hp StorageWorks multi-site disaster tolerant solution

implementation blueprint

administration guide

Confidential

This document contains confidential and proprietary Information belonging to Hewlett-Packard, Inc. Access to this information is strictly limited to specifically authorized employees of Hewlett-Packard. All other persons are expressly forbidden to read or review this document without the prior written consent of Hewlett-Packard.

notice

Edition 0503

HP, StorageWorks, ServiceGuard, MetroCluster, Continental Clusters, Continuous Access, Continuous Access Extension, Business Copy, RAID Manager XP, LUN Configuration and Security Manager XP, LUN Configuration Manager XP, Secure Manager XP, and the HP logo are trademarks of Hewlett-Packard Company in the United States and other countries. UNIX is a trademark of The Open Group. All other product names mentioned herein may be trademarks of their respective companies.

Use of the terms "CA," "sync CA," "sync-CA," "async CA," or "async-CA" within this document refer to the HP StorageWorks Continuous Access or Continuous Access Extension products, and have no connection with the term "CA" copyrighted by Computer Associates.

Hewlett-Packard Company makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

format conventions

note This is a Note.

caution This is a Caution.

warning This is a Warning.

`User Input`	Specifies text to be typed exactly as shown, such as commands, path names, file names, and directory names.
`variable`	Indicates that you must supply a value.
`Screen text`	Denotes text displayed on the screen.

contents

1 Introduction

This *Multi-Site Disaster Tolerant Solution Implementation Blueprint: Administration Guide* explains how to perform the common tasks of maintaining, expanding, and administering the HP Multi-Site Disaster Tolerant (MSDT) solution.

audience

The audience for this guide includes the HP Account Support Engineer, the HP Customer Engineer, and the customer's IT personnel.

solution overview

In the MSDT Solution, two nearby sites—less than 100km apart—protect each other in case of an in-region disaster. Critical applications are mirrored synchronously, providing high availability for online transaction processing. If a failure occurs at one site, the other site can take over application processing, with minimal interruption, at virtually the exact point where it was interrupted. A third site (Site 3) located well outside of the region offers protection should both primary sites (Sites 1 and 2) be lost. A point-in-time copy is made at Site 2 and then mirrored to Site 3 at customer-defined intervals. Copying is a scheduled, scripted process enabled by the MSDT Management Tools. The data recovery point for Site 3 would be determined by the last time a copy cycle was completed.

The MSDT Solution solves the following business issues:

- Application continuance in case of a local or regional disaster while maximizing transaction-processing availability. Mirroring of Site 2 to Site 3 using a point-in-time copy allows placement of Site 3 out of the disaster region.
- High availability for 24 x 7 operations.
- ServiceGuard with MetroCluster and Continental Clusters provide a recovery time objective (RTO) of less than one hour. Synchronous capability between sites 1 and 2 allows for recovery point objective (RPO) of virtually zero.
- Customizable management scripts allow specific integration with customer environment and applications.

required skills and knowledge

The solution administrator must be familiar with:

- HP UX servers and HP StorageWorks XP disk arrays
- Device mapping and device access
- The HP StorageWorks software products used in the solution, including HP StorageWorks RAID Manager XP (RM), HP StorageWorks Business Copy (BC), HP StorageWorks Continuous Access (CA), and Continuous Access Extension (CA Ext)
- The HP cluster software products used in the solution, including HP ServiceGuard (SG), MetroCluster, Continental Clusters, and XP-CA Toolkit for MetroCluster
- The customer's MSDT solution design
- The customer's current network configuration
- The customer's disaster recovery business process and procedures

document assumptions

This guide assumes that the MSDT solution has been installed and tested.

The terms “host” and “node” are used interchangeably.

The terms “application” and “package” are used interchangeably.

In the MSDT Solution, there are four device groups:

device groups

CA_1	All devices in the synchronous replication between Site 1 and Site 2.
BC_1	All the devices used in the BC point-in-time copy created at Site 2.
CA_2	All the devices used in the long-distance replication copy from Site 2 to Site 3.
BC_2	All devices used in the creation of a point-in-time BC at Site 3.

If the customer has opted not to have a BC at Site 3, you will have only three device groups.

related documents

For more information on administering the MSDT Solution, see the following documents:

- *Multi-Site Disaster Tolerant Solution Business Blueprint*
- *Multi-Site Disaster Tolerant Solution Technical Blueprint*
- *Multi-Site Disaster Tolerant Solution Implementation Blueprint: Multi-Site DT Management Tools User Guide*
- *Designing Disaster Tolerant High Availability Clusters*: B7660-90013
- *Managing MC/ServiceGuard*: B3936-90065
- *HP ServiceGuard Quorum Server Version A.02.00 Release Notes*: B8467-90011
- *MetroCluster with Continuous Access XP Version A.04.20 Release Notes*: B8109-90013
- Product documentation for all HP products used in the solution

You can also request additional information at **`www.hp.com/go/storageblueprints.`**

2 Administrative Tasks

The MSDT Solution Administrator performs routine maintenance, system upgrades, and, in the event of a disaster, oversees disaster recovery operations. Each site has a primary administrator, identified during the solution design phase (Administrative Personnel Requirements Checklist), and should have at least one backup administrator who can handle disaster recovery tasks in the event that the primary administrator is unavailable. HP recommends that the customer select one system from which to perform all system administration, preferably at Site 3. Selecting a primary management system allows a single point of management and the creation a central daily event log for the MSDT topology.

finding the status of the MSDT solution

Administrators must often determine the status of an application and all the device groups within the configuration.

Because the solution includes a number of hosts, each with access to a different part of the solution, HP recommends that you use the MSDT Tools scripts to determine:

- The current status of each of the four device groups
- The current location of the application

The application's location indicates the point of data entry; from that point on there should be a logical data flow through the system.

The **`disp_conf`** script takes the application name as input, and uses information in the MSDT Tools configuration file to provide a visual output of the status of the application and all mirrored device groups.

sample output

The following is the output of a normal system with:

- **`msdt_type=4`** indicating the four links to manage
- **`cycle_type=0`** indicating that all devices must be suspended for the cycle process to be able to run

The application is running on Alpha109 at site “DC_one”; therefore, the P-vol is at Site 1. Because the BC_1 P-vol is at Site 2, the status of the CA_1 link is very important for data flow to Site 3.

```
      DC_one                           DC_two                          DC_three
====================             ====================             ====================
      alpha108                         alpha155                         alpha154
      alpha109                         alpha156
====================             ====================             ====================
Application fs3 status : Running
      alpha109
====================             ====================             ====================
                      CA_1
 PVOL_PAIR  - ------- PAIR  ------- > SVOL_PAIR
                     vg_fs3           PVOL_PSUS                            SVOL_PSUS

                                          |                                    |

                                          | BC_1                               | BC_2
                                        PSUS                                 PSUS
                                          | bc1_vg_fs3                         | bc2_vg_fs3

                                          |                                    |

                                      SVOL_PSUS            CA_2            PVOL_PSUS
                                      PVOL_PSUS    - - - - PSUS  - - - -   SVOL_PSUS
                                                         ca2_vg_fs3
```

In the sample output below, the CA_2 link has suspended with an error status (`PSUE`) because of a problem with the physical link between Sites 2 and 3.

The BC_2 device group is also in `PAIR` status, indicating that any changes on the CA volumes will be copied to the BC_2 devices.

Always suspend the BC_2 group before performing a **`resumepair`** on the CA_2 link.

Use the **`suspendpair -g bc2_vg_fs4`** command for this operation.

Once the physical link between Sites 2 and 3 is operational again, perform a **`resumepair -g CA2_vg_fs4`** to continue normal operations. The **`resumepair`** command starts the update copy process, and the device group will eventually reach `PAIR` status. To continue the **`cycle_pair`** operation, you must suspend this device group again because suspend is the expected status.

```
     DC_one                           DC_two                          DC_three
====================             ====================             ====================
      alpha108                         alpha155                        alpha154
      alpha109                         alpha156
====================             ====================             ====================
Application fs4 status : Running
      alpha108
====================             ====================             ====================
                      CA_1
 PVOL_PAIR  - ------- PAIR  ------- > SVOL_PAIR
                     vg_fs4           PVOL_PSUS                            SVOL_PAIR
                                                                               ^
                                          |                                    |
                                                                               |
                                          | BC_1                               | BC_2
                                        PSUS                                 PAIR
                                          | bc1_vg_fs4                         | bc2_vg_fs4
                                                                               |
                                          |                                    |
                                                                               |
                                      SVOL_PSUS            CA_2            PVOL_PAIR
                                      PVOL_PSUE    x x x x PSUE  x x x x   SVOL_PSUE
                                                         ca2_vg_fs4
```

The output below indicates that the CA_2 device group is in an error status that was not resolved with the **`disp_config`** script. The status reported by the device groups is inconsistent. This could happen if you run display group while a status change is in progress. Therefore, the P-vol reports the old status, and the S-vol reports the new status.

Run the **`disp_config`** script again. If this problem persists, use the RM command on the individual hosts to verify the status.

Check the RCU status of the links using CommandView XP.

```
      DC_one                           DC_two                          DC_three
====================             ====================             ====================
      alpha108                         alpha155                         alpha154
      alpha109                         alpha156
====================             ====================             ====================
Application fs5 status : Running
      alpha108
====================             ====================             ====================
                      CA_1
 PVOL_PAIR  - ------- PAIR  ------- > SVOL_PAIR
                     vg_fs5           PVOL_PSUS                            SVOL_PAIR
                                                                               ^
                                          |                                    |
                                                                               |
                                          | BC_1                               | BC_2
                                        PSUS                                 PAIR
                                          | bc1_vg_fs5                         | bc2_vg_fs5
                                                                               |
                                          |                                    |
                                                                               |
                                      SVOL_PSUS            CA_2            PVOL_PAIR
                                      PVOL_PSUE            ERROR           SVOL_PSUS
                                                         ca2_vg_fs5
```

The output below shows a configuration where the BC_1 device group is located at Site 1. In the output, CA_2 and BC_2 are in `PAIR` status. Having CA_2 and BC_2 in `PAIR` status at the same time is not desirable because any changes on the CA_2 link are immediately copied to the BC_2 group, which nullifies the protection offered by BC_2 copy.

