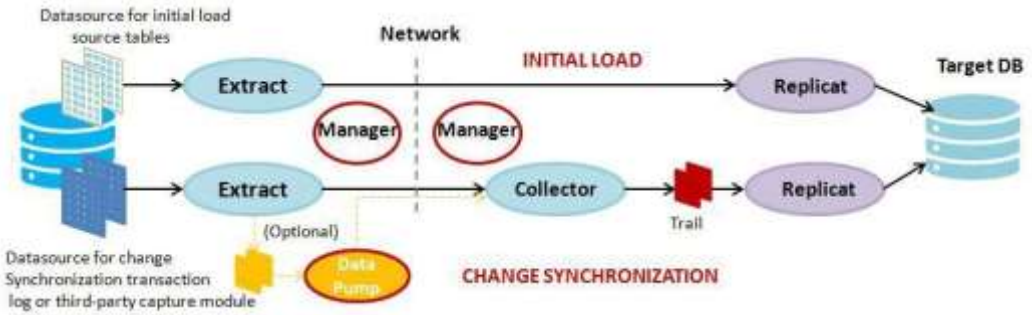
最高可用性

最高可用性在保持主数据库可用的同时，提供级别稍低的数据保护。与最大保护一样，重做数据通过SYNC 重做传输服务同步传输到备用数据库。只有当Data Guard确认事务数据在至少一台备用服务器的磁盘上可用之后，日志写入器才会确认主数据库已提交事务。但与最大保护模式不同的是，如果最后参与的备用数据库不可用（例如出现网络连接问题或备用故障），主数据库处理将继续进行。断开连接时，与主数据库相比，备用数据库可能暂时滞后。而当再次建立连接时，Data Guard 将自动重新同步备用数据库和主数据库。由于同步的重做传输，这种保护模式可能影响响应时间和吞吐量。通过配置一个带宽足够应付高峰事务负载的低延迟网络，可以将这种影响降至最小。最高可用性模式适用于想要确保获得零数据丢失保护但不想让生产数据库受网络或备用服务器故障影响的企业。如果在最初的网络或备用故障得到解决和配置重新同步之前，又一个故障随后影响了生产数据库，则会发生数据丢失，采用这种保护模式的企业要接受这样的事实。

最高性能

最高性能是默认的保护模式。与最高可用性模式相比，它提供了稍差的数据保护，但可为主数据库提供更高的性能。在这种模式下，当主数据库处理事务时，重做数据通过ASYNC 重做传输异步传输到备用数据库。主数据库上的日志写入器不用等待备用数据库确认即可确认到客户端应用程序的提交。这消除了网络往返行程和备用磁盘输入/输出可能生成的主数据库上的响应时间开销。如果任何备用目标不可用，处理将在主数据库上继续进行，对性能只有很小的影响或没有影响。在正常操作中，可能丢失的数据量只限于在主数据库和备用数据库之间传输的数据量 — 这个量由处理主数据库生成的重做数据量的网络容量决定。如果有足够的带宽，可能丢失的总数据量非常小或为零。当主数据库上的可用性和性能比丢失少量数据的风险更重要时，应该使用最高性能模式。这种模式还适合于客户网络固有延迟高到可能限制同步重做传输的适用性时的DataGuard 部署。

GoldenGate



Extract：

此进程运行在Source System，其主要用于以下目的：

- InitialLoads：初始化加载数据，从源对象直接捕获提取一个当前的、静态的数据集。
- ChangeSynchronization：改变同步，在初始化同步完成后Extract捕获DML和DDL 操作，使Source Database 与另一个数据集保持同步。

  Data Pump

  此组件是Oracle Golden Gate 配置在Source 端的二级Extract Group，如果Data Pump 没有被使用，那么Extract 要负责将捕获的操作传送到Target 端的Remote Trail。在使用Data Pump 的典型配置中，Primary Extract 进程只负责将捕获的操作写入到Source Database 的Trail 文件中，Data Pump 负责读取这些Trail 通过网络将其传送到Target Database 的Remote Trail。Data Pump 增加了存储的灵活性，并且实现了Primary Extract 进程与TCP/IP 的有效隔离。

  通常情况下，Data Pump 能够执行过滤、映射、转换等操作，或者也可以将其设置为Pass-through 模式，这种模式下数据被动传输，这种模式增大了Data Pump 的吞吐量，因为可以绕过查找对象定义的功能。

  Replicat

  此组件运行在Target 端，它读取系统上的Trail 文件，并重新架构DML 和DDL 操作，将它们应用到目标数据库。你以配置Replicat 进程用于以下目的之一：

- InitialLoads：用于初始化数据加载。

- ChangeSynchronization：当配置为ChangeSynchronization时，Replicat进程使用本地数据库接口或ODBC，依赖于数据类型，将从Source复制来的数据操作应用到Target；为了保证数据的完整性，Replicat按照源端提交的方式应用复制操作。

  可以配置多个Replicat 进程并发来提高吞吐量，为了保证数据的完整性，不同的Replicat 进程处理不同的对象集，需要为每个Replicat 进程分配一个Group。

  可以延迟复制使它在应用复制的操作到目标之前等待一个你指定的时间。这个延迟是可以被设计的，比如预防错误的SQL 传播、控制来自不同时区的数据的到达、或者给予另一个计划的事件发生的时间。这个延迟的长度由参数DEFERAPPLYINTERVAL 控制。

  Trails 文件

  为了支持对数据库改变连续的提取和复制，Oracle Goldengate 将捕获改变的记录临时的存储在磁盘上的一系列的文件中，这一系列文件叫做trail。Trail 能够出现在source 端、target 端、intermediary 端或者他们的组合，这依赖于你怎么配置Oracle Goldengate。在local system 它被称为extract trail(local trail)。在remote system 它被称作remote trail。

  使用trail 作为存储，Oracle Goldengate 支持数据准确性和容错。使用trail 也使提取和复制活动彼此独立的发生。由于这些进程的分离，对于数据的处理和交付你有了更多的选择。比如替代提取和复制连续的发生，你可以让提取连续发生，当target 需要的时候再进行复制处理。

  Checkpoints

  为了恢复的目的，Checkpoints 进程存储当前写和读到磁盘。Checkpoints 是为了确认需要同步而被标记的数据的改变是否被Extract 所捕获，并且被Replicat 所应用，并且预防数据冗余。它对由于网络、系统、Oracle Goldengate 进程需要重启而要丢失数据提供了容错。对于复杂的同步配置，checkpoints 能够使多个Extract 或者Replicat 进程读取同一个trail 集，Checkpoints 使用内部进程确认以防止消息在网络中丢失。

  Manager

  Manager 是用来控制Oracle Goldengate 的进程的。在Extract 和Replicat 能运行之前，Manager 在每台系统上必须被运行，并且在那些进程运行期间Manager 必须保持运行以执行管理功能。Manager 执行以下的功能：

  启动Oracle Golden Gate 进程，启动动态进程，维护进程端口，执行Trail 管理，创建event、error 和threshold reports，一个Manager 进程能控制多个Extarct 或者Replicat 进程。

  Collector

  当连续、在线的change synchronization 活动时，Collector 是一个运行在target 端的后台进程。Collector 做以下的事情：

- 提交一个来自远程Extract的连接请求给Manager，扫描和绑定一个可用的端口并且发送这个端口给Manager用来分配给请求的Extract进程。
- 接受来自Extract发送过来的提取了的数据库的改变，并且将他们写到一个trail文件中。当有一个网络连接请求时，Manager自动启动Collector，OracleGoldengate 用户不需要与它进行交互。Collector只能从一个Extract进程接收信息，所以你用的每一个Extract都有一个对应的Collector。

  默认情况下，是由source 端的Extract 初始化TCP/IP 连接到目标端的Collector，但是Oracle Goldengate 也能被配置成由target 端的Collector 初始化连接。由目标

端初始化连接可能是被要求的，比如目标端是在一个信任的网络区域内，而source 端是在一个不被信任的网络区域内。

3.5 容灾管理核心功能

3.5.1 设备资源

点对点容灾方案包括主备站点的两个数据中心，服务器、存储设备较多，整个系统地域跨度广。方案的管理使用统一的管理平台，支持跨站点管理整个生产和容灾系统，包括服务器、虚拟机、交换机和存储设备。

管理平台主要进行状态和性能的监控。所有管理员均可以登录统一的管理平台，针对权限范围内的设备进行管理，包括设备的注册、简单的故障告警、基本设备信息的查看以及权限范围内注册设备组网拓扑图的查看等。

对于容灾最关注的存储，可以直观查看存储RAID 和磁盘配置，LUN 归属等主要信息。

3.5.2 可视化拓扑

方案对应用保护拓扑、全局拓扑、容灾业务执行情况等提供可视化界面展示。

如图3-5 所示，客户可以从应用保护策略拓扑图中看到关键应用的数据文件对应的存储LUN，以及这些存储LUN 的容灾状态。基于应用的容灾视图有效展示了应用容灾状态，简化容灾配置，降低由于应用的部分关键文件没有保护导致灾备端业务无法拉起的风险。

图3-9 应用保护策略拓扑

客户可以从全局拓扑图中看到主备站点主机、存储的容灾业务状态，当容灾链路故障时，如下图所示，存储间的容灾箭头呈红色，提醒管理员系统需要修复。

4 容灾场景操作流程

关于本章

4.1 主备存储同构场景

4.1.1 系统搭建流程

通常建设容灾系统时，生产数据中心已经产生大量的生产数据，为了能尽快完成数据的异地保护，对于初始同步有以下方式：

对于阵列远程复制，如果距离较远且容灾链路带宽较少的情况下，推荐将灾备阵列置于生产数据中心完成初始同步，然后在灾备中心进行增量同步的方式。由于容灾链路带宽较小，如果采用容灾链路进行初始同步，需要的时间比较长（大部分场景超过一周），因此推荐将灾备阵列先运输到生产端，采用生产端机房GE网络或者FC网络进行初始同步。以典型配置图4-5组网为例，容灾系统搭建操作流程如下，整个操作不需要中断生产业务：

1. 将灾备存储运输到生产端，通过ISCSI 网络与生产存储连接。
2. 将需要保护的数据配置从生产存储到灾备存储的异步远程复制，选用手动启动模式，不需要配置复制周期，并启动初始同步。初始同步期间生产存储可以正常对外提供服务。
3. 由于初始同步需要较长时间，在初始同步完成后再进行一次增量同步，保证主备存储数据差异尽量小。
4. 同步完成后，将灾备存储运输到灾备端。灾备端部署灾备应用主机和容灾管理服务器。
5. 灾备存储接入容灾网络，通过BCManager配置保护组、保护策略和恢复计划，配置成功后保护策略会根据设置的时间表定期启动远程复制进行数据的增量同步。

6. 增量同步完成后，可以使用BCManager的一键式容灾测试功能，在灾备存储上自动打快照并拉起灾备应用，测试灾备数据的可用性，进而验证容灾系统搭建是否成功。

4.1.2 测试流程

容灾系统测试支持客户验证灾备端数据的可用性。本方案通过灾备端快照映射的方式实现容灾测试，测试过程中不影响生产业务，并能够一键式自动化完成。

基于应用的一键式容灾测试流程

异步复制场景和同步复制场景演练操作类似，基于Oracle 、SQL Server、DB2 等应用都支持一键式测试，需要如下操作：

1. 在BCManager中，选择远程恢复中已经建好恢复计划，点击“测试”按钮。

2. 观察容灾测试执行结果，确定测试成功，保证灾备数据库可以拉起。
3. 客户如果有额外的数据库测试需求，可以登录灾备服务器的数据库进行读写操作。注意，在容灾测试中进行的读写操作不能被保留，会在下一步清理操作后回滚。
4. 测试完成后进行清理操作，一键式清理演练环境。

在典型配置组网场景下，一键式容灾测试自动执行包括快照映射、启动数据库等所有测试操作，环境清理自动执行包括关闭数据库、删除快照等操作。

说明：

如果远程复制任务正在执行，容灾测试检查LUN 数据一致性会失败，因此建议在远程复制任务执行完成后再进行测试操作。

基于LUN 保护策略的容灾测试流程

一键式容灾操作支持Oracle、SQL Server、DB2 等应用，针对其他应用建议客户配置基于LUN 的保护策略，需要一些手动操作完成容灾测试，客户也可以针对保护策略开发自定义脚本实现一键式管理，自动实现应用拉起，具体容灾测试流程如下：

1. 等待远程复制的增量数据同步完成。
2. 在BCManager中，选择远程恢复中已经建好的恢复计划，点击“测试”。
3. 测试成功后，测试操作人员登录应用主机，扫盘，启动应用。如果已经开发和部署了自定义脚本，此时脚本能够自动进行操作，包括扫盘、启动应用、并测试应用一致性。