The **`cycle_pair`** script will not function in this status because **`cycle_pair`** always expects the CA_2 and BC_2 groups to be in `SUSPEND` status.

Use the **`suspendpair`** command to suspend device groups.

```
      DC_one                           DC_two                          DC_three
====================             ====================             ====================
      alpha155                         alpha108                         alpha154
      alpha156                         alpha109
====================             ====================             ====================
Application fs6 status : Running
      alpha155
====================             ====================             ====================
                      CA_1
 PVOL_PAIR  - ------- PAIR  ------- > SVOL_PAIR
 PVOL_PSUS           vg_fs6                                               SVOL_PAIR
                                                                              ^
     |                                                                        |
                                                                              |
     | BC_1                                                                   | BC_2
   PSUS                                                                     PAIR
     | bc1_vg_fs6                                                             | bc2_vg_fs6
                                                                              |
     |                                                                        |
                                                                              |
 SVOL_PSUS                              CA_2                              PVOL_PAIR
 PVOL_PAIR  - ------------------------ PAIR   ------------------------ > SVOL_PAIR

                                      ca2_vg_fs6
```

The output below shows the application running at Site 2 after a manual or automated application transfer. The CA_1 group status is `PAIR`, indicating that communication to Site 1 is operational, and Sync CA is protecting the data. In this configuration, the CA_1 P-vol is at the same site as the BC_1 P-vol. Therefore, the **`cycle_pair`** does not specifically check the CA_1 link status.

If **`cycle_type=0`**, the configuration is ready for the next cycle process (all devices are suspended). If **`cycle_type=1`**, the BC_1 device must be in `PAIR` status.

note Use the **`resumepair`** command to resume BC_1 copy operation. Wait until the device groups are in `PAIR` status before you run the **`cycle_pair`** script.

```
       DC_one                           DC_two                          DC_three

====================             ====================             ====================

      alpha108                         alpha155                         alpha154

      alpha109                         alpha156

====================             ====================             ====================

Application fs7 status : Running

                                       alpha156

====================             ====================             ====================

                      CA_1

 SVOL_PAIR  < ------- PAIR  ------- - PVOL_PAIR

                     vg_fs7           PVOL_PSUS                           SVOL_PSUS


                                          |                                    |


                                          | BC_1                               | BC_2

                                        PSUS                                 PSUS

                                          | bc1_vg_fs7                         | bc2_vg_fs7


                                          |                                    |


                                      SVOL_PSUS            CA_2               PVOL_PSUS

                                      PVOL_PSUS    - - - - PSUS  - - - -      SVOL_PSUS

                                                         ca2_vg_fs7
```

The output below indicates that the BC_1 device group is in COPY status. This could mean:

- A cycle process is running. The process is waiting for BC_1 to reach PAIR status to continue the cycle operations.
- Someone has recently resumed the mirror processes. The **cycle_pair** process will not start until BC_1 reaches PAIR status.

```
       DC_one                           DC_two                          DC_three
====================             ====================             ====================
      alpha108                         alpha155                        alpha154
      alpha109                         alpha156
====================             ====================             ====================
Application fs3 status : Running
      alpha109
====================             ====================             ====================
                      CA_1
 PVOL_PAIR  - ------- PAIR  ------- > SVOL_PAIR
                     vg_fs3           PVOL_PSUS                            SVOL_PSUS

                                          |                                     |

                                          | BC_1                                | BC_2
                                        PSUS                                 PSUS
                                          | bc1_vg_fs3                          | bc2_vg_fs3

                                          |                                     |

                                      SVOL_PSUS            CA_2            PVOL_PSUS
                                      PVOL_PSUS    - - - - PSUS  - - - -   SVOL_PSUS
                                                         ca2_vg_fs3
```

cycling data

Cycling data is the process of creating a new point-in-time copy on the BC_1 devices and then moving (or cycling) this data to Site 3. This process must run at regular intervals or at specific times during a day. After the initial testing of the **`cycle_pair`** process and any modifications specific to the installation, this process should be performed from a scheduler process. If you do not use a central scheduling tool, you can use the standard UNIX “cron” process to start the process at regular intervals.

The default **`cycle_pair`** script has two options for cycling the data:

1. All devices start in a `SUSPEND` status (except for CA_1 which is always paired). The first action resumes the BC_1 device group. The time it takes for BC_1 to reach `PAIR` status depends on the amount of new data and can be different every time you run the scripts. This option creates the point-in-time copy at an undetermined time after the scripts start. This is not always the best method.
2. Assume that the BC_1 device is in `PAIR` status when the **`cycle_pair`** process begins. This creates the point-in-time copy immediately after the **`cycle_pair`** process starts, giving more consistent timing to the **`cycle_pair`** process.

The **`cycle_pair`** process requires devices to be in a specific status before starting the cycle process. If the devices are not in the proper status, the **`cycle_pair`** process aborts with an error. This error is reported to you via email or another messaging system, and requires immediate attention. Subsequent **`cycle_pair`** operations will continue to fail until you resolve the error condition.

Under normal conditions, the **`cycle_pair`** operation starts and ends with the same status, allowing the next cycle to continue without error. An error during a **`cycle_pair`** process, or human interaction with the device groups during normal operation, might leave the devices in an undesirable status. HP recommends that you monitor the cycle process closely and check the **`cycle_pair`** log file daily to ensure continuous operation of this process.

For more information on cycling data and the **`cycle_pair`** script, see the *MSDT Implementation Blueprint: Multi-Site DT Management Tools User Guide*.

testing metrocluster failovers – between sites 1 & 2

Testing cluster failover options not only enables you to verify the configuration of the systems, but also ensures that if a failure occurs, the failover process will work without any surprises. It is easy to make small changes on one system and forget to make the changes on other systems. During a failure, these small changes can prevent the application from starting correctly, and can result in extended downtime before the problem is corrected.

Schedule downtime to move the application from one system to the next. It is best to move the application to all hosts in the configuration (one at a time) to ensure successful startup and operation on each host. You can move the application to the next host, and then on the next downtime event, move the application back to the previous host. This way, you can check normal operations on all applicable hosts. In addition, testing in this way eliminates any issues regarding the application's ability to run permanently on these hosts. There are two levels of testing that you can perform:

- Testing the application only
- Testing hardware failures

level 1: testing the application only

This test only checks the application to ensure that it can start on the next node in the cluster. It does not test for specific failures in the environment. You can perform this test at regular intervals; it has a very low risk factor. The main goals of this test are to ensure that configuration information is available on all the nodes, and that this information is up to date.

caution	Unless the application is going to remain active on this node, prevent user access to the application while testing to ensure that no unintentional data changes occur.

Follow these steps to perform the MetroCluster failover test for the application:

1. Use the **`cmhaltpkg`** *`package`* command to stop the application on one system.
2. Use the **`cmrunpkg -n`** *`node package`* command to start it on the next system.

   Because there are no node or disk failures initiating the movement of the package, there is very low risk of accidentally causing data inconsistencies. The **`cmhaltpkg`** command gracefully stops the application and unmounts all volume groups for the current system.

   If the command cannot halt the package in an ordinary fashion (due to continuous user access or incorrect shutdown options), investigate the application halt command in the package configuration file (the **`customer_defined_halt_cmds`** section of the **`/etc/cmcluster/`***`package_name`***`/package_name.cntl`** file). Apply any necessary changes to ensure a successful complete halt of the application every time the command is executed.

The **`cmrunpkg`** command performs the following actions:

- Checks to ensure the package is enabled to run on this node.
- Checks to ensure the package is not running on any other node in this cluster or in the Continental Cluster.
- Checks the CA replication status and, if necessary, swaps the personalities of the CA group. Because the array and communication between the arrays are not disabled, the personality swap should work without any errors.
- Activates the volume groups.
- Mounts the file systems.
- Starts the application.

level 2: testing hardware failures

In this test, you must disable or intentionally fail a hardware element to ensure that the cluster software detects the failure and initiates the appropriate movement of the application.

caution When intentionally failing hardware in this manner, you run the risk of data corruption or permanent loss of hardware components. HP recommends that you do not run these tests on a regular basis or without assistance from an HP support team.

host failure only

One of the most common failures at a site is a host system failure. To simulate a host failure:

1. Turn off the power supply to the host system. Pulling the plug from the socket is not recommended.

note A graceful shutdown will not cause the failure you are trying to test.

   The cluster software should detect a heartbeat timeout and reform the cluster with the remaining nodes. After cluster reformation, the package automatically starts on the next available node in the cluster. During package startup, the cluster software evaluates the CA device group status and initiates a personality swap if needed. Because the array and the communication between the arrays are enabled, the personality swap should be successful and the application should start normally. Once the failed node reaches operational status, the node should automatically join the cluster, but will be disabled from running the package.

2. Restore the failed node by restoring the power supply to the host.
3. Check the initial hardware test output for any errors. After initial self-test and OS startup, the node should automatically join the cluster.
4. During the failure process, the cluster software disables this host from running the application, because the host had a hardware failure. After any power-on tests are performed, use the **`cmmodpkg -n`** *`node_name`* **`-e`** *`package_name`* command to re-enable this node to run the application.
5. Use the **`cmviewcl -v`** command to ensure that the node is ready to run the package.

host and array failures

caution Do not initiate a power failure on the XP disk array. A graceful power shutdown can take several hours to complete, and affects other systems sharing the same array. HP recommends that you simulate both array and CA failures by disconnecting the appropriate interface cables.

When the cluster software performs the movement of the package, any attempt to swap the personalities of the CA devices will fail. This leaves the S-vol in `SSWS` status. The parameters in the MetroCluster XP-CA Toolkit configuration file (**`/etc/cmcluster/`***`package_name`***`/`***`package_name`***`_xpca.env`**) determine if the application is allowed to start. Because the CA link between the arrays is down, no synchronous replication can take place, leaving the customer data unprotected against failures. In this case, preventing the application from starting prevents further data loss (in the case of a rolling disaster), but also leaves the application unavailable.

XP-CA toolkit for metrocluster configuration parameters

parameter		description
AUTO_FENCEDATA_SPLIT		This parameter applies only when the fence level is set to DATA. This causes the application to fail if the CA link fails or if the remote site fails.
	Value 0	This value *does not* start the package at the target site. If you set this value, the application will not start until you either fix the hardware problem or force the package to start by creating the **`FORCEFLAG`** file. Use this value to ensure that the S-vol data is always current with the tradeoff of long application downtime while the CA link and/or the remote site are being repaired.
	Value 1 (DEFAULT)	This value starts the package at the target site, and requests the local disk array to automatically split itself from the remote array. This ensures that the application will start at the target site without having to fix the hardware problems immediately. Note that the new data written on the P-vol will not be remotely protected, and the data on S-vol will not be current. When the CA link and the remote site are repaired, use the **`pairresync`** command to rejoin the P-vol and S-vol. Until that command successfully completes, the P-vol will not be remotely protected, and the S-vol data will not be consistent. Use this value to minimize the downtime of the application. The tradeoff is manually resynchronizing the groups while the application is running at the primary site.
AUTO_PSUSSSWS		This parameter applies when the P-vol is in the suspended state (`PSUS`), and S-vol is in the failover state (`PSUS(SSWS)`). When the P-vol and S-vol are in these states, it is hard to tell which side has the latest data. When starting the package in this state on the P-vol side, you run the risk of losing any changed data in the P-vol.
	Value 0 (DEFAULT)	This value *does not* start the package at the primary site. You must choose which side has the latest data, and resynchronize the P-vol and S-vol. You can also force the package to start by creating the **`FORCEFLAG`** file.
	Value 1	This value starts the package after resynchronizing the data from the S-vol side to the P-vol side. This option is risky because the S-vol data may not be acceptable.

note If you disable the CA link during a failover test, you must manually complete the swap command and ensure that data replication continues.

Failback to the P-vol side of this device group is disabled until you perform the **`pairresync`**, **`-swapp`**, or **`–swaps`** command (**`resumepair`**, **`-swapp`**, or **`-swaps`** for MSDT Tools) on either the P-vol or S-vol side of the device group, and return the group to `PAIR` status with the P-vol and S-vol at the correct site. When the device group is in `PAIR` status, you can move the application back to the original node or leave it on the current node.

testing the data copy at site 3

Testing the integrity of data at Site 3 is a bit more complex than the site 1 to 2 MetroCluster failover test (see "testing MetroCluster failovers" on page 15).

Stopping the MetroCluster package (application) and starting the package at Site 3 causes the Continental Cluster scripts to initiate a takeover on the CA_2 link, which fails because it cannot create two S-vols on one device. The device at Site 3 becomes S-vol `SSWS` indicating that it is write enabled and can be used to run the package. Although this is standard procedure during a package failover to Site 3, recovering from this configuration is complex and requires full initial copies over the long distance link.

Because the recovery from a failover to Site 3 is complex and could take a long time (depending on the CA link performance), HP recommends that you regularly test only the integrity of the data on the remote side, but not the failover process. You can test the data copy at Site 3 without initiating a failover to the site.

note Testing the data integrity and configuration at Site 3 requires you to stop the cycle process for the time necessary to test the solution. Any changes made to the data at Site 3 will be lost after the test when the next **`cycle_pair`** process starts.

caution Until step 4 of this process, you must prevent all user access to the application during this test.

The process is as follows:

1. To ensure that Site 3 has the latest data, complete a **`cycle_pair`** process. This protects against failures during the test process.

note If a disaster occurs, you might not have time to complete a **`cycle_pair`** process. For a more accurate test, skip step 1and use whatever data is currently available at Site 3.

2. Use the Tools **`disp_conf`** script to verify that the BC_2 device groups are suspended (default status at the end of the **`cycle_pair`** process).

   This ensures that an untouched copy of the production data remains at Site 3 in the event of a disaster.

3. Suspend the CA_2 link and read/write enable the S-vol device using the **`suspendpair -g`** *`ca_2_device_pair`* **`-rw`** command.

   This enables the hosts at Site 3 to access the data.

4. The XP-CA Toolkit scripts prevent the cluster from running on this device because it is still an S-vol and it is not in `SSWS` status. Disable the toolkit by renaming the configuration file (**`/etc/cmcluster/`***`package_name`***`/`***`package_name`***`_xpca.env`**).

   If this file is not available, the cluster will not check any device group settings and will start the application on one of the local devices.

5. Disable the Continental Cluster. The application will only start if it is not running on any other node in the Continental Cluster.

6. Start the ServiceGuard package on the local system.

7. Check the package startup file for any startup errors.

8. Because the original production data devices belong to a different cluster, you might need to change the cluster ID for each volume group that is part of the package. To change the cluster ID:

   – Remove the current cluster ID using the **`vgchange -c n`** command.

   – Write the new cluster ID to the devices using the **`vgchange -c y`** command.

9. Check the integrity of the database by allowing selective users to access the database and perform queries.
10. Stop the application and enable the Continental Cluster.
11. Restore the original name of the XP-CA Toolkit for MetroCluster configuration file.
12. Resume mirror operations using the **`resumepair -g ca_2_device_group`** command.

note All changes on the S-vol of the CA_2 group will be overwritten by the data from the P-vol side.

13. Resume the cycle operations.

modifying and distributing the msdt tools configuration file

You can modify the MSDT Tools configuration file can from any host that has the MSDT Tools software installed.

1. Create a copy of the existing configuration file and make your changes to this copy.

caution Do not directly edit the existing configuration file. Always make a copy to avoid overwriting critical configuration information.

2. Use the **`msdtverify`** *`filename`* command to check your syntax.

   If the verification fails, check the log file for specifics of the failure and correct the errors.

When you complete the verification, execute the **`dist_conf`** *`filename`* command to distribute the new configuration file to all defined hosts.

The **`dist_conf`** *`filename`* command:

1. Validates the data in the configuration file
2. Copies the entire file to all the hosts defined in the configuration file

This command helps you keep the configuration information consistent between all hosts.

This command outputs a list of hosts that have either successfully received the new configuration file or failed to receive the configuration file. Check any failed hosts manually to ensure that the MSDT Tools software is running and available, and all network communications to these hosts are functional. After manually checking the host, execute the **`dist_conf`** command again.

For details, see the *MSDT Implementation Blueprint: Multi-Site DT Management Tools User Guide*.

adding a device control host

A device control host manages one or more of the device groups in the configuration. When you expand the system or when a device control hosts fails, you must add a new one. A device control host requires:

- A physical connection to one or more of the arrays in the configuration
- A command device from that array

1. If the host has access to the physical devices in the device group (is an application host), implement command device security.

   – or –

   If the host does not have access to the physical devices (only manages the device group status information), disable the command device security.

note Every host should have its own command device; two hosts should not share the same command device.

2. Create the RM instances on the new host. For details, see the RM manual.
3. Start the RM instance(s). For details, see the RM manual.
4. To check the configuration file, run a **`pairdisplay -l`** for the device group.

caution Take great care to identify all the devices correctly and in the same order as the other RM instances managing the same device group. Verify your configuration file and compare it to the other RM configuration files.

5. Once the instance is defined correctly, add the host and instance to the MSDT Tools configuration file.

note If this is a new host in the configuration, you must install the MSDT Tools software on this host. For more information on the Tools, see the *MSDT Implementation Blueprint: Multi-Site DT Management Tools User Guide*.

To test the MSDT Tools commands from a remote system, use the **`-h`** option with the command to specify the new host. For more information on command options, see the *MSDT Implementation Blueprint: Multi-Site DT Management Tools User Guide*.

maintaining the cluster

Maintaining the cluster environment is an important administration task. Check the cluster log files and the application status to ensure the correct response during a cluster failover.

The sample below shows the output of the **`cmviewcl`** command. The output shows the current status of all servers and packages in the MetroCluster configuration. The output shows package *`fs2`* as currently being down. Packages *`fs4`* and *`fs9`* are running on, but the **`AUTO_RUN`** parameter is `disabled`, indicating that the application will not automatically switch if a failure occurs.

```
# cmviewcl

CLUSTER      STATUS
multi_site_1 up

  NODE         STATUS       STATE
  alpha108     up           running

    PACKAGE      STATUS       STATE        AUTO_RUN     NOD
    fs1          up           running      enabled      alpha10
    fs7          up           running      enabled      alpha10
    fs8          up           running      enabled      alpha10
    ccmonpkg_1   up           running      enabled      alpha10

  NODE         STATUS       STATE
  alpha109     up           running

    PACKAGE      STATUS       STATE        AUTO_RUN     NOD

    fs4          up           running      disabled     alpha10
    fs9          up           running      disabled     alpha10

  NODE         STATUS       STATE
  alpha155     up           running

    PACKAGE    STATUS       STATE          AUTO_RUN     NOD
    fs3        up           running        enabled      alpha10

  NODE         STATUS       STATE
  alpha156     up           running

    PACKAGE    STATUS       STATE          AUTO_RUN     NOD
    fs8        up           running        enabled      alpha10

UNOWNED_PACKAGES

    PACKAGE      STATUS       STATE        AUTO_RUN     NOD
    fs2          down         halted       disabled     unowne
```

Using the **-v** option with the **cmviewcl** command provides more critical information. In the example output below, the quorum server (*alpha154*) is down. In the event of a site failure, this could be critical because the failure might result in a loss of 50 percent or more of the active nodes, resulting in a failure of the entire cluster. The example output also shows that package *fs1* is disabled from switching (AUTO_RUN=*disabled*), and host *alpha155* cannot run this package because of a previous failure on this node. Some of these errors might be legitimate problems in the configuration that you are working on, and might also prevent a successful failover in the event of a disaster.

```
# cmviewcl -v

CLUSTER      STATUS
multi_site_1 up

  NODE         STATUS       STATE
  alpha108     up           running

    Quorum_Server_Status:
    NAME                STATUS       STATU
    alpha154            down         unknow

    Network_Parameters:
    INTERFACE    STATUS       PATH         NAM
    PRIMARY      up           10/4/4.1     lan
    PRIMARY      up           10/4/8.1     lan
    PRIMARY      up           10/12/6      lan

    PACKAGE      STATUS       STATE        AUTO_RUN     NOD
    fs1          up           running      disabled     alpha10

      Policy_Parameters:
      POLICY_NAME     CONFIGURED_VALUE
      Failover        configured_node
      Failback        manual

      Node_Switching_Parameters
      NODE_TYPE    STATUS       SWITCHING    NAM
      Primary      up           enabled      alpha108     (current
      Alternate    up           enabled      alpha10
      Alternate    up           disabled     alpha15
      Alternate    up           enabled      alpha15
```

You can run the **cmviewcl -v** from any command. For more information, see the ServiceGuard manual.