4. 检查应用拉起是否成功，客户可以根据自己需求对应用进行读写操作。
5. 测试完成后关闭应用。
6. 在BCManager中，选择远程恢复中已经建好的LUN对LUN恢复计划，点击“清理”。
7. 清理完成后，容灾测试整个流程完成。

注明：如果使用AIX 系统，在每次删除快照或LUN 映射后，都需要在AIX 系统上删除对应的无效盘符。

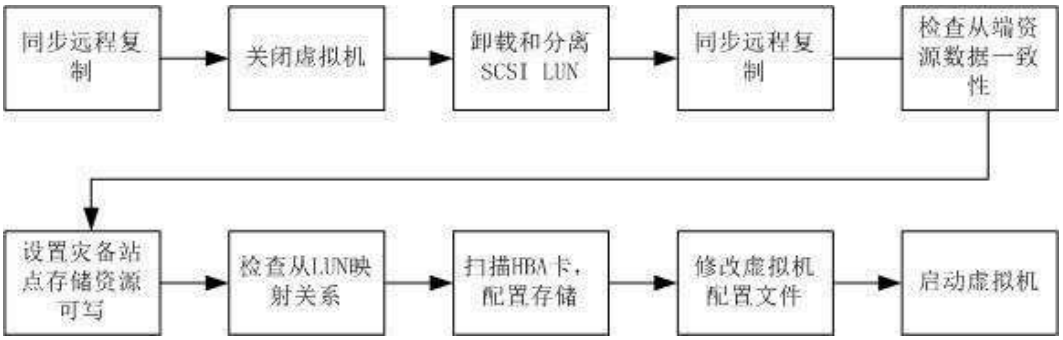
4.1.3 容灾演练流程

容灾演练是训练人员和提高灾难恢复能力而根据灾难恢复预案进行活动的过程。因此容灾演练需要模拟生产端故障发生后，在灾备端恢复业务的全流程。本方案通过计划性迁移的方式实现容灾演练。

本方案提供的演练流程可以在生产端正常停止业务后，一键式快速将业务迁移到灾备端运行。灾备端运行一段时间后，可以手动回切，灾备端的增量数据同步回生产端，生产端恢复业务。

基于应用的一键式演练流程

异步复制场景和同步复制场景演练操作类似，基于Oracle 、SQL Server、DB2 应用、VMware vSphere 虚拟机、FusionSphere 虚拟机等应用，支持通过一键式“计划性迁移”操作来实现演练，以VMware vSphere 虚拟机为例，整个计划性迁移的流程如图所示：

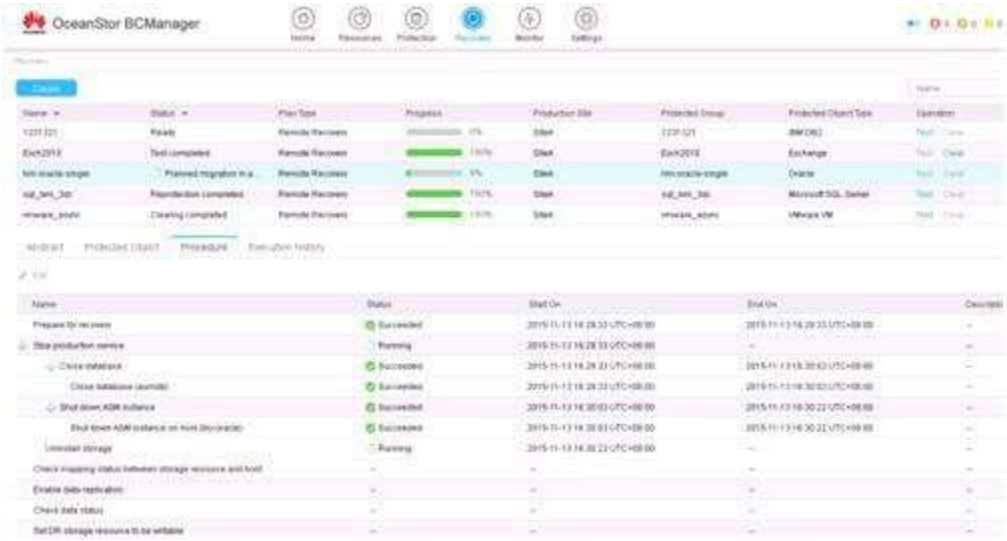


需要注意的是：容灾演练之前，必须先对该恢复计划执行至少一次测试，并完成清理。

BCManager 容灾演练操作步骤如下：

1. 在BCManager中，选中“恢复”中已经建好恢复计划，单击“计划性迁移”按钮。

2. 计划性迁移流程开始自动执行。



3. 计划性迁移流程执行成功后，客户可以在灾备端运行生产业务。

基于LUN保护策略的演练流程

一键式容灾操作只支持Oracle、SQL Server、DB2、Exchange、VMware vSphere 虚拟机和FusionSphere 等应用，针对不在BCManager 应用兼容性内的应用，客户可以通过配置基于LUN 的保护策略来实现管理，需要一些手动操作完成容灾演练，客户也可以针对保护策略开发自定义脚本实现一键式管理，自动实现应用拉起。

需要注意的是：容灾演练之前，必须先对该恢复计划执行至少一次测试，并完成清理。

具体容灾演练流程如下：

1. 停止生产应用，在ISM上解除生产主机到生产存储的映射。
2. 在BCManager中，选择远程恢复中已经建好恢复计划，单击“执行”，选择灾备中心即将要挂载的灾备主机。
3. 计划性迁移流程开始自动执行。

计划性迁移流程执行成功后，操作人员登录灾备主机，灾备主机扫盘，并启动灾备应用，在灾备端运行生产业务。如果已经开发和部署了自定义脚本，此时脚本能够自动进行扫盘、启动应用的操作。