Use the **cmviewcl -v** to check the status of all nodes, resources, and applications in the cluster, and to ensure that all hosts are able to run the appropriate applications. Use this command to ensure that automated failover is enabled for all applications.

You must distribute any change to cluster configuration files, application control files, MetroCluster configuration files and file system configurations to all nodes in the cluster. For details, see "modifying and distributing the msdt tools configuration file" on page 20.

maintaining the volume group (lvm)

In a cluster environment, it is very important to keep all volume group configurations consistent on all nodes. Keeping configuration information consistent ensures that all devices are available on all systems. It also minimizes application startup failure if a disaster occurs. If you do not keep the configuration information consistent, the required information may not be available in the event of a failure, and you will have to perform a manual recovery procedure. For details, see the MetroCluster manual.

monitoring the network

There are three network environments to consider in this solution. The first two are low impact to the solution but very specific to the application and cluster usage. Make sure the networks remain operational and are not over-utilized.

user access network

This network enables the application user to access application information. It is part of the shared public network, and provides access to a number of users over a wide geographical area. Redundancy on this network ensures uninterrupted access to the application. Network maintenance is outside the solution scope, but it may have an impact on the method of re-routing users after a failover. Because MetroCluster uses the same subnet across the two sites and utilizes the floating IP address, the application IP remains the same regardless of where the application is running. You can access the application as long as the external network infrastructure is functional.

heartbeat network

This network is for private heartbeat communication between nodes in the cluster. The heartbeat network is vital to the cluster's health. Dedicated heartbeat networks prevent timeouts caused by large transfers from another application.

note You can use the user access network as a standby heartbeat network.

private network

The third, and most important, network for the MSDT Solution is the private network for CA over IP implementation between Site 2 and Site 3. This network should be dedicated to CA traffic only, and should have specific guaranteed service level agreements (SLA). The latency, bandwidth and packet loss characteristics of this network determine the performance of the Asynchronous CA copy and ultimately impacts the time to cycle the data.

note You must ensure that the networks are operational, and sized correctly. Also make sure to maintain the private network's available bandwidth as application demands grow.

adding storage to the solution

As the customer's business expands, an administrator must regularly expand the application.

note In a multi-site configuration, you must provide a new device(s) in each of the four device groups over the three arrays, and extend each of the device groups.

1. Plan the extension — determine how many devices the expansion requires, and what size they must be.

note Do this on the production system first to ensure that you can extend the application to the desired size.

2. Identify the new devices added to the production system.

   Use host commands like **`ioscan`** or the **`xpinfo`** tools to identify the new devices. This information identifies the CU:Ldev numbers of the newly added devices. After the devices are identified, add them to the application.

3. Identify secondary devices for the CA_1 link on the array at Site 2.

   You must specify the CU:Ldev numbers of the newly added devices, as well as the device's array port and port assignments.

   Take note of the CU number of these devices, and ensure that there is a RCU link configured for them. If these devices have a new CU without a pre configured RCU to the remote system, then you must create a new RCU between the arrays.

4. Update the RM configuration files for all hosts that manage the CA_1 link.

caution Be sure to specify the correct port target LUN numbers for each device, and maintain consistency between all configuration files on the host. .

5. Restart the RM instance.
6. Use the **`paircreate`** command to create the CA_1 link.
7. Identify the secondary devices for the BC_1 device group on the system that manages the device group at Site 2.

   You must specify the CU:Ldev numbers of the newly added devices, as well as the device's array port and port assignments.

8. Add the new devices to all RM configuration files on all hosts that manage the BC_1 device group.

caution Be sure to specify the correct port target LUN numbers for each device, and maintain consistency between all configuration files on the host.

9. Restart the RM instance.
10. Use the **`paircreate`** command to create the BC_1 group.

note When adding only single devices to an existing group, you must run the **`paircreate`** command for each device you add to the system.

11. Identify the target devices for the CA_2 device group on the system at Site 3. You must supply information that identifies the CU:Ldev numbers of the newly added devices, as well as the array port and port assignments used for these devices..

    Take note of the CU number of these devices to ensure that there is a RCU link available for them.

12. Update the RM configuration files on all hosts that manage the CA_2 device group.

caution Be sure to specify the correct port target LUN numbers for each device, and maintain consistency between all configuration files on the host. .

13. Use the **`paircreate`** command to create the CA_2 link
14. Identify the target devices for the BC_2 device group on the system at Site 3.

    You must specify the CU:Ldev numbers of the newly added devices, as well as the device's array port and port assignments.

15. Add the new devices to all RM configuration files on all hosts that manage the BC_2 device group.

caution Be sure to specify the correct port target LUN numbers for each device, and maintain consistency between all configuration files on the host. .

16. Restart the RM instance.
17. Use the **`paircreate`** command to create the BC_2 group.

note Because RM only reports the status of the devices defined within the configuration file of the instance used to create the device group, you must ensure that all instance configuration files define the correct number of devices and the correct logical devices.

18. Use the **`vgexport`** and **`vgimport`** commands to distribute to all nodes in the cluster any changes to the underlying file system and logical volumes on the hosts.

note The cycle process functions normally even if the RM instance does not address the correct devices.

adding and removing users

The MSDT Solution does not require any specific user configurations, but the application that the solution uses in the solution may require an administrator on each system.

Ensure that the required users exist on every system and have the same user ID, password, and permissions on each of the systems in the cluster. Enter new or updated user information on every system that runs the application.

upgrading software

All systems in the solution must run the same version of the application software.

You can use a rolling method to upgrade the software, where a host that is not currently running the application is updated first, the application data is moved to this host, and the new version of software is tested.

If satisfied with the new version, update all other systems using the same process. After updating the software on all hosts, perform failover and data integrity tests on all systems to ensure that failover is still working. For more information about rolling upgrades, consult the ServiceGuard administration manual.

upgrading hardware

Hardware upgrades are possible at any time on a standby system.

For upgrades on an active system:

1. Migrate the application to a standby system
2. Upgrade the primary system

caution When upgrading links between sites, ensure that the failover link is functioning. Perform upgrades in a scheduled downtime to avoid accidental impact to applications.

3 Disaster Recovery Tasks

failing over to site 2

If the host at Site 1 fails, but the storage system is still operational, the application automatically migrates to the host at Site 2, and continues to operate as normal. This automatic failover is part of the MetroCluster software function, which moves the application within the cluster from one host to another.

If the hosts and storage system at Site 1 fail, the application migrates to Site 2. However, because the CA_1 link has failed, the application fails to start unless you force it to start on unprotected devices. You can do this in two ways:

- Use the **AUTO_FENCEDATA_SPLIT** parameter in the configuration file. When the fence level is set to DATA, the application fails if the CA link fails or if the remote site fails.
- Use a "force flag" file. The forceflag starts the package on the local devices and overwrites the standard functionality of the MetroCluster software. Enable the forceflag by creating a file named **FORCEFLAG** in the application directory (**/etc/cmcluster/***application_name***/FORCEFLAG**).

AUTO_FENCEDATA_SPLIT parameter

value	description
0	This value *does not* start the package at the target site. If you set this value, the application will not start until you either fix the hardware problem or force the package to start by creating the **FORCEFLAG** file. Use this value to ensure that the S-vol data is always current, with the tradeoff of long application downtime while the CA link and/or the remote site are being repaired.
1 (default)	This value starts the package at the target site, and requests the local disk array to automatically split itself from the remote array. This ensures that the application will start at the target site without having to fix the hardware problems immediately. Note that the new data written on the P-vol will not be remotely protected, and the data on S-vol will not be current. When the CA link and the remote site are repaired, use the **pairresync** command to rejoin the P-vol and S-vol. Until that command successfully completes, the P-vol will not be remotely protected, and the S-vol data will not be consistent. Use this value to minimize the downtime of the application. The tradeoff is manually resynchronizing the groups while the application is running at the primary site.

The cycling of data from Site 2 to 3 can continue as normal because the BC_1 copy is located at Site 2. If the Site 1 failure is permanent, change the configuration to a two-site async CA solution (see "reconfiguring the solution after failure" on page 42).

failing over to site 3

When both Site 1 and Site 2 are disabled by disaster, you must start the application at Site 3, and then manually initiate the recovery processes.

Factors that influence the decision to failover to Site 3 include:

- The type of disaster at Site 1 and Site 2
- The length of the expected outage
- Status of the data at Site 3

To determine the status of the data at Site 3, investigate:

- The current status of the local devices
- Last entries in the **`cycle_pair`** log file

The **`cycle_pair`** process distributes log events to all hosts in the solution, so all the hosts at Site 3 have a **`cycle_pair`** log file that contains the latest status information.

If the failure occurs between two **`cycle_pair`** events, the data on the CA_2 S-vol device is valid, but one cycle old. If the failure occurs during a cycle process, and the data was being copied from Site 2 to Site 3, then the data on the CA_2 device is inconsistent and not usable. In this case, restore data from the BC_2 devices; this data is at least 2 cycles old.

Investigate the communication links between Site 2 and 3. This link may still be operational. If so, attempt to complete a new cycle process, and move the latest data from Site 2 to Site 3. In this case, there would be no data loss when starting the application at Site 3.

When you start the application at Site 3, you only need to issue one command (**`cmrecovercl`**) to start the application.

caution	Before issuing the startup command, finish all attempts to copy the latest data from Site 2 to 3, and finish the data restore process.

When the system is ready to be started, issue the **`cmrecovercl`** command to start all recovery packages at Site 3. During the startup process, the MetroCluster software (CA integration scripts) attempts a takeover on the CA_2 link. This takeover fails because Site 2 is down, but it leaves the S-vol in `SSWS` status and enables read/write access. Alternatively, you can use the **`cmrunpkg`** command to start a specific package.

The cluster software should automatically change the cluster ID on the devices. If the cluster software fails to change the cluster IDs, you can change them manually using the **`vgchange -c n`** and **`vgchange -c y`** commands.

Once the application is running, you might need to reconfigure some network equipment (DNS and routers) to allow users to access the application again on a new subnet.

preventing the application from starting incorrectly

The MetroCluster software, designed to support a two-site configuration, only manages one device group and the status of that device group (typically the CA_1 device group in this solution).

- For the MetroCluster configuration, the cluster software evaluates the status of the CA_1 device group and, if the status is valid and the data is usable, starts the application. If the data is inconsistent and not usable, the software does not start the application.
- For the Continental Cluster configuration, the cluster software evaluates the status of the CA_2 device group and, if the status is valid and the data is usable, starts the application. If the data is inconsistent and not usable, the software does not start the application.

The MetroCluster/Continental Cluster combination implemented in this solution ensures that the application is only running at one site at any time and only in one cluster at any time. However, the cluster software does not prevent the application from restarting at Site 1 or 2 once the application has started at Site 3 and begun updating information. After you disable the automatic startup at Sites 1 and 2, the Continental Cluster software ensures that the application cannot start on those sites while the application is running at Site 3.

note Starting the application at Site 3 requires a manual decision and a push-button startup process. Usually, you would start the application at Site 3 because Sites 1 and 2 are unavailable. If those sites are down, you cannot change their cluster/package configuration files and prevent the application from automatically starting at one of those sites.

If Sites 1 and 2 have recovered from the failure, then you must prevent all hosts in this cluster from running the application. Do this one of two ways:

- Use the **`cmmodpkg`** command to disable all hosts running the application. This tells the cluster that this host cannot run the package. This setting is only maintained while the cluster is running. If the cluster must be restarted, this setting resets to the default configuration set in the package configuration file.

  – or –

- Change the automated startup parameter in the cluster package configuration file so that the application cannot start automatically. Set **`AUTO_RUN = NO`** in the cluster package configuration file (**`/etc/cmcluster/`***`package_name`***`/`***`package_name`***`.ascii`**). Run the **`cmapplyconf -k -P`** *`package_name`***`.ascii`** command to apply this change to the cluster package configuration file.

To determine if it is safe to start the application at Site 1 or 2, run the **`disp_conf`** script to display the status of the device groups in the configuration. If the data at Site 3 resides on an S-vol in `SSWS` status or a P-vol in the CA_2 device group, then it is not safe to start the application at Site 1 or 2 without the risk of inconsistent data. The **`disp_conf`** script outputs a warning message if this risk occurs.

initiating failback

When the failed site is restored, the system administrator initiates failback to return the solution to its original configuration.

recovering from metrocluster failure

Recovery from a failure in the MetroCluster configuration (Sites 1 and 2) is relatively easy. During the failure the application's automated startup attempts to swap the local device group device functions from Secondary volume (S-vol) to Primary Volume (P-vol).

- If the link to P-vol arrays is still operational, the swap succeeds: the S-vol becomes the P-vol and the old P-vol becomes an S-vol.

  — or —

- If the link to P-vol arrays is not operational, the swap fails: the S-vol remains in `SSWS` status, enabling read/write access to this device.

note With the correct settings (**`AUTO_NONCURDATA=1`**) in the MetroCluster configuration file, the application can run, but cannot return to the primary side of the array.

To get the application back to the original node, the device group swap process must succeed.

1. Recover the physical links between the arrays and verify that they are operational. This can be performed from the Command View array management system.
2. Complete a device group swap using one of these commands (both commands perform the same action):
   - The **`resumepair -swapp`** command from the P-vol side of the array

     — or —

   - The **`resumepair -swaps`** command from the S-vol side of the array

   The S-vol (in `SSWS` status) becomes a P-vol and the old P-vol (in `PSUE` status) becomes an S-vol. At this stage the primary volume for the device group is located on Site 2 (failover site) and the original swap process (performed during the failover of the application) is completed.
3. The array will perform a delta resync process to copy all changes on the new P-vol to the new S-vol.

   Upon completion, the device group should be in `PAIR` status.
4. Halt the application on the current site using the **`cmhaltpkg`** command.
5. Start the application on the primary site using the **`cmrunpkg`** command.

   During the startup process the MetroCluster scripts perform another swap: the primary volume (P-vol) for the device group will now be located on Site 1 and the secondary volume (S-vol) will be located on Site 2.

note Because the physical links are operational, this swap should work without errors.

recovering from continental cluster failure

During a high impact disaster that disables both Sites 1 and 2, the application failover to Site 3. Once you recover Site 1 and 2 and verify that they are operational, you must move the data from Site 3 to Sites 1 and 2. This long and complex operation requires deleting and re-creating some device groups.

[DW – there are pieces missing from this process. They are broken up in chunks and I think it's confusing. It would be better if we could just present a process from beginning to end without all the explanation of why we're doing such and such, or at least move the explanation to notes...]

```
Printing configuration diagram:

            DC_one                           DC_two                          DC_three
     ====================             ====================             ==================
           alpha108                         alpha155                         alpha154
           alpha109                         alpha156
     ====================             ====================             ==================
     "fs9" status : Running
                                                                             alpha154
     ====================             ====================             ==================
                           CA_1
      PVOL_PAIR  - ------- PAIR  ------- > SVOL_PAIR
                          vg_fs9           PVOL_PAIR                            SVOL_PSUS
                                               |
                                               |                                  |
                                               |
                                               | BC_1                               | BC_2
                                             PAIR                               PSUS
                                               | bc1_vg_fs9                         | bc2_vg_fs9
                                               |
                                               |                                  |
                                               V
                                           SVOL_PAIR            CA_2           PVOL_PSUS
                                           PVOL_PSUS    x x x x SSWS  x x x x   SVOL_SSWS
                                                              ca2_vg_fs9

 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
 ***WARNING*** A takeover on CA_2 has failed and left devices in SSWS status
               This is S-vol suspend with swap pending status
               (Both P-vol and S-vol is read/write enabled)
               This indicate that application fs9 was started on DC_three

 ****WARNING****
 DO NOT START APPLICATION IN ANY OTHER DATA CENTER BEFORE RECOVERY FROM THIS STATUS IS COMPLETED

 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

Application is writing to this device

step 1 — swap the CA_2 device group

The application startup command on Site 3 attempts to swap the device groups on CA_2. However, the BC_1 device group's S-vol is the same device as the new S-vol for CA_2. A physical device cannot be the secondary volume for two replication groups at the same time. Therefore, this swap fails. The local device remains in S-vol `SSWS` status, and the application can run at this site but with no mirror protection.

To enable the swap of the CA_2 device group:

1. Delete the BC_1 device group using the **`deletepair -g BC_1_device_group -S`** command.

   The BC_1 group is deleted and will be in simplex (SMPL) status.

2. Complete a device group swap for CA_2 using:

   – The **`resumepair -swapp`** command from the P-vol side of the array

   — or —

   – The **`resumepair -swaps`** command from the S-vol side of the array

   The device at Site 3 becomes the P-vol of the CA_2 device group and the device at Site 2 becomes the S-vol of the group. All data on the P-vol at Site 3 is copied to Site 2.

note You can perform this process with the application still running at Site 3.

```
Printing configuration diagram:

            DC_one                           DC_two                           DC_three
     ====================             ====================             ===================
           alpha108                         alpha155                          alpha154
           alpha109                         alpha156
     ====================             ====================             ===================
     "fs9" status : Running
                                                                              alpha154
     ====================             ====================             ===================
                           CA_1
      PVOL_PAIR  - ------- PAIR  ------- > SVOL_PAIR
                          vg_fs9             SMPL                               SVOL_PSUS

                                                                                   |

                                                 BC_1                                | BC_2
                                             SMPL                                 PSUS
                                                 bc1_vg_fs9                          | bc2_vg_fs9

                                                                                   |

                                             SMPL                CA_2          PVOL_PSUS
                                           SVOL_COPY   < <<<<<<< COPY  <<<<<<<   PVOL_COPY
                                                               ca2_vg_fs9
```

step 2 — re-create the BC_1 device group

Re-Creating the BC_1 device group at this stage will result in a failure to perform the paircreate command. It is not possible to create a mirror group if the secondary device is already part of another mirror group. A create action performs a full initial copy operation (in no specific IO order) and invalidates all tracks of the target device, and therefore leaving the target devices and there associated target devices invalid for a extended periods of time. The array prevent this situation by not allowing the create process to succeed. . Therefore, you cannot re-create the BC_1 device group while the CA_1 device group exists.

1. Use the **`deletepair -g ca_1_device_group -S`** command to delete the CA_1 device group.

Use the **`createpair -g bc_1_device_group -h hostname -I inst -vl -m grp`** command to create the BC_1 device group, taking care to specify the correct direction for creating the device group. The instances managing a BC device group can exist on the same host, so you must specify both the hostname and instance number. The **`-vl`** (vector local) parameter indicates that the host and instance specified will be the P-vol of the device group.

note You can create the BC_1 group while the CA_2 group is still in `COPY` status.