4.1.4 灾难恢复流程

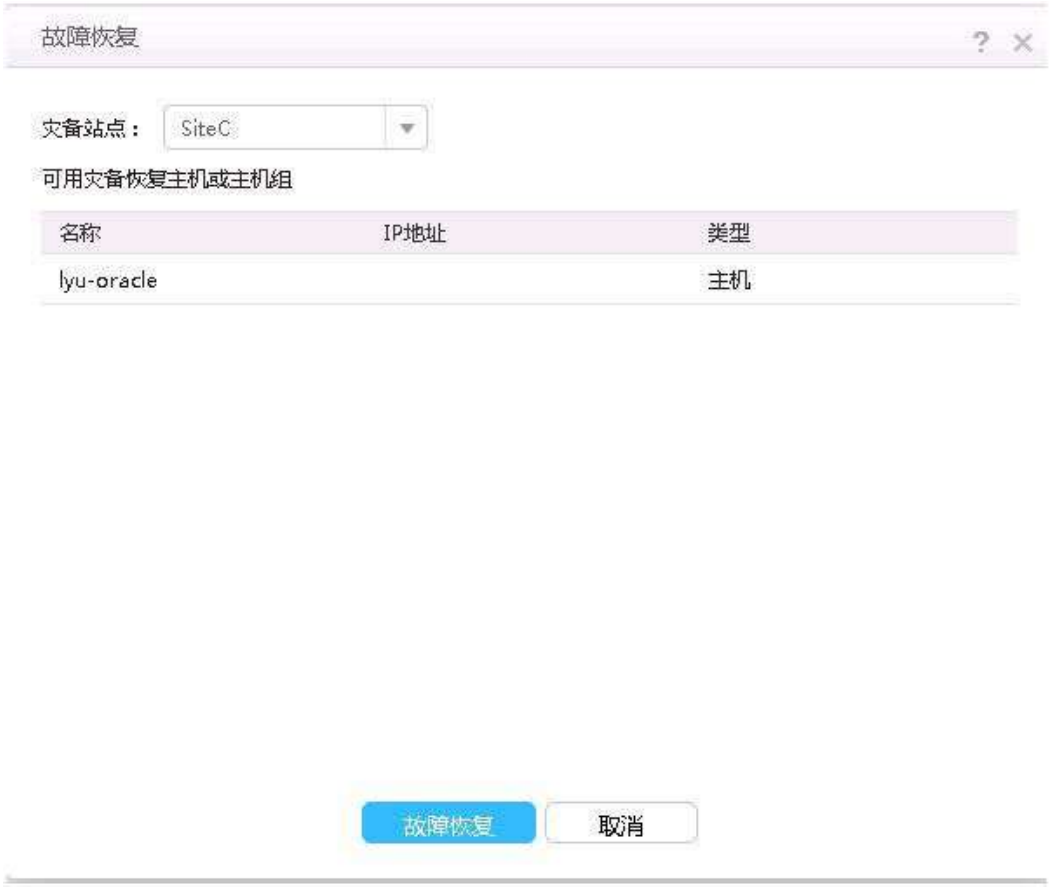
《信息系统灾难恢复规范》中定义，灾难恢复流程是指为了将信息系统从灾难造成的故障或瘫痪状态恢复到可正常运行状态、并将其支持的业务功能从灾难造成的不正常状态恢复到可接受状态，而设计的活动和流程。点对点方案的故障恢复流程是指当生产站点故障时，将生产业务切换至灾备站点运行的操作流程。

基于应用的一键式故障恢复流程

基于Oracle、SQL Server、DB2 等应用，支持一键式灾难恢复流程，操作如下：

1. 在BCManager中，选择远程恢复中已经建好恢复计划，单击“执行”。

2. 选择“故障恢复”恢复模式。



3. 故障恢复流程开始自动执行，等待执行成功。
4. 故障流程执行成功后，客户可以在灾备端运行生产业务。

说明：

灾备端主机的IP 与生产主机不同，容灾管理软件不支持网络或IP 切换操作。如果客户需要灾备主机IP 或者DNS 修改，需要客户手动修改。

基于LUN 保护策略的故障恢复流程

一键式容灾操作只支持Oracle、SQL Server、DB2 等应用，其他应用建议客户配置基于LUN 的保护策略，需要一些手动操作完成故障恢复，客户也可以针对保护策略开发自定义脚本实现一键式管理，自动实现应用拉起，具体故障恢复流程如下：

1. 在BCManager中，选择远程恢复中已经建好恢复计划，单击“执行”。
2. 选择“故障恢复”恢复模式。
3. 故障恢复流程开始自动执行，等待执行成功。
4. 故障流程执行成功后，操作人员登录灾备主机，灾备主机扫盘，并启动灾备应用，在灾备端运行生产业务。如果已经开发和部署了自定义脚本，此时脚本能够自动进行扫盘、启动应用的操作。

说明：

灾备端主机的IP 与生产主机不同，容灾管理软件不支持网络或IP 切换操作。如果客户需要灾备主机IP 或者DNS 修改，需要客户手动修改。

4.2 主备存储异构场景

4.2.1 系统搭建流程

使用OceanStor V3 进行异构虚拟化接管，并构建主备容灾环境的步骤如下：

步骤1 配置生产镜像存储并配置双交换机冗余组网。

步骤2 停止生产业务。

步骤3 解除友商阵列到生产主机的映射，将友商阵列上的LUN 映射到V3 阵列，然后从V3 阵列映射到生产主机。

步骤4 生产主机恢复业务。

步骤5 后续步骤与主备同构场景章节下系统搭建流程相同。

----结束

4.2.2 容灾操作流程

测试流程与4.1.2测试流程相同。

容灾演练流程与4.1.3容灾演练流程相同。

灾难恢复流程：

- 如果生产中心只是配置了异构虚拟化接管，发生灾难时切换至灾备中心继续运行，则灾难恢复流程与4.1.4灾难恢复流程相同。

- 如果生产中心配置的卷镜像，当异构存储阵列故障时，业务可以继续运行，无需灾难恢复，只需要将异构存储阵列修复即可。

4.3 数据库容灾场景

4.3.1 系统搭建流程

Data Guard

可以使用下面几种方法来创建Data Guard 物理备库：

- 使用RMAN“Activedatabaseduplicate”方式，支持11g 以上版本
- 使用RMAN传统方式备份，并在容灾端恢复
- 使用冷备的方式，并在容灾段恢复

本方案我们使用第一种方法，用“RMAN DUPLICATE FROM ACTIVE DATABASE”创建备库，在创建的过程中不需要关闭主库。以下是主要步骤：

1. 在主库上执行必要的更改：
   a. 启用force logging.
   b. 创建密码文件
   c. 创建备库redolog文件
   d. 根据dataguard需求修改主库参数文件，需要修改以下参数文件：LOG_ARCHIVE_CONFIG，LOG_ARCHIVE_DEST_1，LOG_ARCHIVE_DEST_2，LOG_ARCHIVE_DEST_STATE_1，FAL_SERVER，FAL_CLIENT，DB_FILE_NAME_CONVERT，LOG_FILE_NAME_CONVERT
2. 配置静态监听和tnsnames.ora，确保从备库连到主库的sql*net 网络连接工作正常
3. 通过生产端和容灾端之间的网络连接创建备库
   a. 创建备库密码文件
   b. 创建备库的初始化参数文件
   c. 为备库的数据文件创建合适的mountpoint和文件夹
   d. 在备库连到主库的前提下运行以下命令：

```
DUPLICATE TARGET DATABASE
FOR STANDBY
FROM ACTIVE DATABASE
SPFILE
PARAMETER_VALUE_CONVERT '', ''
SET DB_FILE_NAME_CONVERT '', ''
SET LOG_FILE_NAME_CONVERT '', ''
SET SGA_MAX_SIZE 200M
SET SGA_TARGET 125M;
```

4. 验证在备库日志文件传输情况和应用情况

5. 在主备站点都创建broker配置文件，设置参数dg_broker_config_file1，并启动broker：

   SQL> alter system set dg_broker_start=true scope=both;

   System altered.