```
Printing configuration diagram:

            DC_one                           DC_two                           DC_three
     ====================             ====================             ===================
           alpha108                         alpha155                          alpha154
           alpha109                         alpha156
     ====================             ====================             ===================
     "fs9" status : Running
                                                                              alpha154
     ====================             ====================             ===================
                           CA_1
        SMPL               SMPL              SMPL
                          vg_fs9           SVOL_COPY                            SVOL_PSUS
                                                ^
                                                ^                                  |
                                                ^
                                                ^ BC_1                               | BC_2
                                             COPY                                 PSUS
                                                ^ bc1_vg_fs9                         | bc2_vg_fs9
                                                ^
                                                ^                                  |

                                           PVOL_COPY             CA_2          PVOL_PSUS
                                           SVOL_PAIR  < ------- PAIR  ------- - PVOL_PAIR
                                                               ca2_vg_fs9
```

step 3 — re-create the CA1 device group

Because of different copy technologies between BC and CA functions, a BC device group and CA device group cannot both be in `PAIR` status at the same time. One must be in `SUSPEND` status if the other is in `PAIR` status. Therefore, you can only create the CA_1 device group after suspending the BC_1 device group.

1. Suspend the BC_1 group using the **`suspendpair -g bc_1_device_group`** command.

   The BC_1 device group goes into `SUSPEND` status (`PSUS`).

2. Re-create the CA_1 device group using the **`paircreate`** command.

note Specifying the correct host name with optional instance number ensures that the group is created in the correct direction.

```
Printing configuration diagram:

            DC_one                           DC_two                           DC_three
     ====================             ====================             ===================
           alpha108                         alpha155                          alpha154
           alpha109                         alpha156
     ====================             ====================             ===================
     "fs9" status : Running
                                                                              alpha154
     ====================             ====================             ===================
                           CA_1
      SVOL_COPY  < <<<<<<< COPY  <<<<<<<   PVOL_COPY
                          vg_fs9           SVOL_PSUS                            SVOL_PSUS

                                                |                                  |

                                                | BC_1                               | BC_2
                                             PSUS                                 PSUS
                                                | bc1_vg_fs9                         | bc2_vg_fs9

                                                |                                  |

                                           PVOL_PSUS             CA_2          PVOL_PSUS
                                           SVOL_PAIR  < ------- PAIR  ------- - PVOL_PAIR
                                                               ca2_vg_fs9
```

Application is still running here

step 4 — stop application and obtain data consistency throughout the configuration.

Until now, the application has been running at Site 3, adding new data to the CA_2 device group. Because the BC_1 device group is in `SUSPEND` status, the latest data has not reached the CA_1 device group.

1. Suspend the CA_1 device group using the **`suspendpair –g CA_1_device_group`** command. This is required to be able to resume the BC_1 device group.
2. Resume copy operations on the BC_1 device group using the **`resumepair`** command.
3. Wait until the BC_1 device group reaches `PAIR` status
4. Stop the application at Site 3 using the **`cmhaltpkg`** *`package_name`* command.

   No new data can enter the system.

```
Printing configuration diagram:

            DC_one                            DC_two                           DC_three
     ====================              ====================             ===================
           alpha108                          alpha155                          alpha154
           alpha109                          alpha156
     ====================              ====================             ===================
     The application fs9 was not found running on any Hosts
     ====================              ====================             ===================
                           CA_1
      SVOL_PSUS    - - - - PSUS  - - - -    PVOL_PSUS
                          vg_fs9            SVOL_PAIR                            SVOL_PSUS
                                                ^
                                                |                                   |
                                                |
                                                | BC_1                                | BC_2
                                              PAIR                               PSUS
                                                | bc1_vg_fs9                          | bc2_vg_fs9
                                                |
                                                |                                   |
                                                |
                                            PVOL_PAIR            CA_2           PVOL_PSUS
                                            SVOL_PAIR  < ------- PAIR  ------- - PVOL_PAIR
                                                               ca2_vg_fs9
```

5. Ensure that all data in the sidefile at Site 3 has reached the remote side by Suspending the CA_2 device group using the **`suspendpair –g CA_2_device_group`** **command**.

   All the latest data is now available for the BC_1 device group at Site 2.
6. Suspend the BC_1 device using the **`suspendpair –g bc_1_device_group`** command.

   The BC_1 S-vol now has the latest data. This is the same device as the CA_1 P-vol on Site 2.
7. Resume copy operations on the CA_1 device group using the **`resumepair`** command.

   This will copy all the latest data to Site1. Application data is now consistent throughout the configuration and the application can be started.

```
Printing configuration diagram:

            DC_one                           DC_two                            DC_three
     ====================             ====================              ====================
           alpha108                         alpha155                          alpha154
           alpha109                         alpha156
     ====================             ====================              ====================
     "fs9" status : Running
                                            alpha155
     ====================             ====================              ====================
                           CA_1
      SVOL_PSUS <  ------- PAIR  ------- -    PVOL_PSUS
                          vg_fs9           PVOL_COPY                               SMPL

                                               V
                                               V
                                               V BC_1                                  BC_2
                                             COPY                                  SMPL
                                               V bc1_vg_fs9                            bc2_vg_fs9
                                               V
                                               V
                                               V
                                           SVOL_COPY             CA_2              SMPL
                                             SMPL                SMPL              SMPL
                                                               ca2_vg_fs9
```

step 5 — change the cluster id

During the failover process, the cluster at Site 3 wrote a new cluster ID to the CA_2 devices, this ID is now copied to the all the Sites, including the cluster on Site 1 & 2. Because MetroCluster was designed to work in a single cluster solution, it does not change cluster IDs. Therefore you must manually change the cluster ID on the system on which you want to start the application.

1. Log on to the system.
2. Look in the package configuration file to find the volume group names that are part of the package.
3. For each of the volume groups in the package, remove the volume group from the cluster using the **`vgchange -c n`** command.
4. For each of the volume groups in the package, write the new cluster ID to the volume group using the **`vgchange -c y`** command.

step 6 — start the application in MetroCluster (Site 1 or 2).

1. Start the application on either site 1 or 2 using the **`cmrunpkg`** command.

   If application is started on Site 2, (the current owner of the CA_1 P-vol device) no device functionality swapping will be required.

   If the application is started on Site 1, (the current owner of the CA_1 S-vol) device functionality swapping will be required.

note This operation should successfully complete because there are no restrictions preventing the device at Site 2 from becoming an S-vol.

At this point, the data is again protected by sync CA, but the cycling process cannot resume since BC_1, CA_2 and BC_2 device groups are in the incorrect direction.

step 7 — re-create the device groups to site 3

To get the cycle process functional again, you must re-create the device groups to Site 3 (in the reverse direction).

1. Delete the BC_1, CA_2, and BC_2 device groups using the **`deletepair`** command.
2. Create the BC_1 device group using the **`paircreate`** command.

caution Make sure that you specify the direction for **`paircreate`** correctly!

Wait for the copy process to complete, and the BC_1 device group to reach `PAIR` status.

3. Suspend the BC_1 device group using the **`suspendpair`** command.
4. Create the CA_2 device group using the **`createpair`** command .

   Wait for the copy process to complete, and the CA_2 device group reaches `PAIR` status.

note This may take some time dependant on the performance of the WAN link between Site 2 and 3.

5. Suspend the CA_2 device group using **`suspendpair`** command.
6. Create the BC_2 device group using the **`createpair`** command.

   Wait for the copy process to complete, and the BC_2 device group reaches `PAIR` status.
7. Suspend the BC_2 device group using the **`suspendpair`** command.

step 8 — resume the cycle process

The configuration is now ready to resume the cycle process. Depending on the cycle type, you might need to resume copying to the BC_1 device group in order to have the BC_1 in `PAIR` status at the beginning of the cycle process.

```
Printing configuration diagram:

            DC_one                           DC_two                          DC_three
     ====================             ====================             ====================
           alpha108                         alpha155                         alpha154
           alpha109                         alpha156
     ====================             ====================             ====================
     "fs9" status : Running
           alpha109
     ====================             ====================             ====================
                           CA_1
      PVOL_PAIR  - ------- PAIR  ------- > SVOL_PAIR
                          vg_fs9           PVOL_PSUS                            SVOL_PSUS

                                               |                                    |

                                               | BC_1                               | BC_2
                                             PSUS                                 PSUS
                                               | bc1_vg_fs9                         | bc2_vg_fs9