6. 在第三站点配置一台oberver，安装oracle数据库客户端软件，配置到主备库的TNS 服务连接。

7. 在observer上启用快速启动故障切换（FSFO）：

   DGMGRL>edit configuration set protection mode as maxavilability;

   DGMGRL>edit database adg set property logxptmode=sync;

   DGMGRL>edit database adgdr set property logxptmode=sync;

   DGMGRL>edit database adg set propertyFastStartFailoverTarget=adgdr;

   DGMGRL>edit database adgdrsetproperty FastStartFailoverTarget=adg;

   DGMGRL>enable Fast-StartFailover

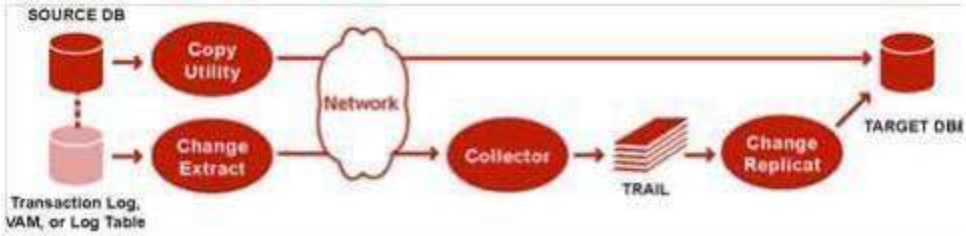
GoldenGate

创建备库可以从一个活动的source 数据库执行初始化加载。当初始化加载运行的时候，用户和应用依然能够访问和更新数据。如果需要延迟访问source 数据库，等到target 数据库表加载完成的话，你可以从一个静默的source 数据库执行初始化加载。

GoldenGate 支持数据加载的方式：

- 使用数据库工具去加载数据。这种方式是依赖与数据库工具的，比如Oracle的数据泵expdp、impdp。
- 从文件加载数据给Replicat。Extract写记录到一个Extract文件，Replicat应用它们到目标表。这种初始化方式是最慢的。
- 从文件加载数据到数据库工具。Extract用外部的ASCII格式写记录到extract文件。目标表通过批量加载工具从Extract文件中加载数据。Replicat运行这个加载过程并产生控制文件。
- 使用OracleGoldenGate 直接加载方式加载数据。Extract 与Replicat通过TCP/IP不使用Collector进程或文件直接通讯。Replicat通过数据库引擎应用数据。
- 使用一个直接批量加载到SQL*LOADER加载数据。Extract使用外部ASCII格式提取记录并且直接交付他们给Replicat，而Replicat 将这些记录发送给Oracle的SQL*Loader 批量加载工具。
- 使用Teradata加载工具加载数据。对于同步两个Teradata数据库这种方法是首选的。建议的工具是MultiLoad。

在本方案中，我们使用Oracle 数据泵进行初始化加载。用数据泵从源数据库将数据导出，并导入数据到目标数据库，在这个过程中需要启动一个change-synchronization Extract 进程组对变化的数据进行提取和记录。当拷贝复制完成，启动change-synchronization Replicat 进程组，把数据泵导出和导入过程中发生改变的数据进行重新同步。从那个点向前，Extract 和Replicat 进程组继续运行去维护数据的同步。这种方法不需要去调用任何特殊的初始化加载Extract 或者Replicat 进程。整个数据加载的过程如下图所示：



在数据初始化加载之后，在生产端和容灾段启动GoldenGate 进程。

生产端：

GGSCI (aprac1) 16> ADD EXTRACT a2pex TRANLOG, BEGIN 2014-12-08 18:00:01,threads 2

GGSCI (aprac1) 17> ADD EXTTRAIL ./dirdat/ap, EXTRACT a2pex, MEGABYTES 20

GGSCI (aprac1) 19> ADD EXTRACT a2pdp, EXTTRAILSOURCE ./dirdat/ap, BEGIN 2014-12-08 18:00:01

容灾端：

GGSCI (apdrora) 33> add replicat a2prp,exttrail ./dirdat/ap,checkpointtable soe.apcheckpoint

GGSCI (apdrora) 35> add checkpointtable soe.apcheckpoint

REPLICAT a2prp

USERID soe,PASSWORDsoe

ASSUMETARGETDEFS

REPERROR (DEFAULT, DISCARD)

DDLERROR DEFAULT DISCARD

DDLOPTIONS REPORT

DDL INCLUDE ALL

ALLOWNOOPUPDATES

MAP soe.*, TARGET soe.*;

4.3.2 测试流程

对于Data Guard 和GoldenGate，在配置好数据复制关系之后，在容灾端都可以进行只读查询。本方案在生产端插入数据，在容灾端验证数据复制成功。

在生产端使用脚本：

drop table soe.test;

CREATE TABLE soe.test ("test" VARCHAR2(50),constraint PK_test primary key ("test"));

commit;

declare

iint:=1;

begin

```
while i=1 loop
insert into soe.test values (to_char(sysdate, 'yyyy-mm-dd hh24:mi:ss'));
dbms_lock.sleep(1);
commit;
end loop;
end;
/
```

在容灾端确认相关表的记录和生产端一致：

```
select * from soe.test;
```

4.3.3 容灾演练流程

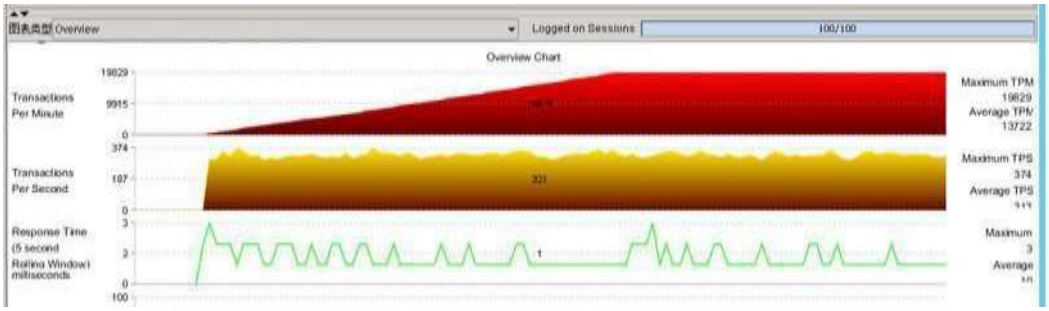
Data Guard

此场景我们使用Broker 来进行容灾演练，在Data Guard 命令行(DGMGRL)下对主备库进行切换，切换成功后进行业务验证，最后再进行回切完成容灾演练流程。

1. 切换前状态，adg是主库，adgdr是备库：

```
DGMGRL> show configuration

Configuration - adg

  Protection Mode: MaxAvailability
  Databases:
    adg   - Primary database
    adgdr - (*) Physical standby database

Fast-Start Failover: ENABLED

Configuration Status:
SUCCESS
```

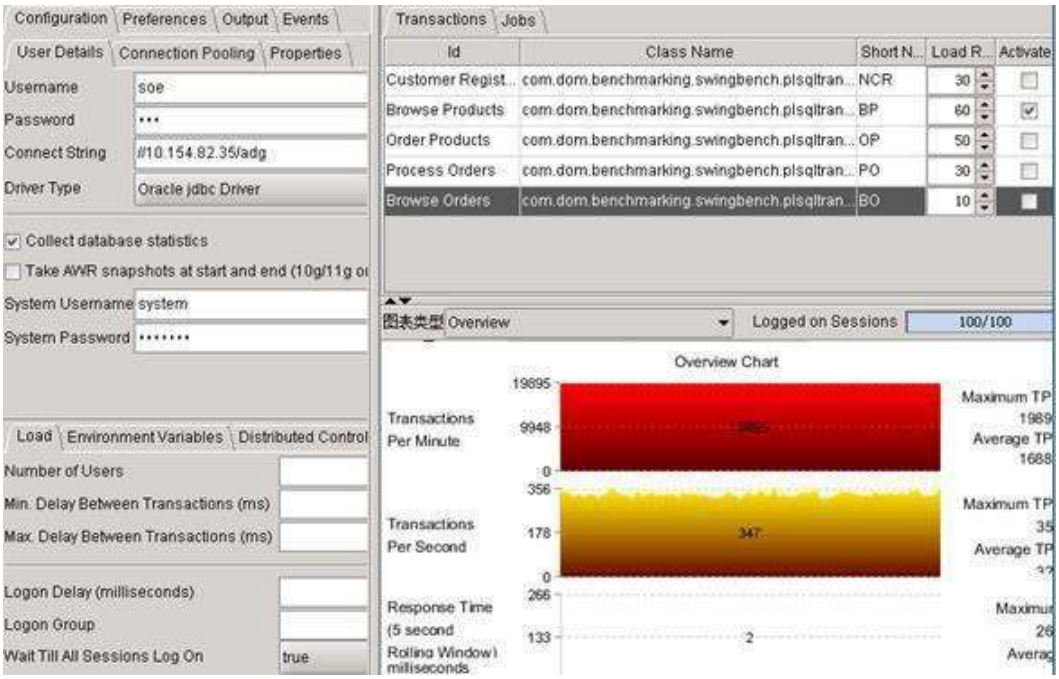
2. 使用模拟OLTP工作负载的工具Swingbench，在主库adg上连接100用户，运行OLTP 工作负载：



3. 使用DGMGRL通过“switchovertoadgdr”命令，切换到备库。切换成功后状态如下图所示：

```
DGMGRL> show configuration verbose

配置 - adg

  保护模式:          MaxAvailability
  数据库:
    adgdr - 主数据库
    adg   - (*) 物理备用数据库

  (*) 快速启动故障转移目标

快速启动故障转移: ENABLED

  阈值:              30 秒
  目标:              adg
  观察程序:          WIN-93OF7DIVATO
  滞后限制:          30 秒 (未使用)
  关闭主数据库:      TRUE
  自动恢复:          TRUE

配置状态:
SUCCESS
```

4. 切换后，adgdr变成新主库，Swingbench业务在adgdr正常运行：



5. 确认业务在灾备端运行正常后，使用dgmgrl将主库切回到adg，结束容灾演练：

```
[oracle@adgrac1 trace]$ dgmgrl
DGMGRL for Linux: Version 11.2.0.3.0 - 64bit Production

Copyright (c) 2000, 2009, Oracle. All rights reserved.

Welcome to DGMGRL, type "help" for information.
DGMGRL> connect sys/huawei123@adg
Connected.
DGMGRL> switchover to adg
Performing switchover NOW, please wait...
New primary database "adg" is opening...
Operation requires shutdown of instance "adg" on database "adgdr"
Shutting down instance "adg"...
ORACLE instance shut down.
Operation requires startup of instance "adg" on database "adgdr"
Starting instance "adg"...
ORACLE instance started.
Database mounted.
Database opened.
Switchover succeeded, new primary is "adg"
```

GoldenGate

针对的场景是数据库容灾演练，或者计划内的停机，更新升级软硬件。在发生数据库迁移，升级，系统维护时，为了最大限度的减少停机时间，需要通过切换数据库到容灾环境，原生产系统变为容灾系统，然后停机进行计划内的维护任务。由于备库在不断同步数据的同时已经为计划内的停机做好了故障切换的准备，通过live standby，进行主备库的切换，通过备库提供业务。在维护工作结束后，将新增加的业务数据同步到原生产数据库，最后回切到原生产数据库。

首先是切换到备库，以下是切换的详细步骤：

1. 首先停止一切在主库上的应用，但是让Extract和DataPump继续运行以捕获生产端未写到Trail文件的事务。在保证没有活动事务的情况下，才能切换干净。
2. 在生产端使用“LAGEXTRACTa2pex”命令了解Extract的延迟，若返回如"At EOF,nomorerecordstoprocess"的信息，则说明所有事务均已被捕获：GGSCI (aprac1) 144> lag a2pex Sending GETLAG request toEXTRACTa2pex ... Last record lag: 0 seconds. At EOF, no more records toprocess.
3. 在生产端执行“STOPEXTRACText_1”，停止Extract进程：GGSCI(aprac1) 146>stopa2pex Sending STOP request to EXTRACT a2pex...