                                               |                                    |

                                           SVOL_PSUS            CA_2            PVOL_PSUS
                                           PVOL_PSUS    - - - - PSUS  - - - -   SVOL_PSUS
                                                              ca2_vg_fs9
```

reconfiguring the solution after failure

A failover from Site 1 to Site 2 is completely automated—the administrator does not need to be present at the time of system failure. However, if it becomes necessary to fail over to Site 3, the system administrator needs to initiate the failover manually.

The replication manager, XP-CA Toolkit for MetroCluster, changes the direction of replication to ensure that data replication continues after a host failure. The replication manager toolkit ensures that only consistent, valid data is used, and prevents the cluster application from accessing inconsistent data. The replication manager does not allow the cluster software to access the devices if the replication status is not correct, or if data integrity or consistency is not guaranteed.

site 1 inoperable — converting to a two-node async solution

If a local disaster disables Site 1, the application automatically moves to Site 2. Cycling of the data can continue, but real-time data protection is disabled because of the failure at Site 1 and the synchronous mirroring to Site 1.

It is possible to continue in this mode with a point-in-time available copy at Site 3, up to two timeframes back. If the failure at Site 1 is going to be long term and you cannot identify a replacement, you can convert the point-in-time configuration to a two-site, long-distance async operation. This minimizes data loss in case of a Site 2 failure, and the BC copy at Site 3 can maintain point-in-time snapshots of the data. You can then continue the point-in time process at Site 3.

After failover from Site 1, the application runs on the P-vol of the BC_1 device. This device was the S-vol of the CA_1 link. To eliminate problems with the CA_1 device group definitions, delete the CA_1 device group at this stage. Because a device can only be in one CA device group at any time, any other CA device groups that require this device will fail. You must use the **`deletepair -g`** *`ca_1_device_pair`* **`-R`** command to delete the device group. Doing so does not affect the application or the BC_1 device group, but it does remove all CA associations and return the CA device to simplex (SMPL) status.

After the deleting the CA_1 device group, you have two options to complete the reconfiguration from the MSDT Solution to a two-site, async solution:

- Move the application one more device to the right to avoid a long initial copy process over the slow, long distance links.
- Re-create the CA_2 link using the current BC_1 P-vol device as the source and perform initial copy to Site 3

option 1 – eliminate the time to perform initial copy over long distance link

To eliminate an extended time to copy all data over the long distance link it is possible to move the application to the S-vol of the BC_1 device group. Because there is no option to swap the functionalities of a BC device group in the XP array, you must use either the physical remapping of logical devices within the array or reconfiguration of the physical connections to the array to accomplish this task.

1. Stop the application.

   This ensure that no new transactions are performed to the P-vol of the BC_1 device group during the operation.

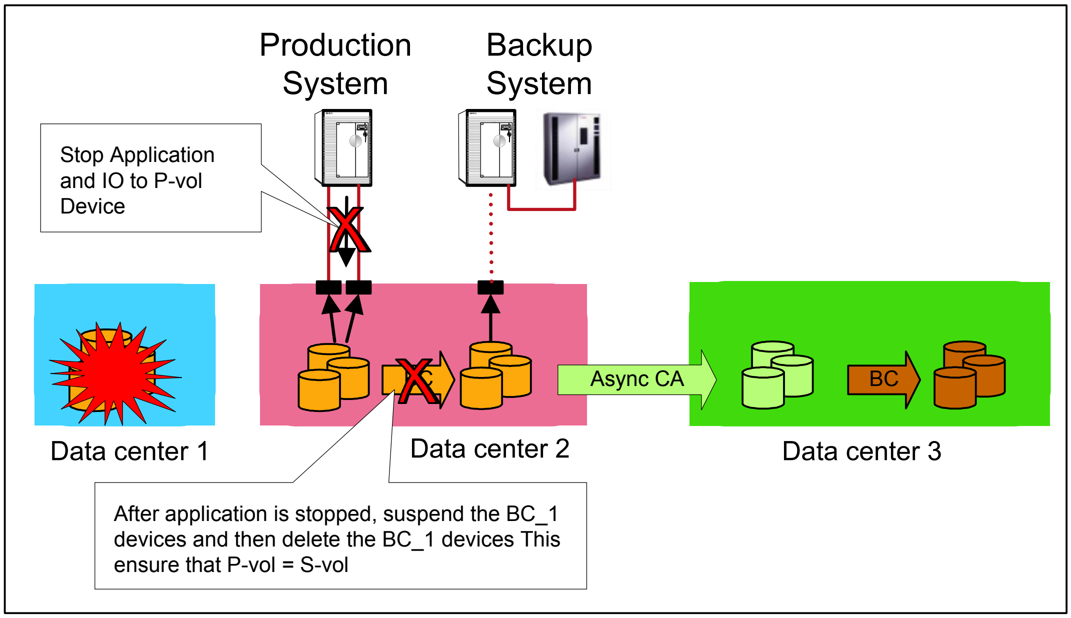
2. Suspend and then delete the BC_1 device group.

   It is important to first suspend the BC_1 devices before deleting the device groups. The XP BC software uses an asynchronous opportunistic copy method—the S-vol is never completely the same as the P-vol unless the device group is suspended.

   Use the MSDT tools **`suspendpair`** command (or **`pairsplit`** for RM) to perform this action.

   Delete the BC device group using the **`deletepair`** command. This will ensure that you do not have to change the RM configuration files after the re-mapping to attempt to delete the BC_1 groups and risking deleting the incorrect devices or missing some devices in the delete process.

Use the MSDT Tools **`deletepair`** command (or **`pairsplit -S`** for RM) to perform this action. Take note of all device ID (CU:Ldev) numbers, and ensure that the correct devices are used after you remap the devices.

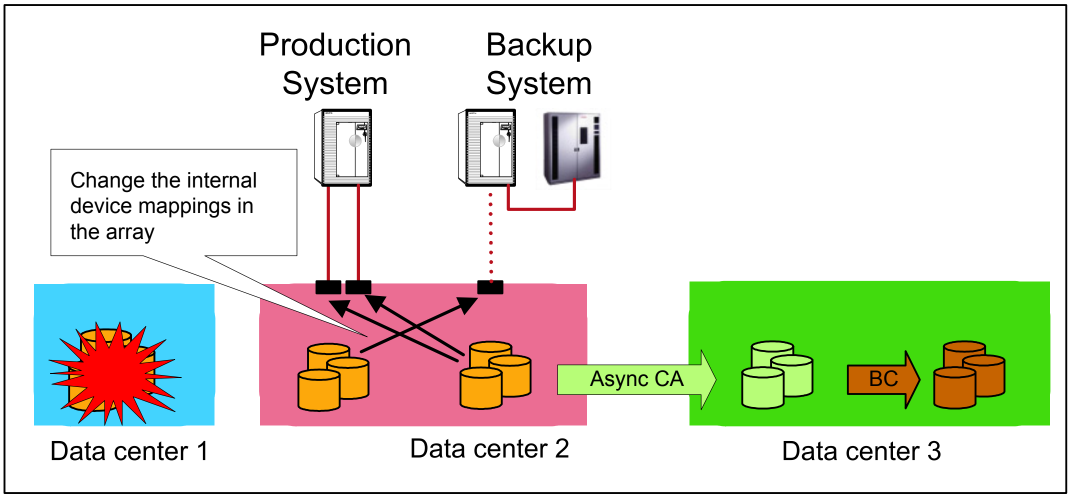

Production System
Backup System
Stop Application and IO to P-vol Device
Async CA
BC
Data center 1
Data center 2
Data center 3
After application is stopped, suspend the BC_1 devices and then delete the BC_1 devices This ensure that P-vol = S-vol

Under normal operations, you must map the S-vol of the BC_1 device group to a port on the array. This port can be on a backup server doing the backups for this application, or in some cases, there might not even be a host connected to a port. The XP array does not allow devices to be part of BC or CA device groups if they are not mapped to a physical port, even a port that is not in use at all, or a backup server that was using this port.

There are two options for changing the host access to the devices.

- **Option 1: Remap all the device files internally to the array**

  Remap all devices using the CommandView XP software, taking care to map the device to the new port first and then removing the device from the old port. In doing this, you guarantee that device is always mapped to at least one port. The CA_2 P-vol devices must be mapped to at least one port at all times. removing the device mapping from the existing port, and mapping the device to the new port. Take care to map devices with the same target LUN numbers to ensure that device files remain the same.



- **Option 2: Change physical port connections**

Swapping the fiber connections from the host to the array from one port to another enables the host to access the devices on that port. This sort of reconfiguration is much faster than remapping devices, and presents less risk of problems. Change the physical host connections to the array. The physical changing of cable connections might require some internal remapping to ensure dual paths to the devices. By swapping the fiber connections on the array, the production system can get access to the data on the CA_2 P-vol (old BC_1 S-vol).

Once the application host can access the old BC_1 S-vol devices, and LVM configuration is repaired on the host, restart the application and test data integrity.

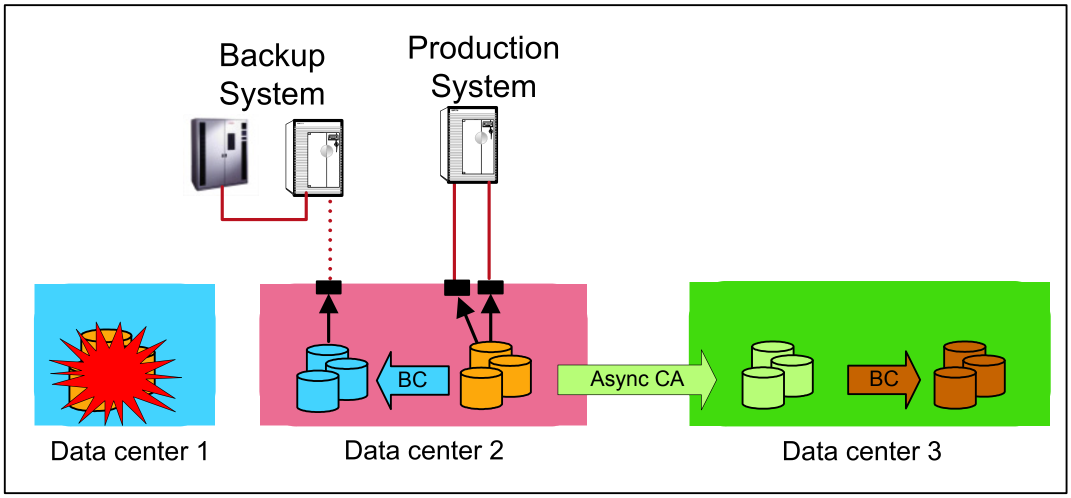
At this point, the application is running on the P-vol devices for the CA_2 device group and is replicated Asynchronously to Site 3. You can use the BC_2 devices at Site 3 to create point-in-time copies of the CA_2 device group on regular intervals to ensure that there is always valid copy on the remote site.

If the long distance links fail for any reason, the CA_2 devices suspend with an error state (`PSUE`). Although the data on the Site 3 is consistent and usable, no new updates will reach this site until you manually resync the device groups. Changes to the P-vol are kept in a bitmap file, and all ordering of IO is lost. During the resync process, a delta resync occurs with out-of-order data. If a failure happens during this time, the data at Site 3 in the S-vol of CA_2 will be corrupt and not usable.

In this event, use the BC_2 devices for recovery. Once the `PAIR` status is reached, order is restored and the data is usable in the event of a failure.

If required, the BC_1 device group can be re-created and used at Site 2 for local backup or other usages. This could also be used in the re-establishing of links to a new Site 1 array.

Use the MSDT Tools **`createpair`** command (or **`paircreate`** for RM) to perform this task.



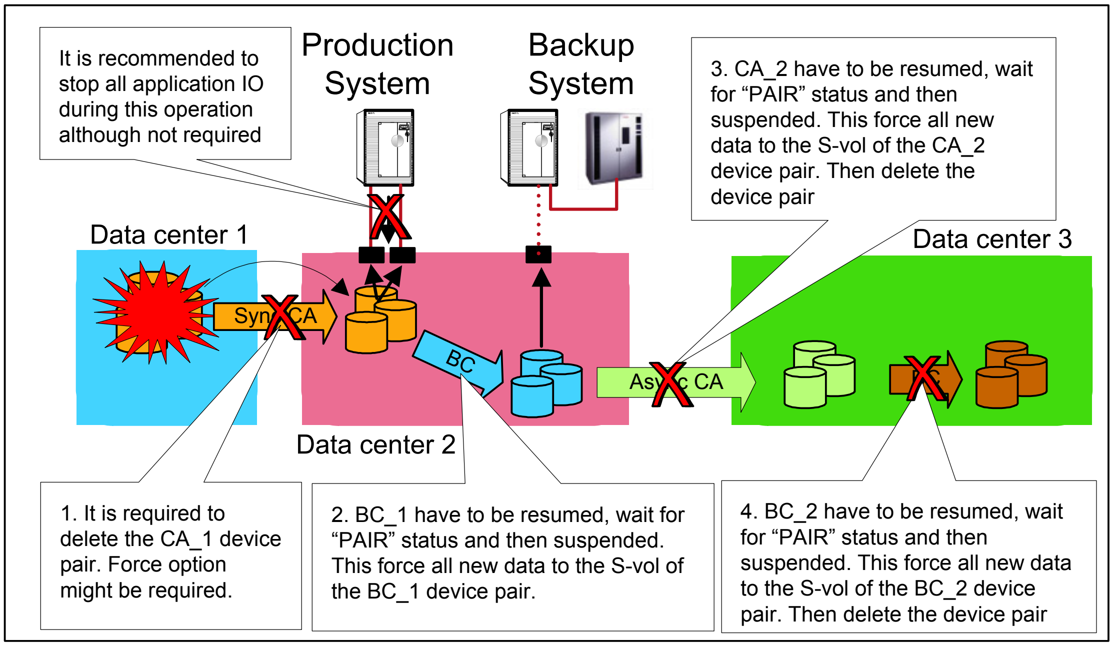
option 2 – re-create long distance group

If the time to complete a full initial copy over the long distance link is not an issue, and you do not want to go through the effort of re-mapping the devices, create a new CA_2.

To create the CA link from the BC_1 P-vol devices to the CA_2 S-vol device, you must delete the CA_1 device groups. A device cannot be in two CA device groups at the same time. You may need to delete these device groups using the **`-R`** option with the **`deletepair`** command.

First, complete a manual cycle process to ensure that the latest information is available at Site 3 on the BC_2 device group. Stop host IO during this process to ensure that recoverability is possible at Site 3 in case of a disaster. If the application cannot be down for that period, continue the process with the application running at Site 2. This operation is not dependant on application downtime, and can be performed with the application running.

1. Resume the BC_1 device group if it is in SUSPEND status. Perform BC_1 split with appropriate pre-exec commands. If the application is running, create a point-in time copy of the latest production information.
2. 2 Resume the CA_2 device group and wait for PAIR status. When PAIR status is reached, suspend the CA_2 device group and delete this group with the **deletepair** command.
3. Resume the BC_2 device group and wait for PAIR status. When PAIR status is reached, suspend and delete the device group. The S-vol of the BC_2 device group now has a copy of the production data and can be used to recover from in the event of a disaster. Delete both the CA_2 and BC_2 device group to create a new CA_2 device group and eventually a new BC_2 device group.



It is recommended to stop all application IO during this operation although not required

3. CA_2 have to be resumed, wait for "PAIR" status and then suspended. This force all new data to the S-vol of the CA_2 device pair. Then delete the device pair

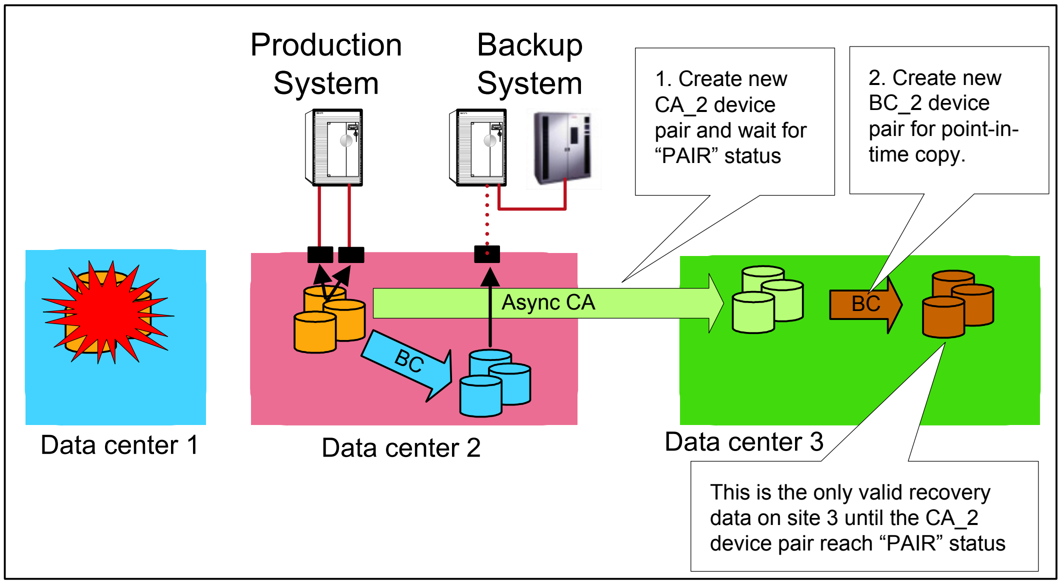
1. It is required to delete the CA_1 device pair. Force option might be required.

2. BC_1 have to be resumed, wait for "PAIR" status and then suspended. This force all new data to the S-vol of the BC_1 device pair.

4. BC_2 have to be resumed, wait for "PAIR" status and then suspended. This force all new data to the S-vol of the BC_2 device pair. Then delete the device pair

Reconfigure the RM instance to create a new device group between the BC_1 P-vol and old CA_2 S-vol devices. Create the Async device group, and wait for it to reach PAIR status. During the copy process, the old BC_2 S-vol device is the only valid data at Site 3.

After the CA_2 device group reaches PAIR status the BC_2 device group can be re-created and used to save a point-in-time copy of the data at Site 3.



1. Create new CA_2 device pair and wait for "PAIR" status

2. Create new BC_2 device pair for point-in-time copy.

This is the only valid recovery data on site 3 until the CA_2 device pair reach "PAIR" status

site 2 inoperable — route emergency connection from site 1 to site 3

A catastrophic disaster at Site 2, array failure Site 2, or a failure of the links between Sites 1 and 2 results in an IO fencing of the application at Site 1 — stopping all IO for the application as well as stopping the actual application. IO fencing is a feature of synchronous replication that stops all IO to the P-vol of a device group if it is not possible to write the IO to the S-vol of the device. This ensures total consistency of data between sites, but at the expense of application downtime. A local restart of the application at Site 1 overwrites the IO fencing and enables the application to continue operating on unprotected, non-replicated devices. This can be performed by means of a manual startup of the cluster package with a force flag file in place to overwrite the default data protection.

At this stage the data is unprotected, and the point-in-time copy process cannot continue. This is an undesirable situation—the application should not remain active in this environment for a long time. The customer is at risk that a subsequent disaster will wipe out an unrecoverable amount of data.

A possible solution to this situation is to provision a standby or temporary link from Site 1 to Site 3. You can have this link in place at all times, and have the physical configuration completed where possible. In the case of failure, re-create the device groups, or provide this link shortly after a failure and reconfigure of the device groups.

The following example assumes the solution model configuration (described in the “solution overview” on page 5), where data is replicated from Site 1 to 2 and then to 3.

To set up a temporary link between Sites 1 and 3:

1. Provide the physical links between sites.

   You can install the physical links before a failure occurs, and leave it there for emergency use. There could also be links that are actively used by other applications, and are available in an emergency. The links can also be provisioned after the disaster. This will take the longest time to recover the protection of the application.

2. Create the logical paths between arrays.

   You can only create the logical links (RCU) between arrays after the physical links are provided. The RCU is created for each CU (control unit) in the array and is limited to only four links per CU. Because of this limitation, you may need to first delete the existing broken links to Site 2, and then create new links to Site 3.

3. Create the device groups.

   Once the physical and logical links are configured, you can create the device groups by reconfiguring the RM instances and running the **`createpair`** command.

To reconfigure the RCU, you need a RCU configuration from Site 1 to 2 (for normal operations) and from Site 1 to 3 (for standby operations) for normal swap operations it would be recommended to have a RCU defined from site 2 to 1 (for MetroCluster) and from 2 to 3 for CA_2 link, and a RCU from Site 3 to 2 (for CA_2 link) and from Site 3 to 1 for the standby link. Because the RCU configuration is limited to a maximum of four RCU configurations, it might not be possible to have all RCU definitions configured all the time and therefore need to be configured only when it is required.

To create the CA device group between Site 1 and 3 we need to delete the remaining CA_1 device group definitions on the array at Site 1 and the remaining CA definitions in the array at Site 3, this can be performed with a **`deletepair -P`** and **`-R`** option on each site. Once the device group definitions is cleared out we need to do a configuration change to the RM setup and ensure that the P-vol devices from CA_1 is in the same device group definition as the S-vol devices from CA_2 and that the two RM instances can communicate with each other. To create the new CA group, the BC_2 device group must be deleted (the target device cannot be part of another group). If the RCU configuration is correct and the links are operational, then use the **`paircreate`** command to create the group from Site 1 to Site 3. When the initial copy process completes, you can re-create the BC_2 group.