   Request processed.
4. 在生产端重复执行“LAGEXTRACT a2pdp”命令，直到返回“AtEOF, nomore recordstoprocess”，表明Pump已经把所有捕获的数据都传送到容灾端的live standby：GGSCI (aprac1) 147> lag a2pdp Sending GETLAG request to EXTRACT a2pdp ... Last record lag: 3 seconds. At EOF, no more records toprocess.
5. 在容灾端执行“STOPEXTRACTa2pdp”，关闭Pump：GGSCI (aprac1)148> stop a2pdp Sending STOP request to EXTRACT a2pdp ...

   Request processed.
6. 使用“STATUSREPLICATa2prp”命令检查容灾端数据库replicat 的同步情况，如返回“AtEOF (end offile).”，则说明所有记录均被复制。
7. 在容灾端数据库执行“STOPREPLICATa2prp”，停止ReplicatGGSCI (apdrora) 72> stop a2prp Sending STOP request to REPLICAT a2prp ... Requestprocessed.
8. 紧接着我们可以在livestandby上为业务应用用户赋予必要的insert、update、delete 权限，启用各种触发器trigger 及cascade delete 约束等；以上手段在主库上对应的操作是收回应用业务的权限，disable 掉各种触发器及cascade delete 约束，

之所以这样做是为了保证在任何时候扮演备库角色的数据库均不应当接受任何除了GoldenGate 外的手动的或者应用驱动的业务数据变更，以保证主备库间的数据一致。然后运行相应脚本，切换应用服务器到容灾端，在容灾端启动应用程序，拷贝一些启动应用程序需要的，但是没有被GoldenGate 复制的文件到容灾端服务器。

9. 修改原备库上的extract的启动时间到现在，已保证它不去抽取那些之前的重做日志。并且同时启动备库上的extract，让它开始捕获新备库上的事务变化数据：GGSCI (apdrora) 75> alter p2aex ,begin now EXTRACTaltered.

   GGSCI (apdrora) 76> start p2aex

   Sending START request to MANAGER ...

   EXTRACT P2AEX starting

   注意：在原备库的extract 启动之后，不要打开原备库上的data pump 进程，也不要打开原主库上的replicat 进程，数据需要保存在原备库的trail 文件上，直到原主库完成停机维护工作，可以继续获取新主库的数据变化。

10. 把用户进程全部切换到新的主库，同时在原生产系统执行系统维护工作。

在原主库完成系统维护工作之后，我们可以将业务切换回主库，以下是详细的切换步骤：

1. 在原备库上，停止所有的应用程序，但是让extract进程继续工作，以保证所有未被捕获的事务都能被捕获
2. 在原主库，使用“STARTEXTRACTa2prp”命令启动replicat进程，准备接收新主库的日志变化。
3. 在原备库，使用“START EXTRACTp2adp”命令启动datapump进程，把本地trail 文件中的数据通过TCP/IP 传输到原主库。
4. 在原备库，使用“LAGEXTRACTp2aex”命令查看事务捕获情况，直到显示“At EOF, no morerecords to process”确保所有数据都已经被捕获。
5. 在原备库，使用“STOPEXTRACTp2aex”来停止extract进程
6. 在原备库上，使用“LAGEXTRACTp2adp”来查看pump进程是否有数据还在传输到原主库，如果输出显示“AtEOF, nomore recordstoprocess”，说明所有改变数据传输完成，使用“STOPEXTRACT p2adp”停止pump进程。
7. 在原主库，执行“STATUSREPLICAT a2prp”直到返回“AtEOF (endoffile)”，确认replicat 已经把trail文件里的数据应用到数据库。
8. 在原主库执行“STOPREPLICATa2prp”停止replicat 进程。
9. 完成在原主库上的维护工作：包括赋予权限，启用触发器，启动应用等。

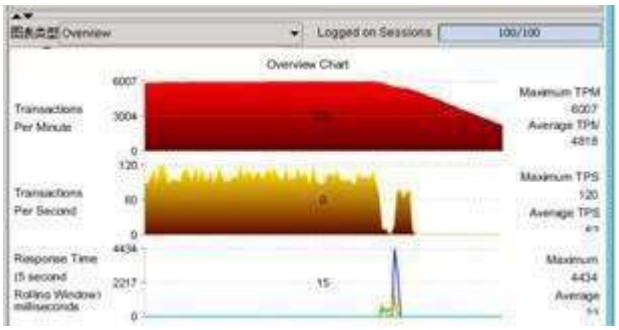
启动最早配置的由主库到备库的extract、pump、replicat，把客户端连接切换到原主库。

4.3.4 灾难恢复流程

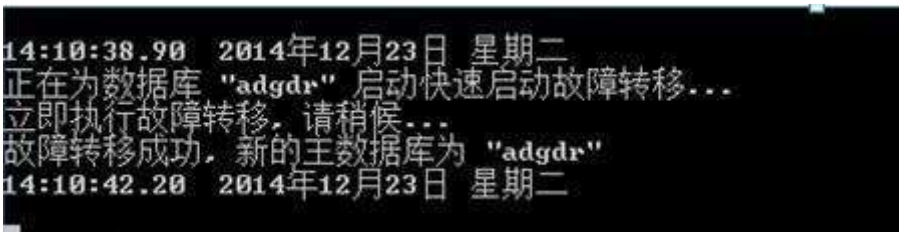
Data Guard

在此场景中，主库发生故障，业务中断。通过Oracle Data Guard 快速启动故障切换特性，自动将生产数据库切到灾备端。在原生产数据库修复后，再把主库切回到生产站点。

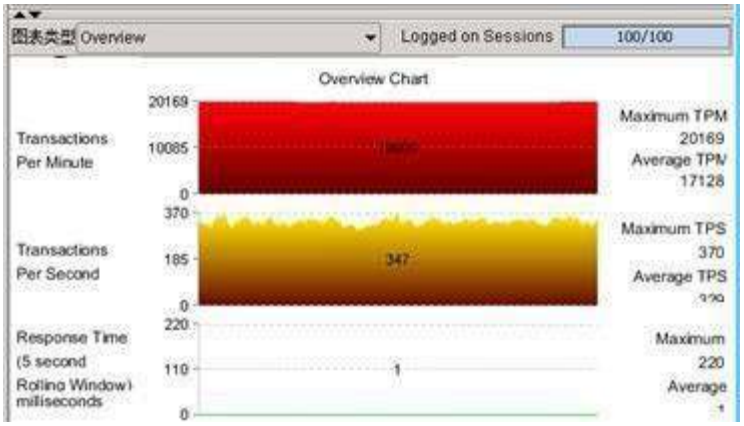
1. 在主库模拟故障，数据库崩溃，swingbench停止工作：



2. observer检测到主库故障，触发自动故障切换：



3. 切换成功后，容灾端成为新的主库，swingbench可以继续在灾备端运行：



4. 在生产端将原主库修复，并且执行startup，触发reinstate：

5. 此时原主库adg变成了新的备库：

```
DGMGRL> show configuration

配置 - adg

  保护模式:        MaxAvailability
  数据库:
    adgdr - 主数据库
    adg   - (*) 物理备用数据库

快速启动故障转移: ENABLED

配置状态:
SUCCESS
```

6. 最后执行回切，将主库切回生产站点，主备库关系回到最初状态：

```
DGMGRL> switchover to adg
Performing switchover NOW, please wait...
New primary database "adg" is opening...
Operation requires shutdown of instance "adg" on database "adgdr"
Shutting down instance "adg"...
ORACLE instance shut down.
Operation requires startup of instance "adg" on database "adgdr"
Starting instance "adg"...
ORACLE instance started.
Database mounted.
Database opened.
Switchover succeeded, new primary is "adg"
DGMGRL>
```

GoldenGate

除去计划内的容灾演练，实际生产中更多的故障切换发生在主机故障或主库不可用的情况下,这种情况下一般我们已经无法在Primary 上停止应用及extract 了；当我们在这样的情况下failover 到Standby 上后如同在DataGuard 环境下一样即便Primary 上的数据库恢复了我们也无法直接进行回切了，需要做的是重新配置Primary 上的 GoldenGate 并将Standby 上的数据以initial load 的形式还原回去，在数据重新同步后才能再切换到Primary 上。下面我们就来介绍如何在计划外的情况下从主库failover 到备库，并尝试回切。

1. 使用“lagreplicat”命令了解备库上的replicat的延迟情况，若返回"At EOF (end of file)"则说明replicat 已应用所有trail 中的数据到备库上。

   GGSCI (apdrora) 5> lag replicat a2prp

   Sending GETLAG request to REPLICAT a2prp ...

   Last record lag: 5 seconds.

   At EOF, no more records to process.

2. 停止备库上的replicat进程：

   GGSCI (apdrora) 6> stop replicat a2prp

   Sending STOP request to REPLICAT a2prp ...

   Request processed.

3. 在备库上执行必要的赋予DML权限，启动triggers触发器和cascadedelete约束的脚本。
4. 启动standby上的extract进程。在此之前先确认Standby上的datapump group不被启动，以保证trail文件堆积在standby上

GGSCI (apdrora) 16> start p2aex

Sending START request to MANAGER ...

EXTRACT p2aex starting

5. 此时可以将应用切换到备库上了，备库变成了新的主库。

以上步骤完成了故障切换到Standby 的过程，接下来我们尝试将应用还原到primary 上。

1. 如果原生产主机已损毁则需要重装Oracle软件，并重建生产系统上的Goldengate 软件目录
2. 从原主库启动GGSCI命令。
3. 删除原主库上相关的extract及EXTTRAIL,并重建

   GGSCI (aprac1) 6>delete extracta2pex

   Deleted EXTRACT a2pex.

   GGSCI (aprac1) 7> delete exttrail /d01/ext/c1

   GGSCI (aprac1) 14> add extract a2pex,tranlog,begin now

   EXTRACT added.

   GGSCI (aprac1) 15> add ./dirdat/ap, EXTRACT a2pex, MEGABYTES 20

   EXTTRAIL added.
4. 在primary 上启动Manager

   GGSCI (aprac1) 18> startManager

   Manager started.
5. 在原主库上执行disabletrigger触发器和cascadedelete约束的脚本
6. 在原备库上对执行热备份(逻辑，物理的均可);并记录该热备的结束时间
7. 使用原备库上的热备份来完成原主库上的initialload。
8. 在原主库，使用“START REPLICATa2prp”启动replicat
9. 在原备库，使用“STARTEXTRACTp2adp”启动pump开始传输改变数据到原主库。
10. 在原主库，执行“INFO REPLICATa2prp”直到确认所有在initialload过程中产生的数据变化都已经复制到原主库。
11. 禁用原主库上目前使用的HANDLECOLLISIONS 选项

    GGSCI (aprac1) 26> send replicat a2prp,NOHANDLECOLLISIONS
12. 关闭之前切换到原备库上的一切应用，关闭原备库上的extract、pump及原主库上的replicat。
13. 在原主库系统上赋予应用相关DML权限，启用触发器及删除约束
14. 修改原主库系统上的extract group的begintime 为当前，启动Primary 到Standby 的extract、pump 及replicat，此时系统切换回原始的主库到备库的复制。

5 结论

主备容灾解决方案是华为推出的针对各种行业，主要包括政府、医疗、金融、能源、大企业，提出的基于同城或者异地的主备模式容灾方案。结合强大的容灾管理功能和存储容灾技术，方案具备便捷运维管理、快速容灾切换和高效数据保护等亮点。

1. 便捷运维管理

   华为提供统一的运维管理平台，可远程登录进行统一管理，包括设备资源、容灾业务、故障告警等。管理软件具备图形化操作和容灾拓扑显示，使用户感知更加直观，运维更加方便。完善的登录和操作日志，为事后审计提供良好支撑。

2. 快速容灾切换

   支持一键式容灾测试、演练和故障切换，自动化脚本代替繁琐的手工操作，降低RTO。提供基于应用的数据一致性保护，能够感知生产主机的关键应用，在异步远程复制开启时，先进行主机内存数据刷盘，保证应用数据一致性，确保灾备端的业务能够拉起。

3. 全面高效的数据保护

   华为存储系统支持高中低端存储容灾业务直接互通，灾备端按需配置，支持配置低端阵列，降低灾备投资50%以上。支持异构虚拟化技术，可以轻松实现异构存储系统之间的容灾数据保护。

4. 本地数据高可用保护

   强大的卷镜像技术，配合异构虚拟化技术，为生产存储提供双保险，实现利旧与数据在设备间的高可用保护。